SARA Chipset
Technical Manual

TranSwitch Corporation
8 Progress Drive
Shelton, CT 06484

Tel: (203) 929-8810
Fax: (203) 926-9453
E-Mail: mktg@txc.com
Web: http://www.transwitch.com

SARA-S/R
Part Numbers: TXC-05501/05601

Document Number: TXC-95000-TM
Edition 6, April 1996

SARA Chipset
Technical Manual

Notice

This Technical Manual describes current SARA chipset devices, which are now in production.

Please refer to the SARA Chipset Deviation List in Appendix H for the list of differences, as of the publication date of this Manual, between the actual performance of identified SARA chips and the specified functional performance in this Technical Manual. Please contact the TranSwitch Applications Engineering Department for current information.

SARA Chipset
Technical Manual

Edition 6

Issued: April 1996.

TranSwitch Corporation
8 Progress Drive
Shelton, CT 06484

Table of Contents

List of Figures ix

List of Tables xii

Preface xv

About This Manual xv
Audience of This Manual xv
Organization of This Manual xv

Chapter 1. Introduction 1-1

1.1 Overall Operation 1-2
1.2 ATM Adaptation Layer 1-3
1.2.1 AAL 3/4 1-4
1.2.2 AAL 5 1-6
1.3 The ATM Layer 1-8

Chapter 2. SARA Functional Description 2-1

2.1 Major SARA Features 2-1
2.2 Segmentation SARA Data Structures 2-1
2.3 SARA-S Operation 2-3
2.3.1 SARA-S Rate Queues 2-3
2.3.2 Peak and Average Metering 2-4
2.3.3 Congestion Control 2-5
2.4 Reassembly SARA Data Structures 2-5
2.5 SARA-R Operation 2-7
2.5.1 Exception Handling 2-8
2.5.2 Packet Timeout 2-8

Chapter 3. Hardware Description 3-1

3.1 Segmentation SARA Hardware Description 3-1
3.1.1 Segmentation SARA Internal Block Description 3-1
3.1.2 Segmentation SARA Hardware Interfaces 3-2
3.1.2.1 Processor Interface 3-2
3.1.2.2 Cell Interface 3-3
3.1.2.3 Control Memory Interface 3-3
3.1.2.4 Packet Memory Interface 3-4
3.1.2.5 Congestion Control Interface 3-5
3.1.2.6 Constant Bit Rate Traffic Interface 3-5
3.1.2.7 Miscellaneous Signals 3-6
3.2 Reassembly SARA Hardware Description 3-7
3.2.1 Reassembly SARA Internal Block Description 3-7
3.2.2 Reassembly SARA Hardware Interfaces 3-8

3.2.2.1 Processor Interface 3-8
3.2.2.2 Cell Interface 3-9
3.2.2.3 Control Memory Interface 3-9
3.2.2.4 Packet Memory Interface 3-10
3.2.2.5 Congestion Control Interface 3-11
3.2.2.6 Miscellaneous Signals 3-11
3.3 Hardware Interface Operation 3-12
3.3.1 Processor Interface 3-12
3.3.2 Control Memory Interface 3-14
3.3.2.1 Address Bus CA(23:1).....3-14
3.3.2.2 Data Bus CD(17:0).....3-14
3.3.2.3 Control Interface.....3-14
3.3.2.4 Synchronous Mode.....3-14
3.3.2.5 Asynchronous Mode.....3-22
3.3.3 Packet Memory Interface 3-23
3.3.3.1 Data Bus PD(35:0).....3-23
3.3.3.2 Address Bus PA(15:0) 3-23
3.3.3.3 Control Signals 3-24
3.3.3.4 Synchronous Mode 3-24
3.3.3.5 Asynchronous Mode 3-28
3.3.4 Link/Cell Interface 3-28
3.3.4.1 SARA-S 3-29
3.3.4.2 SARA-R 3-30
3.3.5 Congestion Control Interface 3-33
3.3.6 Constant Bit Rate Interface 3-33
3.3.6.1 SARA-S 3-33
3.3.6.2 SARA-R 3-34
3.3.7 Cross Reference by Pin Name (SARA-S) 3-36
3.3.8 SARA-S Pin Configuration 3-37
3.3.9 Cross Reference by Pin Name (SARA-R) 3-38
3.3.10 SARA-R Pin Configuration 3-39
3.3.11 SARA Packaging Dimensions 3-40
3.4 DC Characteristics 3-41
3.5 AC Characteristics 3-42
3.5.1 Switching Characteristics 3-42
3.5.2 Interface Signal Timings 3-47

Chapter 4. SARA-S Register Descriptions 4-1

4.1 Mode Registers 4-3
4.1.1 Mode Register 0 4-4
4.1.2 Mode Register 1 4-7
4.2 Interrupt Status Register 4-9

4.3 State Register 4-11
4.4 Interrupt Mask Register 4-13
4.5 Rate Queue Registers 4-15
4.6 Queue Base Address Register 4-17
4.7 Descriptor Table Base Address Register 4-17
4.8 Virtual Circuit (VC) Table Base Address Register 4-18
4.9 Packet Memory Address Match Register 4-21
4.10 Command Register 4-21
4.11 Communication Queues 4-22
4.12 Cell Counter Registers 4-23
4.13 Constant Bit Rate Address Registers 4-23
4.14 Rate Queue Starting Descriptor Number Diagnostic Registers 4-24
4.15 Current Descriptor Number Diagnostic Register 4-24
4.16 CC_Label Diagnostic Register 4-24

Chapter 5. SARA-R Register Descriptions 5-1

5.1 Mode Registers 5-3
5.1.1 Mode Register 0 5-4
5.1.2 Mode Register 1 5-7
5.2 Interrupt Status Register 5-9
5.3 State Register 5-11
5.4 Interrupt Mask Register 5-13
5.5 Queue Base Address Register 5-15
5.6 Descriptor Table Base Address Register 5-15
5.7 VC Table Base Address Register 5-16
5.8 VP Base Address Register 5-18
5.9 VP Filter Register 5-18
5.10 Reassembly Table Base Register 5-19
5.11 Packet Memory Address Match Register 5-22
5.12 Command Register 5-22
5.13 Dropped Packet Count Register 5-23
5.14 Error Count Register 5-23
5.15 Dropped CBR Cells Count Register Queue 5-23
5.16 Received Cell Count Register 5-23
5.17 Small Buffer Check Register 5-23
5.18 Large Buffer Check Register 5-24
5.19 Large Buffer Size Register 5-24
5.20 Packet Timeout Count Register 5-24
5.21 Packet Aging Interval Count Register 5-24
5.22 Timeout Index Register 5-24
5.23 Timeout Index Range Register 5-24
5.24 Communication Queues 5-25
5.24.1 Small Free Descriptor Queue 5-26
5.24.2 Large Free Descriptor Queue 5-26

5.24.3 Packet Complete Queue 5-26
5.24.4 Exception Queue 5-26
5.25 CBR and Raw Cell Queue Base Address Registers 5-27
5.26 CBR Queue Registers 5-27
5.27 Raw Cell Queue Registers 5-28
5.28 OAMCC_CHK Register 5-28

Chapter 6. SARA-S Software Interfaces 6-1

6.1 SARA-S Control Memory 6-1
6.1.1 Buffer Descriptor Table 6-1
6.1.2 Virtual Circuit (VC) Table: 6-4
6.1.3 Average Metering Function 6-7
6.1.4 Packet Ready Queue 6-9
6.1.5 Transmit Complete Queue 6-11
6.2 SARA-S Packet Memory 6-12
6.2.1 Packet Data Buffers 6-12
6.2.2 OAM (Test) Data 6-13
6.2.3 CBR Data 6-13
6.3 SARA-S Basic Software Functions 6-14
6.3.1 Startup Initialization 6-14
6.3.1.1 Allocating Buffers in the Packet Memory 6-15
6.3.1.2 Allocating and Initializing the Buffer Descriptors 6-15
6.3.1.3 Allocating the VC Table 6-15
6.3.1.4 Setting-up the Communication Queues 6-16
6.3.1.5 Setting the Peak-Rate Metering Parameters 6-16
6.3.1.6 Initializing the Mask Bits 6-16
6.3.1.7 Setting the Mode Registers 6-17
6.3.2 Virtual Circuit Setup 6-17
6.3.3 Transmitting a Packet 6-17
6.3.4 Transmitting OAM Cells 6-18
6.3.5 Transmitting a CBR Cell 6-19

Chapter 7. SARA-R Software Interfaces 7-1

7.1 SARA-R Control Memory 7-1
7.1.1 Buffer Descriptor Table 7-1
7.1.2 Reassembly Table Pointer/Descriptor Table Entry Generation 7-5
7.1.3 VC Table 7-7
7.1.4 VP Table 7-8
7.1.5 Reassembly Table 7-9
7.1.6 Small/Large Free Descriptor Queues 7-10
7.1.7 Packet Complete Queue 7-11
7.1.8 Exception Queue 7-11

7.2 SARA-R Packet Memory 7-13
7.2.1 Packet Data for Reassembly 7-13
7.2.2 CBR Data 7-14
7.2.3 Raw Cell Queue 7-15
7.3 SARA-R Basic Software Functions 7-16
7.3.1 Startup Initialization 7-16
7.3.1.1 Allocating Buffers in the Packet Memory 7-16
7.3.1.2 Allocating and Initializing the Buffer Descriptors 7-17
7.3.1.3 Allocating the VP Table 7-17
7.3.1.4 Allocating the VC Table 7-17
7.3.1.5 Allocating the Reassembly Table 7-18
7.3.1.6 Setting-up the VC Table 7-18
7.3.1.7 Setting-up the VP Table 7-18
7.3.1.8 Setting-up the Reassembly Table 7-18
7.3.1.9 Setting-up the Communication Queues 7-18
7.3.1.10 Initializing the Mask Bits 7-19
7.3.1.11 Setting the Mode Registers 7-19
7.3.2 Receiving a Cell 7-19
7.3.3 Receiving a Packet 7-20
7.3.4 Receiving Constant Bit Rate Traffic 7-21
7.3.5 Receiving Congestion Notification Cells 7-21
7.3.6 Exception Queue Handling 7-22

Appendix A. Control Memory Size Examples A-1

Appendix B. Packet Memory Interface Example B-1

Appendix C. Link Interface Example C-1

Appendix D. Control Memory Example D-1

Appendix E. SARA Performance Examples E-1

Appendix F. Congestion Control in SARA F-1

Appendix G. SARA Time-Out Operation G-1

Appendix H. SARA Chipset Deviation List H-1

Standards Documentation Sources Sources-1

List of Technical Manual Changes Changes-1

Glossary Glossary-1

List of Figures

Figure 1-1. Typical ATM Network Interface Unit Implementation 1-2
Figure 1-2. Protocol Model for Variable Bit Rate AAL Type 3/4 and Type 5 1-4
Figure 1-3. Structure of the AAL 3/4 CPCS-PDU 1-4
Figure 1-4. CRC-32 in an 802.6 Frame 1-5
Figure 1-5. TranSwitch Proprietary CRC-32 1-5
Figure 1-6. The Segmentation Process for AAL 3/4 1-5
Figure 1-7. Structure of the AAL 3/4 SAR_PDU 1-6
Figure 1-8. Structure of the AAL 5 CPCS_PDU 1-7
Figure 1-9. Segmentation Process for AAL 5 1-7
Figure 1-10. Structure of an ATM Cell at the User Network Interface 1-8
Figure 2-1. Segmentation SARA Data Structures 2-2
Figure 2-2. Typical Rate Queue Data Structure 2-4
Figure 2-3. SARA-R Data Structures 2-6
Figure 3-1. Segmentation SARA Block Diagram 3-1
Figure 3-2. Block Diagram of Reassembly SARA 3-7
Figure 3-3. Processor Interface Functional Timing 3-13
Figure 3-4. Control Memory Interface—Single Access Read and Write—Normal Write Mode 3-15
Figure 3-5. Control Memory Interface—Single Access Read and Write—Early Write Mode 3-16
Figure 3-6. Control Memory Interface—Multiple Reads 3-17
Figure 3-7. Control Memory Interface—Multiple Writes 3-18
Figure 3-8. Control Memory Interface—Single Read/ Multiple Writes—Normal Write Signal 3-19
Figure 3-9. Control Memory Interface—Single Read/ Multiple Writes—Early Write Signal 3-20
Figure 3-10. Control Memory Interface—Without Last Access 3-21
Figure 3-11. Packet Memory Interface—Address Driven on Data Bus During Bus Access 3-26
Figure 3-12. Packet Memory Interface—Bus Preempted by De-assertion of PGRT 3-27
Figure 3-13. Cell Interface—SARA-S 3-31
Figure 3-14. Cell Interface—SARA-R 3-32
Figure 3-15. CCDATA Format 3-33
Figure 3-16. Constant Bit Rate Interface—SARA-S 3-35
Figure 3-17. 208-pin PQFP Pin Configuration—SARA-S 3-37
Figure 3-18. 208-pin PQFP Pin Configuration—SARA-R 3-39
Figure 3-19. SARA Packaging 3-40

Figure 3-20A.Congestion-Control Interface Signals Between the SARA-S and SARA-R ("A" Versions) 3-48
Figure 3-21A.Constant Bit Rate Interface Signals of SARA-S and SARA-R ("A" Versions) 3-49
Figure 3-22A.Processor-Interface Signal Timing for "A" Versions 3-50
Figure 3-23A.Cell-Interface Signals for the SARA-R ("A" Version) 3-51
Figure 3-24A.Cell-Interface Signals for the SARA-S ("A" Version) 3-52
Figure 3-25A.Control Memory Interface Signal Timing-Read Operation for "A" Versions 3-53
Figure 3-26A.Control Memory Interface Signal Timings-Write Operation for "A" Versions 3-54
Figure 3-27A.Packet Memory Interface Signal Timings-Write Operation for the SARA-R ("A" Version) 3-55
Figure 3-28A.Packet Memory Interface Signal Timings-Read Operation for the SARA-S ("A" Version) 3-56
Figure 3-20B.Congestion-Control Interface Signals Between the SARA-S and SARA-R ("B" Versions) 3-57
Figure 3-21B.Constant Bit Rate Interface Signals of SARA-S and SARA-R ("B" Versions).....3-58
Figure 3-22B.Processor-Interface Signal Timing for "B" Versions 3-59
Figure 3-23B.Cell-Interface Signals for the SARA-R ("B" Version) 3-60
Figure 3-24B.Cell-Interface Signals for the SARA-S ("B" Version) 3-61
Figure 3-25B.Control Memory Interface Signal Timing-Read Operation for "B" Versions 3-62
Figure 3-26B.Control Memory Interface Signal Timings-Write Operation for "B" Versions 3-63
Figure 3-27B.Packet Memory Interface Signal Timings-Write Operation for the SARA-R ("B" Version) 3-64
Figure 3-28B.Packet Memory Interface Signal Timings-Read Operation for the SARA-S ("B" Version) 3-65
Figure 4-1. Mode Register 0 Data Bits 4-4
Figure 4-2. Mode Register 1 Data Bits 4-7
Figure 4-3. Interrupt Status Register Bit Positions 4-10
Figure 4-4. State Register Bit Positions 4-12
Figure 4-5. Bit Positions for SARA-S Interrupt Mask Register 4-13
Figure 4-6. Rate Queue Register 4-15
Figure 4-7. SARA-S Queue Address Generation 4-17
Figure 4-8. SARA-S Descriptor Table Address Generation 4-18
Figure 4-9. SARA-S VC Table Entry Address Generation 4-20
Figure 4-10. SARA-S Packet Memory Address Match Register 4-21
Figure 4-11. Transmit Complete Queue 4-22

Figure 4-12. SARA-S Packet Ready and Transmit Complete Queue Operational States 4-23
Figure 5-1. SARA-R Mode Register 0 Bit Positions 5-3
Figure 5-2. SARA-R Mode Register 1 Bit Positions 5-7
Figure 5-3. Interrupt Status Register Bit Positions 5-10
Figure 5-4. SARA-R State Register Bit Positions 5-12
Figure 5-5. Mask Register Bit Positions 5-13
Figure 5-6. SARA-R Queue Address Generation 5-15
Figure 5-7. SARA-R Descriptor Table Address generation 5-15
Figure 5-8. SARA-R VC Table Address Generation 5-17
Figure 5-9. SARA-R VP Base Address Register 5-18
Figure 5-10. VP Filter Register 5-19
Figure 5-11. Reassembly Table Pointer Construction 5-21
Figure 5-12. SARA-R Packet Memory Address Match Register 5-22
Figure 5-13. Timeout Index Range Register 5-25
Figure 5-14. SARA-R Communication queues - Operational States 5-26
Figure 5-15. Entries in Exception Queue 5-27
Figure 5-16. CBR and Raw Cell Queue Address Generation 5-27
Figure 5-17. OAMCC_CHK Register 5-28
Figure 6-1. Buffer Descriptors Mapped into Two Regions 6-1
Figure 6-2. Descriptor Table Entry Organization 6-2
Figure 6-3. Descriptor Mode Bits 6-2
Figure 6-4. Virtual Circuit Table Entry 6-4
Figure 6-5. Virtual Circuit Mode Bits Word 6-5
Figure 6-6. Packet Ready Queue Entry 6-9
Figure 6-7. Communication Queue Operational States 6-10
Figure 6-8. Address Generation of Queue Entry 6-11
Figure 6-9. Transmit Complete Queue Entry 6-11
Figure 6-10. Address of Data in Packet Memory 6-12
Figure 6-11. Address of SARA-S CBR Data in Packet Memory 6-13
Figure 6-12. Buffers in Packet Memory 6-15
Figure 6-13. AAL 3/4 Frame Without CRC-32 6-17
Figure 6-14. AAL 3/4 802.6 Frame With CRC-32 6-18
Figure 6-15. AAL 3/4, Non-802.6 Frame with CRC-32 6-18
Figure 6-16. AAL 5 Frame with CRC-32 6-18
Figure 7-1. Receive Descriptor Table Entry Organization 7-2
Figure 7-2. Descriptor Mode/Status Bits 7-2
Figure 7-3. Reassembly Table Pointer Process 7-7
Figure 7-4. VC Table Entry 7-8

Figure 7-5. VP Table Entry 7-9
Figure 7-6. Reassembly Table Entry 7-9
Figure 7-7. Small/Large Free Descriptor and Packet Complete Queue Entry 7-11
Figure 7-8. Exception Queue Entry 7-12
Figure 7-9. Address of SARA-R CBR/Raw Cell Data in Packet Memory 7-14
Figure 7-10. Packet Memory Data Structure Alignments 7-16
Figure 7-11. Control Memory Data Structure Alignments 7-19
Figure A-1. SARA-R Control Memory Map for Small System A-3
Figure A-2. SARA-S Control Memory Map for Small System A-3
Figure A-3. SARA-R Control Memory Map for Large System A-6
Figure A-4. SARA-S Control Memory Map for Large System A-6
Figure B-1. Packet Memory Interface Example B-1
Figure C-1. Transmit and Receive Link Interface Example C-1
Figure C-2. Alternative Interface Applications C-2
Figure D-1. Control Memory Example D-1
Figure D-2. Control Memory State Diagram D-2

List of Tables

Table 3-1. SARA-S Processor Interface Pin Descriptions 3-2
Table 3-2. SARA-S Cell Interface Pin Descriptions 3-3
Table 3-3. SARA-S Control Memory Interface Pin Descriptions 3-3
Table 3-4. SARA-S Packet Memory Interface Pin Descriptions 3-4
Table 3-5. SARA-S Congestion Control Interface Pin Descriptions 3-5
Table 3-6. SARA-S Constant Bit Rate Transfer Interface Pin Description 3-5
Table 3-7. SARA-S Miscellaneous Signals Pin Description 3-6
Table 3-8. SARA-R Processor Interface Pin Descriptions 3-8
Table 3-9. SARA-R Cell Interface Pin Description 3-9
Table 3-10. SARA-R Control Memory Interface Pin Descriptions 3-10
Table 3-11. SARA-R Packet Memory Interface Pin Descriptions 3-10
Table 3-12. SARA-R Congestion Control Interface Pin Descriptions 3-11
Table 3-13. SARA-R Miscellaneous Signals Pin Descriptions 3-11
Table 3-14. Interactions of PREQ, PGRT, and PLWADR with the Packet Data Bus PD(31:0) 3-23
Table 3-15. Mode Bit/PLWADR Interactions 3-24
Table 3-16. Byte Ordering of Data for Link/Cell Interface 3-29
Table 3-17. Cross Reference by Pin Name for SARA-S 3-36
Table 3-18. Cross Reference by Pin Name for SARA-R 3-38
Table 3-19. SARA Packaging Dimensions 3-40
Table 3-20. SARA DC Characteristics Over Commercial Operating Ranges 3-41
Table 3-21. SARA Capacitance Characteristics 3-41
Table 3-22A.Switching Characteristics for “A” Versions over Commercial Operating Conditions 3-42
Table 3-22B.Switching Characteristics for "B" Versions over Commercial Operating Conditions 3-45
Table 4-1. SARA-S Internal Registers 4-1
Table 4-2. SARA-S Diagnostic Registers 4-3
Table 4-3. Interactions of PREQ, PGRT, and PLWADR with Packet Data Bus PD(31:0)—SARA-S 4-5
Table 4-4. Rate Queue Registers 4-15
Table 4-5. Peak Segmentation Rates for Different Prescaler Values 4-16
Table 4-6. Rate Queue Register Programming Values 4-17
Table 4-7. Number of VC Table Entries for Various Values of VC_LKUP_BASE Register 4-19
Table 5-1. SARA-R Internal Registers 5-1

Table 5-2. Interactions of PREQ, PGRT, and PLWADR with Packet Data Bus PD(31:0)—SARA-R 5-5
Table 5-3. Number of VC Table Entries for Various Values of the VC_LKUP_BASE Register 5-18
Table 5-4. Reassembly Table Pointer Construction 5-20
Table 6-1. Congestion Control Mode 6-5
Table 6-2. Rate Queue Modes 6-6
Table 6-3. Software Congestion Level Modes 6-6
Table 6-4. Average Metering Rate 6-8
Table 6-5. Completion Codes for Transmit Complete Queue 6-11
Table 6-6. Packet Memory Data in Big-Endian 32-bit and 16-bit Modes 6-13
Table 6-7. Packet Memory Data in Little-Endian 32-bit and 16-bit Modes 6-13
Table 6-8. Packet Memory Data in Big-Endian 32-bit and 16-bit Modes for CBR Cells 6-14
Table 6-9. Packet Memory Data in Little-Endian 32-bit and 16-bit Modes for CBR Cells 6-14
Table 6-10. Setting the Number of VCs for SARA-S 6-16
Table 7-1. Number of MID Bits in Reassembly Pointer 7-8
Table 7-2. Number of VCI Bits in the Reassembly Pointer 7-9
Table 7-3. Reassembly Table Entry State Bits Description 7-10
Table 7-4. Error Code Explanations for Exception Queue 7-12
Table 7-5. Packet Memory Data in Big Endian 32-bit and 16-bit Modes 7-13
Table 7-6. Packet Memory Data in Little Endian 32-bit and 16-bit Modes 7-13
Table 7-7. Packet Memory 32-bit and 16-bit Data Format of CBR /Raw Cells—Big Endian Mode 7-15
Table 7-8. Packet Memory 32-bit and 16-bit Data Format of CBR /Raw Cells—Little Endian Mode 7-15
Table 7-9. Setting the Number of VCs for SARA-R 7-18
Table 7-10. Receive Descriptor Error Status Bits 7-20
Table A-1. SARA-R Control Memory Data Structures A-1
Table A-2. SARA-S Control Memory Data Structures A-1
Table A-3. SARA-R Control Memory Size Example A-2
Table A-4. SARA-S Control Memory Size Example A-2
Table D-1. Explanation of Symbols in the Control Memory State Diagram D-1
Table E-1. SARA-S Throughput Performance for AAL 3/4 ..... E-1
Table E-2. SARA-R Throughput Performance for AAL 3/4 ..... E-1

Preface

About This Manual

This manual describes the characteristics of the SARA ATM interface devices together with their hardware and software features. Sufficient detail is provided to facilitate their incorporation in the design of ATM network interfaces.

In this edition of the manual (Edition 6), two functional versions of the SARA chipset are described. The original functional “A” version, which operates at a maximum clock speed of 20 MHz, was the only version described in earlier editions. This edition also describes the functional “B” version introduced in 1995, which operates at a maximum clock speed of 33 MHz. The two functional versions are very similar, and have common descriptions in this manual, except for differences in their part numbers (Section 1), DC characteristics (Section 3.4) and AC characteristics (Section 3.5).

Audience of This Manual

This manual is written for hardware and software engineers who design ATM (Asynchronous Transfer Mode) interface devices. Specifically, the chipset described in this manual is intended for application in ATM network interfaces for terminals, bridges, routers, workstations or PCs which may attach to an ATM network.

Organization of This Manual

Chapter 1—Introduction

This chapter introduces the SARA devices, and describes their application in a typical network interface unit (NIU) design. The functions of the Segmentation and Reassembly SARA devices (SARA-S, TXC-05501; and SARA-R, TXC-05601) are briefly described. The segmentation and reassembly process required to achieve packet transmission across an ATM network as defined in the ANSI draft standards for B-ISDN are briefly described.

Chapter 2—SARA Functional Description

Chapter 2 provides an overview of the most important features and functions of the SARA chipset. The features of both the SARA-S and SARA-R chips, taken together, are summarized. The individual features of SARA-S and SARA-R are presented next, with particular attention to the interfaces between each SARA device and the packet and control memory. The data structures in the packet and control memories are outlined.

The data structures and interactions with the SARA-S rate queue structure are described in detail, together with the methods of peak and average metering and congestion control. The process of cell reassembly into packets by SARA-R is discussed. Attention is given to the exception handling for errors in the cell syntax, in the received packets, and those errors not associated with packets. The buffer recovery mechanism in SARA-R for packets that have “timed-out” is also given.

Chapter 3—Hardware Description

Chapter 3 describes the SARA chipset as hardware devices. Separately for both SARA-S and SARA-R, signals are grouped according to their associated interface, categorized as input, output or both, and their functions described. For SARA-S, the interfaces are the processor, cell, control memory, packet memory, congestion control, constant bit rate and miscellaneous. For SARA-R, the same interfaces apply except for the constant bit rate.

Each interface is then described in detail. The signals are described functionally and their timing is shown with respect to each other.

A cross reference table by pin name is given for both SARA-S and SARA-R together with pin configuration diagrams for both chips. The DC and capacitance characteristics of each chip are given. AC characteristics, including signal timings for all interfaces, are specified.

Chapter 4—SARA-S Register Descriptions

The SARA-S registers are described as a software reference. The register name, address, description, read/write type and reset value are provided for all the internal and diagnostic registers of each chip. A bit-by-bit description is presented for each register. Similar treatment is given for each queue.

Chapter 5—SARA-R Register Descriptions

The SARA-R registers are described as a software reference as the SARA-S registers are described in Chapter 4.

Chapter 6—SARA-S Software Interfaces

The interfaces from the SARA-S to control and packet memory data structures are described in considerably greater detail than that given in Chapter 2. For SARA-S, the control memory structures include the buffer descriptor table, the VC table, the packet ready queue, and the transmit complete queue.

The packet memory structures described for SARA-S include the packet data and CBR data.

Software functions are described. For SARA-S these include start-up initialization, virtual circuit setup, and transmitting a packet.

Chapter 7—SARA-R Software Interfaces

The SARA-R software interfaces are described as the SARA-S software interfaces are described in Chapter 6. For SARA-R, the control memory structures include the buffer descriptor table, the VP table, the reassembly table, the VC table, the small/large free descriptor queues, the packet complete queue and exception queue. The packet memory structures described for SARA-R are packet data, CBR data and raw cell data. SARA-R software functions described are start-up initialization, virtual circuit setup, and packet reception.

Appendices

Appendix A gives control memory size examples. In Appendix B a packet memory interface example is illustrated. Link interface and application examples are given in Appendix C. Appendix D shows a specific example of the SARA and a microprocessor interface to control memory, with a description of the state machine inside the arbitration and control block. SARA performance examples are given in Appendix E. Appendix F discusses congestion control in SARA. SARA Time-Out Operation is presented in Appendix G. The SARA Chipset Deviation List is presented in Appendix H.

Standards Documentation Sources

The addresses and telephone/fax numbers of the international organizations that publish telecommunication technical standards and reference documentation are listed.

List of Technical Manual Changes

This list of technical manual changes is a comprehensive list of all significant changes made since the last published edition of the SARA Manual.

Glossary

A glossary of ATM, data communications, and networking terms is given.

Chapter 1. Introduction

In this edition of the manual (Edition 6), two functional versions of the SARA chipset are described. The original functional "A" version, which operates at a maximum clock speed of 20 MHz, was the only version described in earlier editions. This edition also describes the functional "B" version introduced in 1995, which operates 65% faster at a maximum clock speed of 33 MHz, and consumes 30% less power, while retaining form, fit and function compatibility with the functional "A" version. The two functional versions are very similar, and have common descriptions in this manual, except for differences in their part numbers (described below), DC characteristics (Section 3.4) and AC characteristics (Section 3.5).

The part numbers which may be used for ordering SARA chipset devices are as follows:

	Functional "A" Parts	Functional "B" Parts
SARA-S	TXC-05501-ACPQ	TXC-05501-BCPQ
SARA-R	TXC-05601-ACPQ	TXC-05601-BCPQ

The two-chip SARA chipset provides a powerful and easy-to-use interface between packet-oriented equipment and ATM networks. SARA simplifies the design of high performance interfaces for computers, workstations, bridges, routers, etc.—conveniently providing segmentation and reassembly functions in compliance with both the T1S1.5 ATM Adaptation Layer Type 3/4 and Type 5 Common Part draft standards*,†. Most overhead functions associated with conversion to and from cells are provided with a high-performance, low-overhead ATM interface. This includes packet queueing, buffer management and DMA, CRC generation and checking, and those Asynchronous Transfer Mode (ATM) network-specific functions required to segment and reassemble cells over the attached network link. SARA supports the T1S1.5 ATM Adaptation Layer (AAL) Type 3/4 and Type 5 Common Parts, including the Common Part Convergence Sublayer (CPCS) and the Segmentation and Reassembly Sublayer (SAR). The SARA chipset can reassemble frames from an input stream of cells that originates from many interleaved frames.

SARA also supports some ATM layer functions, as described later in this chapter. ATM cells consist of a five-byte header and a 48-byte payload (53 bytes total).

The SARA chipset is designed to provide routers, bridges, computers, front-end processors and workstations with a high performance chipset to interface with an ATM network. Hardware support for the functions of frame-to-cell segmentation and reassembly in accordance with the requirements of Broadband ISDN (B-ISDN) and Switched Multi-Megabit Data Service (SMDS) are provided. SARA supports up to a maximum of 65,536 virtual circuits. Up to 8,191 virtual circuits may have packets queued for segmentation, with an equal number being concurrently reassembled. A throughput rate of at least 150 Mbps in each direction may be achieved (suitable for SONET STS-3).

In segmentation, the chipset receives both a packet and a corresponding packet descriptor. The packet is segmented into cells. The appropriate ATM cell header is added to each cell, and the cells are sent out on the transmission line interface.

In reassembly, the device receives a multiplexed cell stream from the transmission line interface. ATM layer operations and maintenance (OAM) cells are filtered out of the incoming cell stream. The incoming cell stream is then demultiplexed into its component streams using the virtual channel identifier (VCI) and the message identification (MID) field (AAL 3/4 only), or the virtual path identifier (VPI) and the VCI, to form the reassembly identifier. The cells in

*. T1S1.5/92-003R2 "Broadband ISDN—ATM Adaptation Layer 3/4 Common Part Functionality and Specification" May 1992.
†. T1S1.5/92-010 "Broadband ISDN—ATM Adaptation Layer 5 Common Part Functionality and Specification" May 1992.

each component stream are reassembled into frames using the ATM adaptation layer information fields. The AAL and ATM headers are removed and the completed frames queued for software processing.

Constant Bit Rate (CBR) traffic can also be accommodated by the SARA chipset. The SARA-S can add headers to an external queue of CBR payload cells and multiplex them with the Variable Bit Rate (VBR) cell stream using an interrupt-like mechanism. The SARA-R demultiplexes incoming CBR cells and stores them in an external CBR cell queue.

Both SARA chips are packaged in a 208-pin EIA-J Plastic Quad Flat Package (PQFP). They are both fabricated with one micron CMOS technology.

1.1 Overall Operation

Figure 1-1 shows an example of an ATM Network Interface Unit (NIU) implementation using SARA segmentation and reassembly devices (SARA-S and SARA-R). In some applications, a local processor may be included (not shown in Figure 1-1). If the application performance is not too demanding, the control RAM and packet memory can be externally combined into a single memory.

Data packets are queued, segmented, and reassembled in the packet memory. Buffer descriptors, virtual circuit parameters, and SARA operating variables are all stored in tables that reside in the control RAM. The processor-to-SARA communication queues also reside in the control RAM. The control RAM can be separated into two control RAMs (one RAM for segmentation and one for reassembly) externally, depending on the application performance requirements.

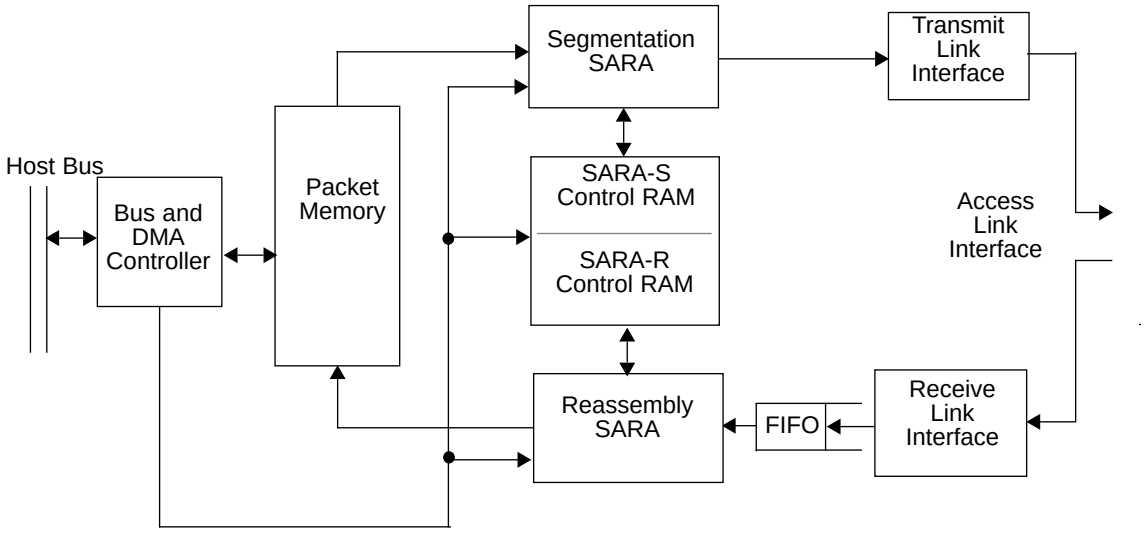


Figure 1-1. Typical ATM Network Interface Unit Implementation

A DMA controller places packets received across the host bus into buffers in the packet memory. The software informs SARA of the packet's location. The packet is then segmented into cells, and the cells are multiplexed with cells from other packets that are undergoing segmentation. Each cell carries the appropriate header information together with the adaptation layer fields according to the B-ISDN ATM Adaptation Layer protocol in the 48-byte cell payload. Up to eight peak emission rates (the inter-cell gap on each virtual circuit) may be programmed. Using a leaky bucket algorithm, the average cell emission rate can likewise be programmed for each virtual circuit. When segmentation is completed the packet buffer is returned to the software.

The reassembly device (SARA-R) reassembles an incoming multiplexed stream of ATM cells into packets using the virtual channel identifier (VCI) and the message identification (MID) field (AAL 3/4 only), or the virtual path identifier (VPI) and the VCI, to form the reassembly identifier. When a cell containing the beginning of a new packet is received, SARA-R fetches a buffer and DMAs the cell into it. Subsequent cells from the same packet are accumulated in the buffer. When a cell indicating the end of the packet is received, it is stored in the buffer, and the processor is informed. All of the adaptation layer fields are processed within the chip with all error conditions being monitored and reported.

After a packet is queued for segmentation, processor intervention is not required until the empty buffer is returned after the packet is sent. Similarly, there is no processor intervention on the reassembly side until a packet is completely received and checked for errors. There is no cell-level interaction with the processor; all processor interactions are at the packet level.

The SARA-S and SARA-R support network congestion control by means of backward explicit congestion notification. If congestion is detected in the network, notification cells may be returned by the network to all sources currently transmitting over the congested path. The SARA-R passes control information to the SARA-S for every congestion notification cell it receives. On receiving a congestion notification cell, the SARA-S will throttle the peak rate of the appropriate virtual circuit. Subsequent congestion notification cells that are received for the same virtual circuit will actuate further throttling. After a programmable time interval, the transmission rate will be gradually restored, provided no further congestion notification cells have been received for that virtual circuit.

1.2 ATM Adaptation Layer

The SARA chipset has been designed to support the segmentation and reassembly functions required to achieve packet transmission across an ATM network (or an SMDS network). In this and the following section these functions, as defined in the ANSI draft standards for B-ISDN, are briefly reviewed.

The protocol model for the variable bit rate (i.e. packet based) ATM Adaptation Layer (AAL) Type 3/4 and Type 5 as defined in the ANSI draft standards[*,†] is illustrated in Figure 1-2. The AAL is split into three sublayers. The service specific convergence sublayer (SSCS) is used to add additional service features such as assured transfer (error correction by retransmission). It may be a null sublayer for unassured transfer (error detection but no correction). The Common Part Convergence Sublayer (CPCS) performs error detection and control functions at the frame level. The Segmentation and Reassembly Sublayer (SAR) performs the functions necessary to segment frames into ATM cells and reassemble a multiplexed stream of ATM cells back into their original frames. Two AAL types have been specified to implement the functions of the CPCS and SAR sublayers: AAL 3/4 common part and AAL 5 common part. The SARA chipset supports the functions of the CPCS and the SAR sublayers for both AAL 3/4 common part and AAL 5 common part, and some functions of the ATM layer.

*. T1S1.5/92-003R2 “Broadband ISDN—ATM Adaptation Layer 3/4 Common Part Functionality and Specification” May 1992.
†. T1S1.5/92-010 “Broadband ISDN—ATM Adaptation Layer 5 Common Part Functionality and Specification” May 1992.

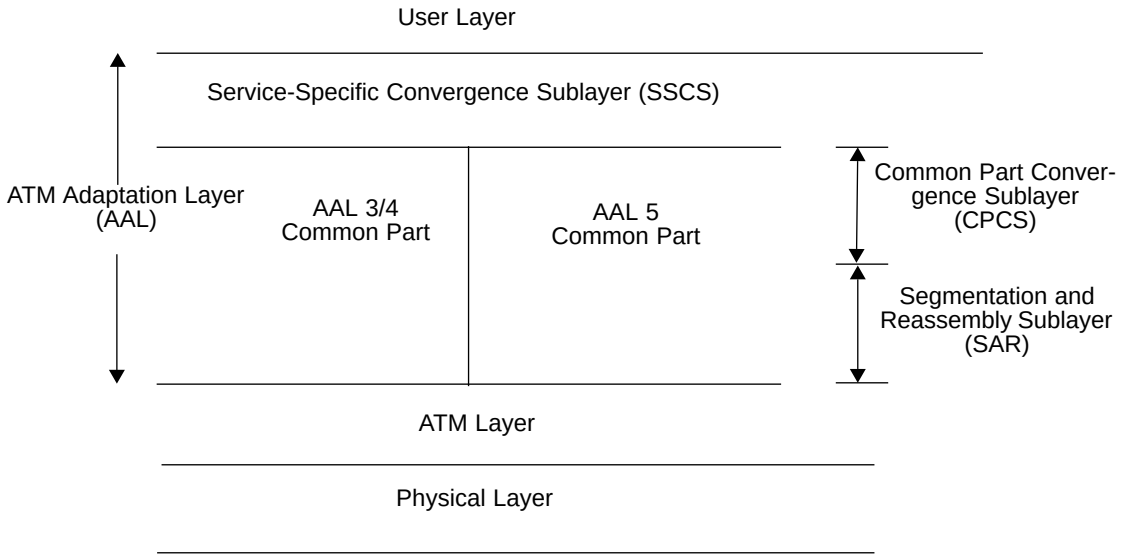


Figure 1-2. Protocol Model for Variable Bit Rate AAL Type 3/4 and Type 5

1.2.1 AAL 3/4

The structure of the CPCS-PDU for AAL3/4 is shown in Figure 1-3. This packet structure is accepted for transmission by the SARA-S and delivered by the SARA-R. The CPI field and the AL field are not used by the SARA. The BTag and ETag are used to check for certain reassembly errors. They should both contain the same eight-bit value but are not modified or checked by the SARA. The BASize field is used by the SARA-R to indicate the minimum buffer size (in bytes) that can contain the CPCS-Payload and PAD. The length field indicates the length of the payload but is not used by the SARA. The CPCS-Payload is a variable length field containing 0-65,535 bytes of CPCS-user information. A PAD field of 0 to 3 bytes is present to align the CPCS-PDU to a four-byte boundary.

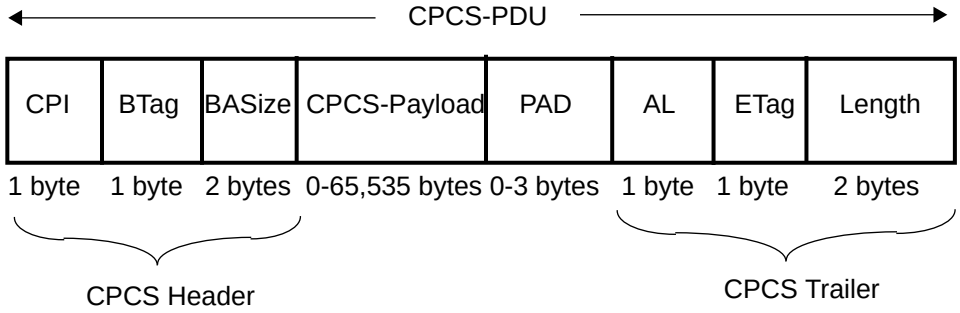


Figure 1-3. Structure of the AAL 3/4 CPCS-PDU

The SARA can optionally insert and check the CRC-32 field as shown in Figure 1-4 and Figure 1-5. For the 802.6 frame, the CRC-32 is computed on the CPCS_Payload and the pad field. The bridging field (the last two bytes of the 20-byte header in the CPCS_Payload) is considered to be zero for the CRC computation.

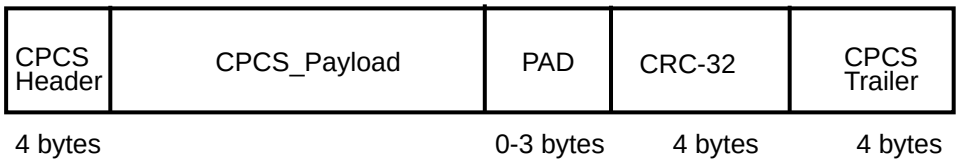


Figure 1-4. CRC-32 in an 802.6 Frame

The CRC_32 can optionally be computed over the entire CS_PDU and inserted at the end of the last cell as shown in Figure 1-5.

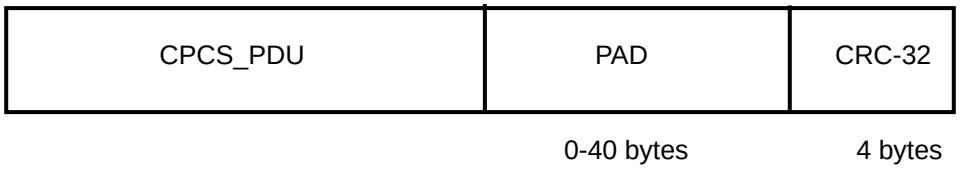


Figure 1-5. *TranSwitch* Proprietary CRC-32

The CRC-32 computation is the same as in the ANSI FDDI, 802.5, and ANSI fiber channel standards.

The segmentation process for AAL 3/4 is illustrated in Figure 1-6. The CPCS-PDU is segmented into 44-byte segments. Each segment forms the payload of an SAR-PDU. A two-byte SAR header and a two-byte SAR trailer are added to each segment to form the SAR-PDU. Each SAR-PDU then forms the 48-byte payload of an ATM cell. Each ATM cell has a five-byte cell header.

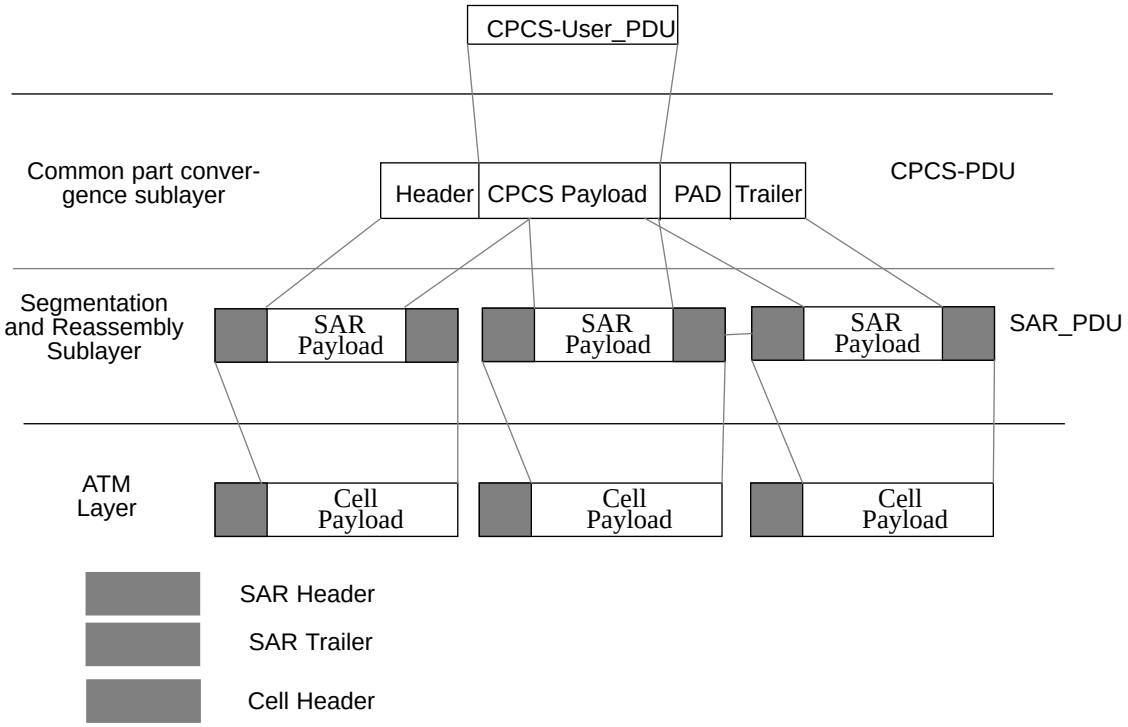


Figure 1-6. The Segmentation Process for AAL 3/4

The structure of the SAR-PDU for AAL 3/4 is shown in Figure 1-7. The segment type (ST) field indicates whether the cell is a beginning of message (BOM), continuation of message (COM), or end of message (EOM) cell, or whether it contains a complete CPCS-PDU (SSM). The sequence number (SN) is incremented modulo-16 for each successive SAR-PDU on the same virtual connection (regardless of packet boundaries). The message identification (MID) field has the same value for all SAR-PDUs derived from the same CPCS-PDU. It may be used to form part of the reassembly identifier to demultiplex and reassemble cells from multiple interleaved packets. The length indicator (LI) gives the length of the SAR-PDU payload in bytes. The CRC is a cyclic redundancy check computed over the entire SAR-PDU and is used for error detection. All of the above fields of the SAR-PDU are set, checked and handled by the SARA, according to the ANSI draft standard*, without any requirement for per cell user interaction.

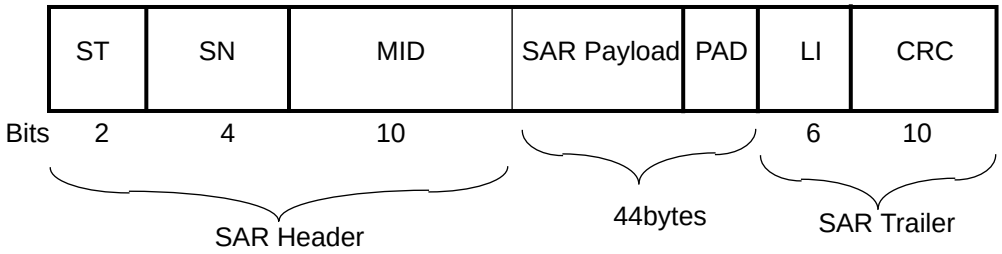


Figure 1-7. Structure of the AAL 3/4 SAR_PDU

1.2.2 AAL 5

The structure of the CPCS_PDU for AAL 5 is shown in Figure 1-8. This packet structure is accepted for transmission by the SARA-S and delivered by the SARA-R. The CPCS-payload is a variable length field containing 0 - 65,535 bytes of CPCS-user information. The PAD field is used to align the CPCS-PDU to a 48-byte boundary. The PAD field may contain any value, and it is included in the CRC-32 calculation. The control field is reserved for future use. The length field indicates the length in bytes of the payload field.

The CRC-32 field contains the result of the CRC-32 calculation performed over the entire CPCS-PDU as specified in the ANSI draft standard.† (This CRC-32 computation is the same as in the ANSI FDDI, IEEE 802.5, and the ANSI Fiber Channel standards.) The CRC-32 computation is optionally performed by the SARA-S and checked by the SARA-R.

*. T1S1.5/92-003R2 “Broadband ISDN—ATM Adaptation Layer 3/4 Common Part Functionality and Specification” May 1992.
†. T1S1.5/92-010 “Broadband ISDN—ATM Adaptation Layer 5 Common Part Functionality and Specification” May 1992.

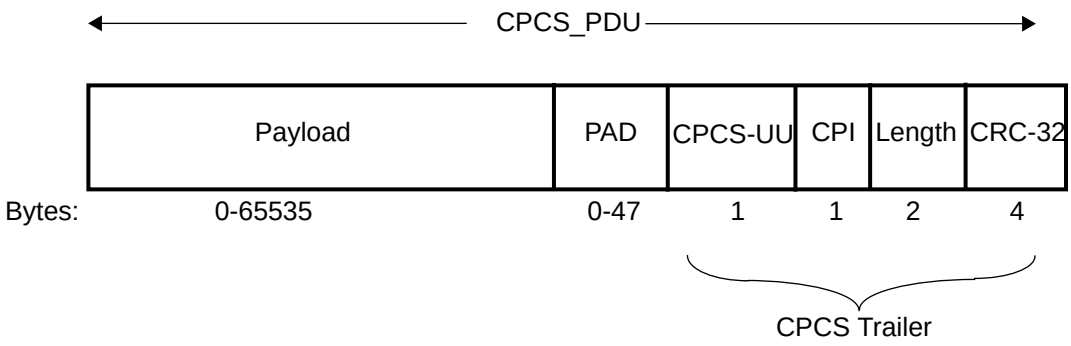


Figure 1-8. Structure of the AAL 5 CPCS_PDU

The segmentation process for AAL 5 is illustrated in Figure 1-9.

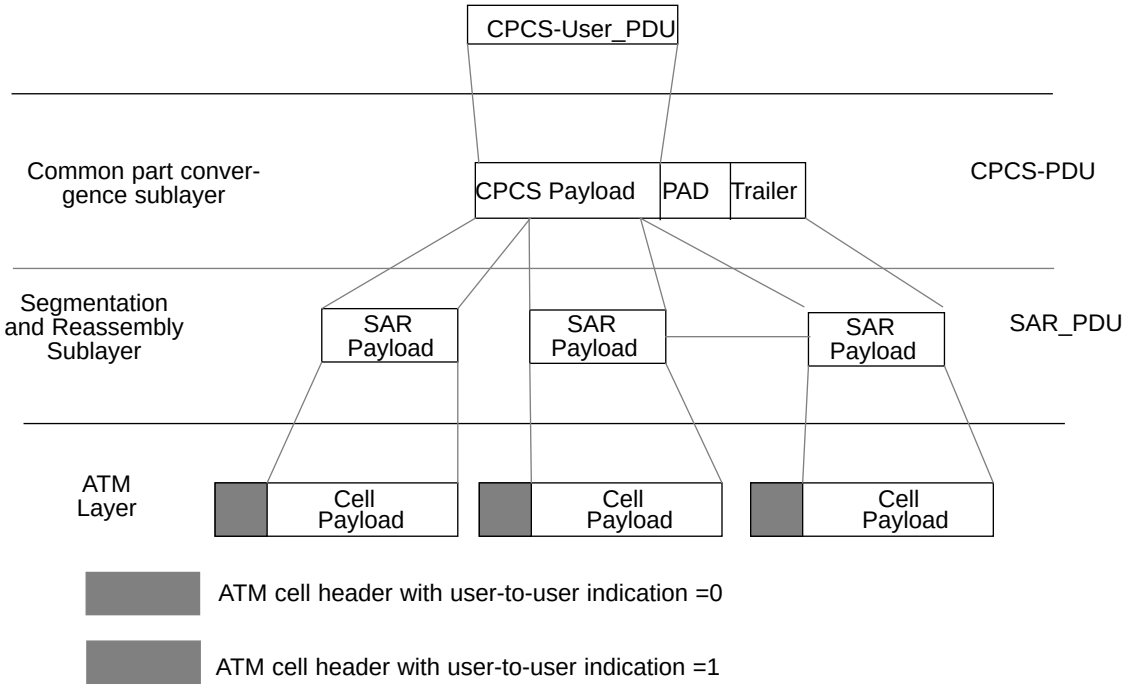


Figure 1-9. Segmentation Process for AAL 5

The CPCS_PDU is segmented into 48-byte segments, with each segment forming a SAR_-PDU. No SAR_PDUs are partially filled, since the CPCS_PDU is aligned to a 48 byte boundary. There is no SAR header or trailer on the SAR_PDU. Each SAR_PDU forms the 48-byte payload of an ATM cell.

Each ATM cell has a five-byte cell header. An ATM-user-to-user indication is available in the payload type field of the ATM cell header. AAL 5 uses an ATM user-to-user indication of “1” to indicate the last SAR_PDU of a CPCS_PDU. All other SAR_PDUs are carried in ATM cells

with an ATM user-to-user indication of ‘0’. The SARA-R can reassemble ATM cells carrying AAL 5 SAR_PDUs into AAL 5 CPCS_PDUs without any requirement for per-cell user interaction. All possible reassembly errors are checked and reported by the SARA-R.

1.3 The ATM Layer

The format of the ATM cell at the user network interface (UNI) of an ATM network as defined by the CCITT for the broadband ISDN* is shown in Figure 1-10. The generic flow control (GFC) field is used to ensure fair and efficient access between multiple devices sharing a single UNI. A label space of 24 bits is provided, divided into two fields: an eight-bit virtual path identifier (VPI), and a 16-bit virtual channel identifier (VCI)†. The VPI allows a group of virtual connections, called a virtual path, to be identified and the VCI identifies the individual virtual connections within each virtual path. The payload type (PT) field is used to distinguish user information and network information. For user information cells, the payload type field carries a single bit ATM user-to-user identification. ATM cells with a payload type field that indicates network information may be inserted by the SARA-S and filtered out by the SARA-R. The cell loss priority (CLP) bit permits two priorities of cell to be defined where the network may discard low priority cells under congestion conditions. The header error check (HEC) field provides an eight-bit cyclic redundancy check on the contents of the cell header.

The SARA-S maintains a connection table with a copy of the four ATM cell header bytes for each virtual connection (the HEC field is computed for each cell and therefore is not stored in the connection table). The SARA-S can thus insert any pattern of bits into the cell header, and may therefore originate cells with any VPI or VCI. The SARA-R performs cell to packet reassembly using up to 16 bits from the VCI field optionally combined with up to ten bits from the MID field of the SAR_PDU, or eight bits from the VPI field combined with up to ten bits from the VCI field.

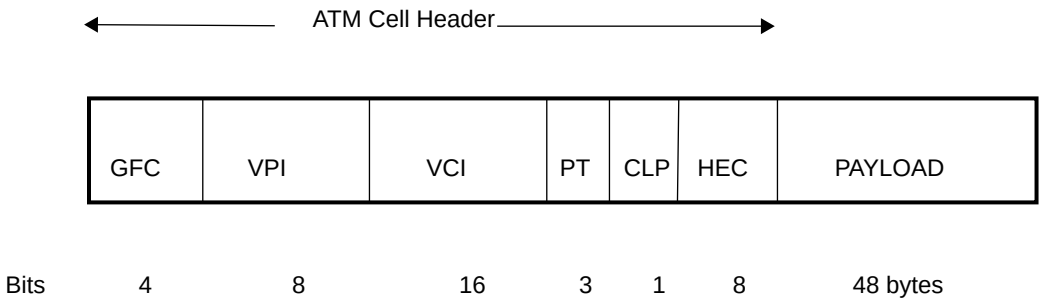


Figure 1-10. Structure of an ATM Cell at the User Network Interface

*. CCITT Recommendations I.150, I.361 (Geneva, June 1992).

†. Within the telecommunications industry, the terms “virtual connection”, “virtual channel”, and “virtual circuit” tend to be used as synonyms. In this document the more general term “virtual circuit” has been adopted.

Chapter 2. SARA Functional Description

This chapter highlights the main features of the SARA chipset. The SARA data structures and operation are also described briefly.

2.1 Major SARA Features

The major features of the SARA chipset are as follows:

- Support for B-ISDN standard ATM Adaptation Layer (AAL) 3/4 and AAL 5 Common Parts.
- Selectable packet-level CRC generation and checking on a per connection basis.
- B-ISDN cell payload CRC generation and checking (AAL 3/4).
- B-ISDN cell header CRC generation and checking (with no correction).
- Programmable support of between 512 to 65,536 virtual circuits.
- Support of up to 8,191 buffers in each direction, for simultaneous segmentation and reassembly of up to 16,382 packets.
- Management of constant bit rate traffic.
- Management of OAM cells.
- Management of host interface queues.
- Availability of either synchronous or asynchronous memory interface handshake modes.
- Selectable 16- or 32-bit packet memory interface width.
- 16-bit control memory interface.
- DMA controllers for packet and control memory.
- Simultaneous support for CBR, AAL 3/4, and AAL 5 traffic.
- Selectable eight-bit or 16-bit network cell interface width.
- Optional parity checking on control memory, packet memory and cell interfaces on an individual basis.
- Peak rate metering of cells for each virtual circuit with up to eight programmable peak rates.
- Average metering of cells on each virtual circuit.
- Queueing of multiple frames for each virtual circuit.
- Automatic throttling of virtual circuits upon receipt of backward congestion notification.
- Reassembly of 53-byte cells into packets in programmable-size large or small buffers (AAL 3/4).
- Automatic packet aging and buffer recovery using a packet timeout counter.
- Management of cell and packet exceptions.

2.2 Segmentation SARA Data Structures

Figure 2-1 shows the basic data structures for the Segmentation SARA (SARA-S).

The data structures contained in packet memory are the packet buffers and optionally the constant bit rate (CBR) traffic queue. These data structures contain the data to be transmitted. The packet memory is accessed through the packet memory interface.

The data structures in control memory are the buffer descriptor table, the virtual circuit table and the communication queues. These data structures contain the overhead information and other segmentation variables. The control memory is accessed through the control memory interface.

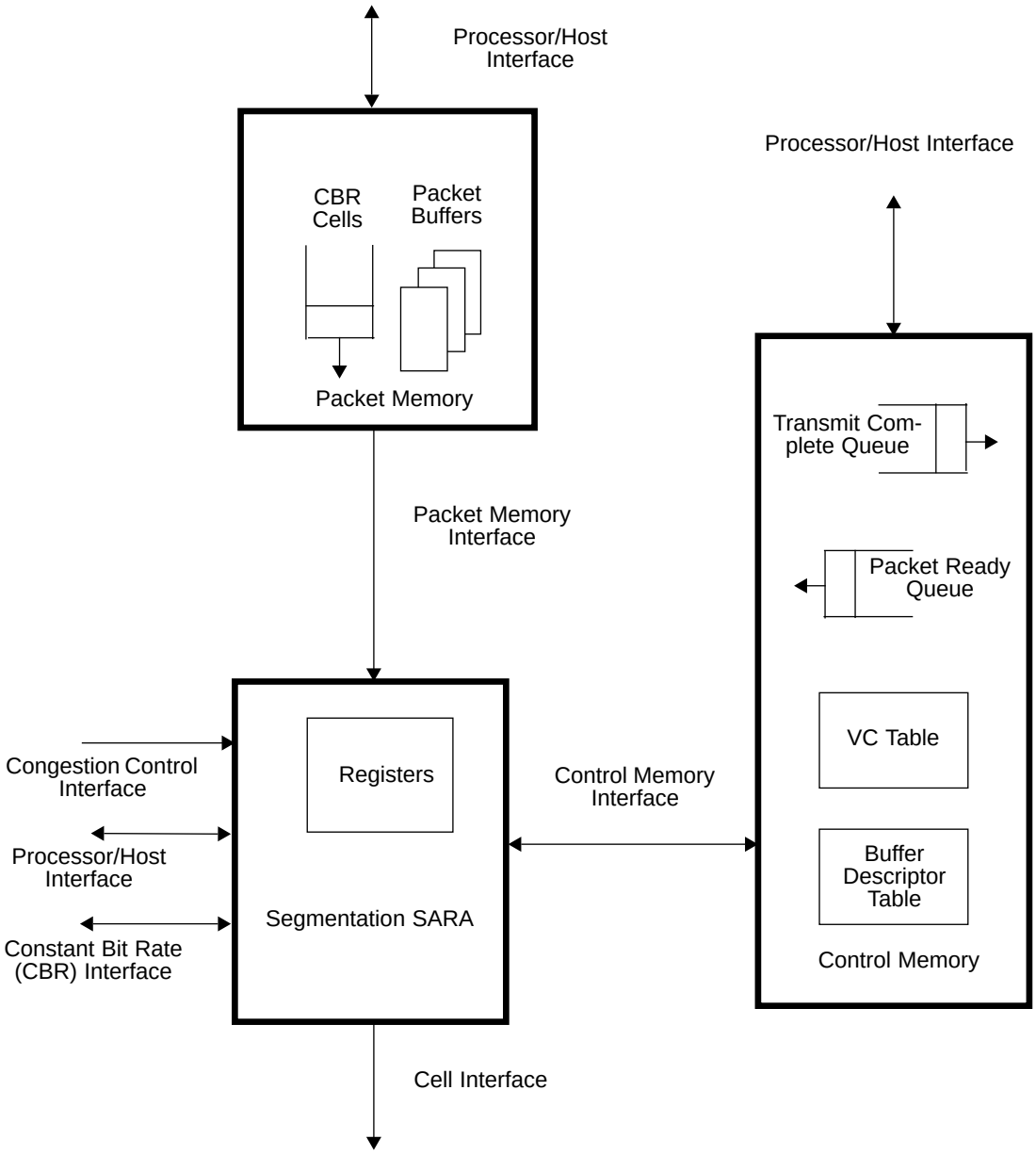


Figure 2-1. Segmentation SARA Data Structures

The internal registers in SARA-S determine the device operation modes. They also contain the base addresses of the buffer descriptor, the virtual circuit tables, the various addresses associated with the communication queues, and the rate queue registers that determine the peak segmentation rate of each rate queue. These registers are accessed using the processor interface.

The packet buffers, the CBR queue, the buffer descriptor table, the virtual circuit (VC) table and the two communication queues - packet ready queue and transmit complete queue - constitute the basic SARA-S data structure elements.

Each packet buffer contains either a packet ready to be segmented or one being segmented. The CBR traffic is passed to SARA-S through a FIFO whose address is mapped into the packet memory address space.

The buffer descriptor table contains packet-specific information for each packet to be segmented. Each table entry includes the virtual circuit identifier (VCI) associated with the packet, the length of the packet to be segmented, the location of the buffer in the packet memory and other temporary variables. SARA-S supports up to 8,191 descriptors, all of which can be active simultaneously for segmenting packets on different virtual circuits.

The virtual circuit (VC) table contains the ATM header information, congestion control information, and the peak and average rate metering information associated with each virtual circuit. SARA-S supports between 512 and 65,536 virtual circuit entries in the VC table. The table entries are setup by software as virtual circuits are established.

The communication queues—the packet ready queue and the transmit complete queue—are used to pass descriptor numbers between SARA-S and the processor. The queue locations are specified by the common queue base address along with the respective start and end addresses.

The packet ready queue contains descriptor numbers of those buffer descriptors that are ready to be linked for segmentation by SARA-S. The transmit complete queue contains descriptor numbers and completion codes for buffer descriptors that have been segmented and delinked by SARA-S.

2.3 SARA-S Operation

Packets to be segmented by SARA-S are placed in a packet buffer associated with a free buffer descriptor by the processor. The packet segmentation process is then initiated by writing the buffer descriptor number into the packet ready queue. SARA-S then reads the descriptor number from the packet ready queue and links the descriptor to one of the eight internal rate queues as specified by the rate queue field in the corresponding VC table. Upon linking, the segmentation of the packet occurs based on the parameters in the buffer descriptor table and its VC table. SARA-S accesses these packets in memory through the packet memory interface to segment them into 44/48-byte ATM cell payloads. As a packet is segmented into cells, the ATM header, checksum and the ATM adaptation layer header (for AAL 3/4) information is added to the front of each cell. The ATM adaptation layer trailer (for AAL 3/4) is appended to the end of the cells as they are transferred to the link via the cell interface.

The packet segmentation rate is determined by the rate queue parameters programmed into the SARA-S rate queue registers and the average rate metering parameters in the VC table. If SARA-S is notified of congestion in the path, the segmentation rate is throttled on specific virtual circuits. SARA-S is notified of congestion through the congestion control interface used by SARA-R to transfer the congestion control information.

SARA-S also reacts to flow control signals (XON/XOFF) intelligently. Packet segmentation on programmed rate queues will be suspended on receipt of an external XOFF condition. The transfer of constant bit rate traffic can also be suspended during XOFF by setting a bit in a programmable register. Normal segmentation will resume after the XOFF condition goes away.

The interfaces and programming details are described in greater detail in subsequent chapters.

2.3.1 SARA-S Rate Queues

Every virtual circuit is associated with a rate queue that determines the peak rate at which the frames are segmented. There are eight rate queues, which are organized into two banks of four rate queues each. These are designated as high priority bank A and low priority bank B. Each rate queue has a rate queue register that sets the rate queue characteristics. The rate queues and registers are discussed further in Chapter 4.

Rate queues within each bank are serviced in a sequential or “round robin” manner. The value programmed into the rate queue register determines the peak rate at which the rate queue is serviced. The rate queue data structure is two-dimensional. All descriptors associated with the same VC are linked horizontally, one behind the other. All descriptors associated with different virtual circuits (VCs) are linked one above the other, or vertically. When the rate queue is serviced, one cell from each descriptor (packet) in the vertical chain is transferred to the link interface. With this scheme, packets to different virtual circuits on the same rate queue are segmented by one cell every time the rate queue is serviced. Correspondingly, every packet on the same virtual circuit is completely segmented before segmentation of the next packet is commenced, maintaining an ordered segmentation of packets on the same virtual circuit.

Figure 2-2 shows a rate queue data structure with descriptors linked to three different virtual circuits. Buffer descriptors A1, A2 and A3 are associated with virtual circuit VC1; B1 is associated with virtual circuit VC2; C1 and C2 are associated with virtual circuit VC3. Every time the rate queue is serviced, one cell from each VC is transferred to the link interface. In this case, one cell from the packet indicated by descriptor A1, one cell from B1 and one cell from C1 are transferred. When A1 is completely transferred, it is delinked and A2 takes its place. Likewise C2 takes the place of C1 when C1 is completely segmented and delinked from the rate queue. Meanwhile if a new descriptor C3 (not shown) on virtual circuit VC3 is queued, it will be linked after C2 so that segmentation of C3 will begin only after segmentation of C2 has completed.

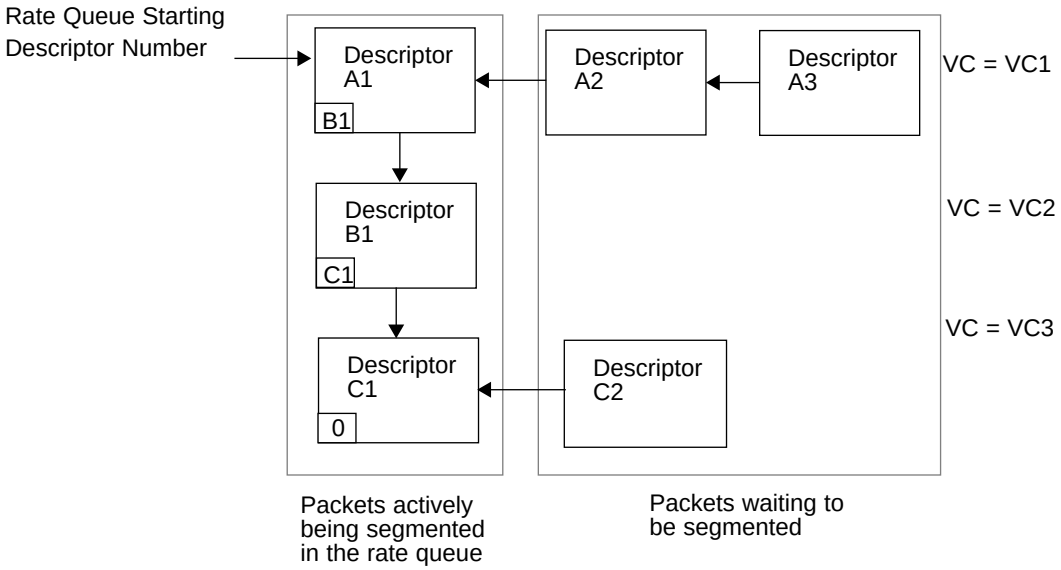


Figure 2-2. Typical Rate Queue Data Structure

2.3.2 Peak and Average Metering

A number of virtual circuits emanate from a particular source. Each virtual circuit can be allocated a peak rate—the maximum rate at which cells will be transmitted on that circuit when there is traffic to send. For some network applications, the peak rate must be controllable in order to both avoid congestion in the network and to avoid too many cells being transmitted beyond the receiver’s capability. To smooth the traffic, the SARA chipset allows peak rates to be defined associated with the eight rate queues. Any virtual circuit can be a member of any one of those rate queues. If the sum of the active rate queues exceeds the transmission bandwidth available, only the higher priority queues will be served. Peak rates define the time between emission of cells on a virtual circuit. The peak rate is determined by the value programmed in the rate queue registers.

The traffic on each virtual circuit will be bursty, i.e. there will be times when there is no traffic to send. Average metering smooths traffic to avoid congestion, assists in statistical multiplexing, and places an upper bound on the average amount of traffic carried across each virtual circuit. Average metering may be disabled on any virtual circuit, which causes the cells to be transmitted at the peak rate.

The mechanism used to perform average metering is a “leaky bucket” algorithm. For each virtual circuit, credits accumulate up to a pre-programmed maximum value at the average cell transmission rate. Each time a cell is sent, a credit is removed until the packet(s) are segmented or no credits are left. If credit is exhausted, it is replenished at the rate of the average transmission; cells are sent at this rate if data is available to send. As long as credits exist, cells are sent at the peak rate.

If there has been a gap in the data to send, credits will accumulate up to the maximum. The maximum credit thus determines the maximum burst length. As long as each VC stays within the designated average, cells are transferred at the peak rate (which is also the burst rate). When a burst of transmitted cells on a VC exceeds the burst length specified for that VC, the VC is considered oversubscribed forcing the cells to be segmented and transferred at the average rate. Refer to Chapter 6 for more details.

2.3.3 Congestion Control

Some networks may provide a *backward explicit congestion notification* scheme for congestion control. When congestion is experienced in the network, the network may issue congestion notification cells. When SARA-R receives a congestion notification cell, it passes the congestion information to the SARA-S which will automatically throttle the specified virtual circuit. The SARA-R may also pass these cells to the packet memory so that the processor/host can process them if needed.The interface between SARA-R and SARA-S is the serial congestion control interface.

The congestion notification cell contains explicit information which indicates the type of congestion. SARA-S recognizes two types of congestion information: moderate and extreme.

When moderate congestion notification is received on a virtual circuit, SARA-S throttles the segmentation peak rate on that virtual circuit by one “notch.” Each notch reduces the peak rate on the circuit by one half of the rate of the previous notch. The fifth notch stops segmentation on the VC. When no congestion notification cells arrive for a throttled VC during a timeout period, the SARA-S will increase the segmentation peak rate to the previous notch. If no further congestion cells are received, this process will continue until the VC is operating at its programmed full peak rate. The VC mode bits in the SARA-S VC Table (Chapter 6) define the timeout period.

When extreme congestion information is received, SARA-S stops segmenting and transmitting additional cells on that particular VC (drops down to the fifth notch) and ramps back up as described above.

2.4 Reassembly SARA Data Structures

Figure 2-3 shows the basic data structures for the Reassembly SARA (SARA-R).

The packet memory contains packet buffers, the raw cells, and, optionally, a constant bit rate (CBR) traffic queue. These data structures contain the data that has been received. The packet memory is accessed through the packet memory interface.

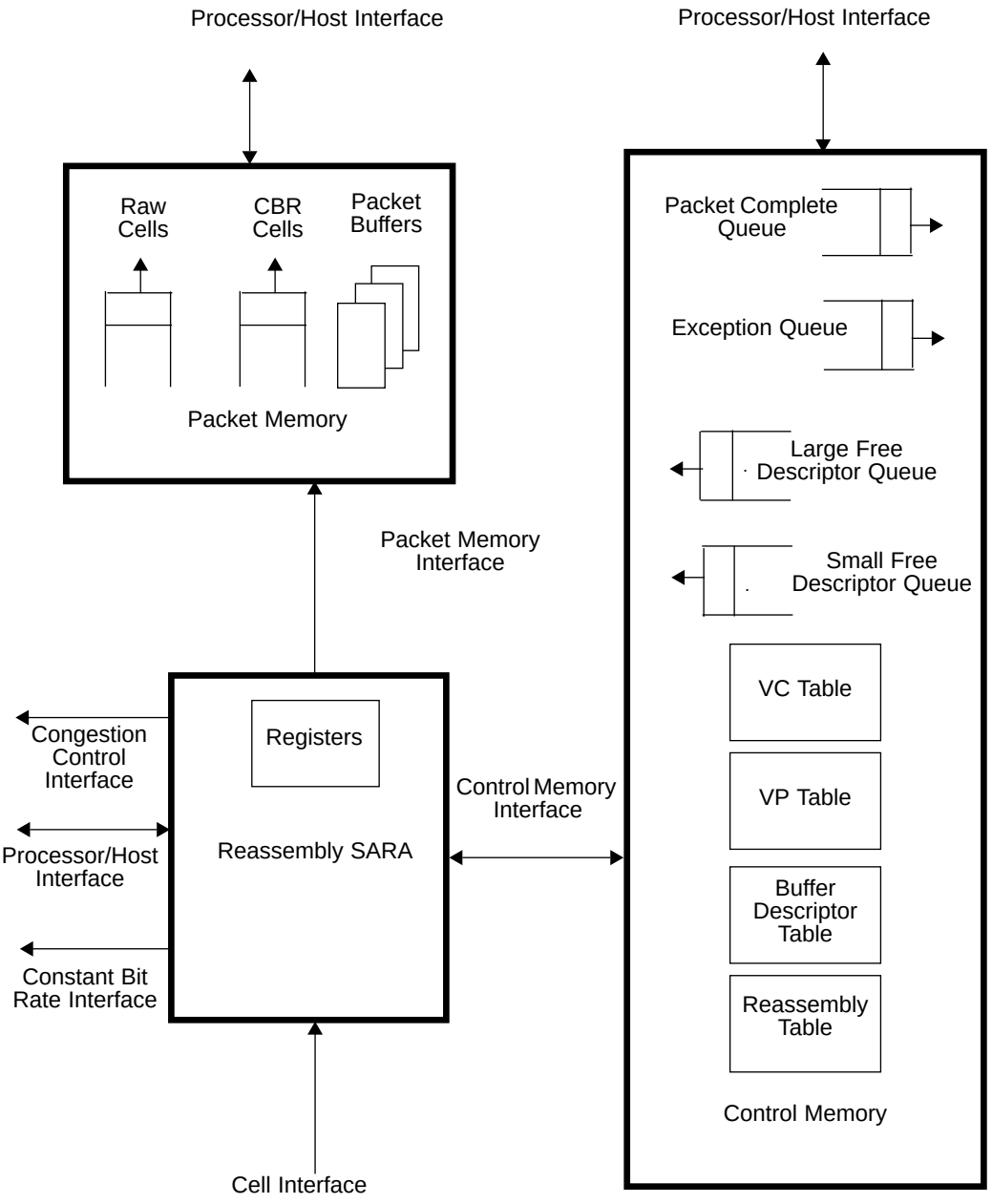


Figure 2-3. SARA-R Data Structures

The control memory contains the buffer descriptor table, virtual circuit table, virtual path table, reassembly table, and the communication queues. These data structures contain the overhead information and other reassembly variables. The control memory is accessed through the control memory interface.

The internal registers in SARA-R determine the device operation mode. They also contain the base addresses of the buffer descriptor, virtual circuit, virtual path, and reassembly tables along with the various addresses associated with the communication queues. These registers are accessed through the processor interface.

Each packet buffer contains a packet either that has been reassembled or is being reassembled. The CBR queue address is mapped into the packet memory address. The received CBR cells are written to this programmable-length queue. The raw cell queue is mapped into the packet memory address space. The OAM F5 cells, congestion notification cells, and cells on virtual circuits marked for the raw cell queue are written into this raw cell queue.

The buffer descriptor table provides packet-level information for each packet that is undergoing or has completed reassembly. Each table entry includes the packet status and errors, the length of the received packet, the location of the buffer in the packet memory and other temporary variables. SARA-R supports up to 8,191 descriptors, all of which can be used simultaneously for reassembling packets.

The virtual circuit (VC) table provides the status of each virtual circuit and the associated reassembly pointer of any packet that is in the process of reassembly. The SARA-R supports between 512 and 65,536 virtual circuit entries.

The virtual path (VP) table provides the status of each virtual path and the associated reassembly pointer of any packet that is in the process of reassembly. The SARA-R supports 256 virtual path entries.

The reassembly table provides the descriptor number associated with each packet that is in the process of reassembly. It also provides the state and type of packet being reassembled, i.e. AAL 3/4, AAL 5, raw cell or CBR.

The communication queues—the small free descriptor queue, the large free descriptor queue, the packet complete queue, and the exception queue—are used to pass descriptor numbers between the SARA-R and the processor. The queue locations are specified by the common queue base address, along with the respective start and end addresses.

The small and large free descriptor queues hold the descriptor numbers that are available to the SARA-R for reassembling packets. The small free descriptor queue contains descriptor numbers for the small buffers in the packet memory and the large free descriptor queue contains descriptor numbers for the large buffers in the packet memory. The two sizes of the buffers are determined by software and programmed in the SARA-R internal registers.

The packet complete queue is used by the SARA-R to pass descriptor numbers of packets that have been completely received or were terminated due to error conditions. The software processes the descriptors from the packet complete queue and places them on the small or large descriptor queue when done.

The exception queue is used by the SARA-R to report error and exception conditions when cells are received on VCs that currently do not have an associated reassembly buffer descriptor.

2.5 SARA-R Operation

SARA-R receives cells from the cell interface. When the first cell of any packet is received, SARA-R determines the size of the packet from its contents (for AAL 3/4 only), and selects a buffer descriptor number from either the large free descriptor queue or the small free descriptor queue to store the reassembled packet. The reassembly table entry is updated to associate the descriptor number for subsequent cells received from that virtual circuit. The ATM header, the header checksum and the ATM adaptation layer information is stripped from the cells before storing them in the packet buffer (except in the case of CBR traffic and raw cells, where the ATM header is also passed to the external buffer). Subsequent cells are processed and stored in the appropriate packet buffer until the last cell of the packet is received. At that time, SARA-R updates the descriptor status information and passes the descriptor to software by writing its number into the packet complete queue. In certain cases when a cell is received and cannot be associated with any descriptor, the virtual circuit number and an error code are written to the exception queue.

The interfaces and programming details are described in greater detail in subsequent chapters.

2.5.1 Exception Handling

During the reassembly process certain types of errors will trigger the SARA-R exception handling mechanism. The exceptions or errors handled by the SARA-R can be divided into the following three categories and are explained in greater detail in Chapter 5.

- *Errors in the Cell Syntax:* These include bad header CRC, bad payload CRC and incorrect cell size based on the cell type (AAL 3/4). When such errors are encountered, the cell is dumped and the error-counter is incremented. Based on the type of the error, if a packet is active, it may be terminated.
- *Errors in the Received Packet:* packet errors include, end-of-frame not received before the beginning of another frame was received (AAL 3/4), packets not completed within a pre-determined amount of time and receive buffer overflow. If a packet is active, it may be terminated.
- *Errors that Cannot be Associated with Packets:* error condition such as the reception of a COM (Continuation-of-Message) or EOM (End-of Message) cell with no preceding BOM (Beginning-of-Message) cell (AAL 3/4), cells received on invalid VPs or VCs, and packets lost due to buffer descriptors not being available. These errors are reported through the exception queue.

2.5.2 Packet Timeout

The Reassembly SARA implements a buffer recovery mechanism for packets that remain active for a long time and are not completely received. This is referred to as the packet-timeout or packet-aging and is described in greater detail in Chapter 5.

When the first cell of a packet is received, the packet timeout count value in the buffer descriptor assigned to this packet is initialized with a value from an internal timeout count register. Whenever an internal (programmable) timer overflows, the SARA-R increments a timeout index. The SARA-R examines the state of the reassembly data structure pointed to by the timeout index. If the reassembly structure is active, the SARA-R increments the packet timeout count in the buffer descriptor associated with that reassembly structure. When the packet timeout count value in the buffer descriptor overflows, the reassembly process on that packet is terminated and the descriptor is placed on the packet complete queue with the appropriate error condition. The state of the corresponding location in the reassembly table will also be set to inactive.

Chapter 3. Hardware Description

3.1 Segmentation SARA Hardware Description

3.1.1 Segmentation SARA Internal Block Description

Figure 3-1 shows a block diagram of the Segmentation SARA chip.

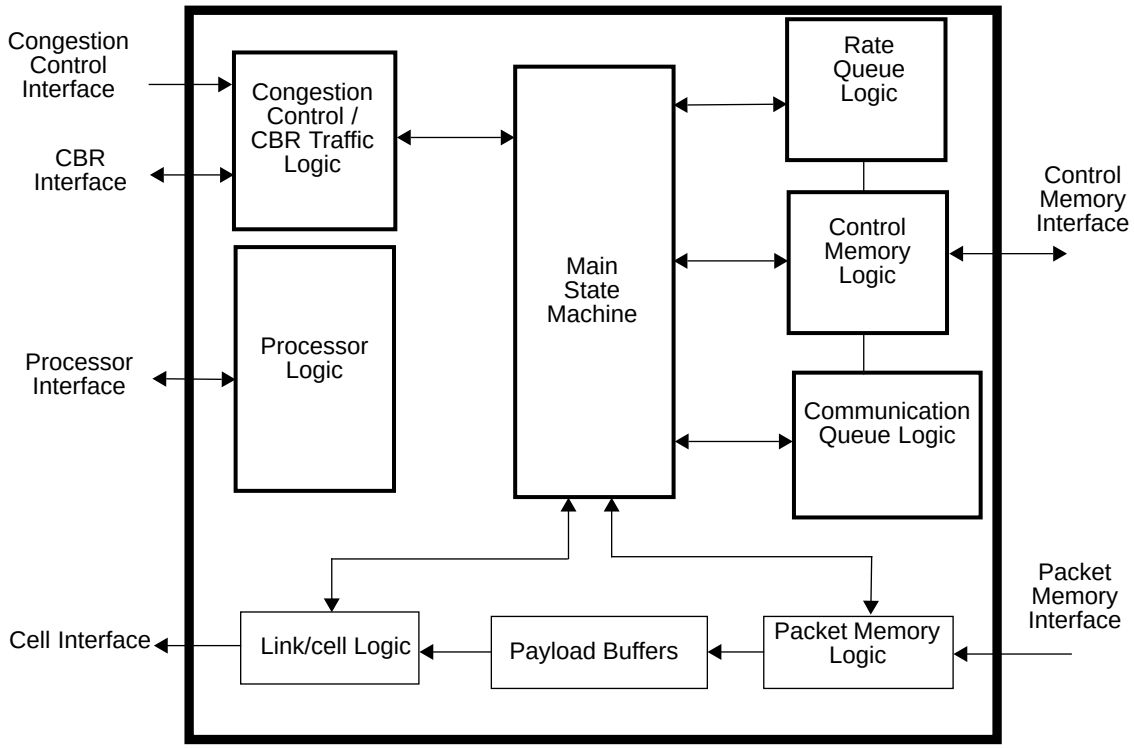


Figure 3-1. Segmentation SARA Block Diagram

The SARA-S internally consists of several functional blocks:

- The *Processor Logic* block interfaces with the processor to access the internal registers of SARA-S and to setup the device. The processor interface is an asynchronous interface.
- The *Control Memory Logic* block generates the bus arbitration signals (to access the control memory) and data transfer signals for the control memory. It can be programmed to generate these signals for both synchronous and asynchronous memory interfaces.
- The *Packet Memory Logic* block generates the bus arbitration signals (to access the packet memory) and data transfer signals for the packet memory. It generates the packet memory address and transfers data from the packet memory to the payload buffer. It can be programmed to generate these signals for both synchronous and asynchronous memory interfaces.
- The *Congestion Control /CBR Traffic Logic* block is used to communicate with the

SARA-R to receive congestion control information and also to communicate with the source generating the constant bit rate traffic.

- The *Cell/Link Logic* is used for transferring ATM cells to the external logic. This block generates the necessary handshake signals needed to interface to a FIFO-like device.
- The *Payload Buffer* is an internal buffer (2 cells deep) that holds the payload ready for transfer to the cell interface.
- The *Rate Queue Logic* contains the rate queue registers and counters. It also contains the priority logic to determine the rate queue to service.
- The *Communications Queue Logic* contains the pointers and queue control logic necessary to pass descriptors between the host and the SARA-S.
- The *Main State Machine* comprises 95 states and controls all internal blocks except the processor interface. It generates all the control signals in proper sequence for correct device operation.

3.1.2 Segmentation SARA Hardware Interfaces

The SARA-S contains the following hardware interfaces:

- Processor Interface
- Cell Interface
- Control Memory Interface
- Packet Memory Interface
- Congestion Control Interface
- Constant Bit Rate Traffic Interface
- Miscellaneous Interface

Notes for Tables:

1. Bit 0 on all busses is the least significant bit.
2. Parity is optional on all data busses that have parity bits.
3. An * at the end of signal name indicates that the signal is an active-low signal.
4. OD indicates an open drain output.
5. POD indicates a programmable open drain output.
6. All input and bi-directional I/O pins have internal pull-ups or pull-downs, which force an inactive state when the pin is left open.
7. All unused input pins must be tied to inactive state.

3.1.2.1 Processor Interface

The processor interface is a slave interface that is used by the local host/processor to read and write the internal registers of the device. These registers are identified in Chapter 4. It is also used to interrupt the processor/host when an unmasked status bit is set. Table 3-1 shows the processor interface pin descriptions.

Symbol	Type	Name and Function
CS*	I	Chip select signal to enable the processor interface for data transfers
DS*	I	Data strobe for enveloping the data transfer
WRT	I	WRITE(1), READ(0) input to indicate the direction of data transfer

Table 3-1. SARA-S Processor Interface Pin Descriptions

Symbol	Type	Name and Function
RDY*	OD	SARA-S signal indicating completion of data transfer
A(7:0)	I	Address bus for internal register selection
D(15:0)	I/O	Processor interface data bus
INTR*	OD	Interrupt signal to the processor

Table 3-1. SARA-S Processor Interface Pin Descriptions

3.1.2.2 Cell Interface

The cell interface is where the transmit ATM cell stream exits the SARA-S. It may be configured to be either eight- or 16-bits wide. Table 3-2 shows the SARA-S cell interface pin descriptions.

Symbol	Type	Name and Function
RDCLK	I	Read clock (free-running) used for operating the cell interface logic.
CELAVL*	O	Cell available signal indicating that an ATM cell is available for transfer (read-out) on the FFD pins.
RDEN	I	Read enable for cell transfer on the FFD bus. The next data word is transferred on to the FFD bus on a low to high transition of RDCLK when CELAVL* and RDEN are active.
FFD(17:0)	O	Cell data output bus. FFD(17) and FFD(16) are the parity bits for FFD(15:8) and FFD(7:0) respectively. Data is driven on FFD(7:0) when this interface is configured in the 8-bit mode and uses FFD(16) as parity.
XON	I	XON input. When low, cell generation on specific rate queues and optionally the CBR traffic is suspended. Cell generation on all queues is enabled when this signal is high.

Table 3-2. SARA-S Cell Interface Pin Descriptions

3.1.2.3 Control Memory Interface

The control memory interface is a bus-master interface to the control memory. It is used to access the communication queues, various data structures, including the buffer descriptors, the prefix and header tables maintained in the control memory. Table 3-3 shows the SARA-S control memory interface pin descriptions.

Symbol	Type	Name and Function
CREQ	O	Control memory bus request.
CGRT	I	Control memory bus grant.

Table 3-3. SARA-S Control Memory Interface Pin Descriptions

Symbol	Type	Name and Function
CMULR	O	Control memory multiple request signal, indicating that more than one data transfer is pending.
CCYCST*	POD	Control memory cycle start signal marking the beginning of data transfer. Programmable as either open drain or as an active output.
CRDY*	I	Control memory ready signal validates each data transfer when CCYCST* is active
CWRT	O	Control memory write signal for direction of transfer. This pin can be programmed as an early write signal to provide additional flexibility in external system design.
CA(23:1)	O	Control memory address bus (16-bit word address).
CD(17:0)	I/O	Control memory data bus. CD(17) & CD(16) are the parity bits for CD(15:8) & CD (7:0) respectively.

Table 3-3. SARA-S Control Memory Interface Pin Descriptions

3.1.2.4 Packet Memory Interface

The packet memory interface is a bus-master interface for reading packet data from packet memory to generate ATM cells. Various bus interfaces are supported.

Table 3-4 shows the SARA-S packet memory interface pin descriptions.

Symbol	Type	Name and Function
PREQ	O	Packet memory bus request
PGRT	I	Packet memory bus grant
PCYCST*	POD	Packet memory cycle start (marks the beginning of data transfers). Programmable as either open drain or as an active output.
PRDY*	I	Packet memory ready, validates each data transfer when PCYCST* is active.
PD(35:32)	I	Input only for packet memory data. PD(35), PD(34), PD(33), PD(32) are the parity bits for PD(31:24), PD(23:16), PD(15:8), PD(7:0) respectively.
PD(31:0)	I/O	Input for data; output for multiplexed address in PM_-REQADR mode (Byte address PA(31:0) is presented on these pins during packet memory bus requests).
PA(15:0)	O	Packet memory address bus. This bus is driven with 16-bits of the byte address of the packet memory location being accessed. The selection of the 16-bits is dependent upon the state of the PLWADR pin and the PM_INTRLV mode bit.

Table 3-4. SARA-S Packet Memory Interface Pin Descriptions

Symbol	Type	Name and Function
PLWADR	I	Input signal to multiplex the upper or lower part of the packet memory address on the PA bus.
PAMTCH	O	Indicates that the packet memory address matches with the contents of the address match register. Used to detect the end of the serial address memory (for VRAMs) or page boundaries.

Table 3-4. SARA-S Packet Memory Interface Pin Descriptions

3.1.2.5 Congestion Control Interface

Table 3-5 shows the SARA-S congestion control interface pin descriptions. This interface is used by the SARA-R to transfer congestion information to the SARA-S to throttle the source virtual circuit segmentation rate.

Symbol	Type	Name and Function
CCDATA	I	Serial input stream of congestion control information from SARA-R to SARA-S.
CCXFER	I	Congestion control transfer enable for CCDATA.
CCHLD	O	Hold congestion control transfer.

Table 3-5. SARA-S Congestion Control Interface Pin Descriptions

3.1.2.6 Constant Bit Rate Traffic Interface

This interface is used to communicate with a constant bit rate (CBR) traffic source for transmitting CBR cells.

Table 3-6 shows the SARA-S constant bit rate interface pin descriptions.

Symbol	Type	Name and Function
CBRXMIT	I	CBR cell transmit signal to the SARA-S. This signal may be de-asserted one clock cycle after the CBRDONE signal is driven low.
CBRDONE	O	CBR cell transmit done signal to complete the handshake with CBRXMIT indicating the successful transfer of a CBR cell to the cell interface.

Table 3-6. SARA-S Constant Bit Rate Transfer Interface Pin Description

3.1.2.7 Miscellaneous Signals

Table 3-7 shows the SARA-S pin descriptions for miscellaneous signals. The test pins TE, TEI, and SI are provided for manufacturing test purposes only. They must be tied to “0” (low) in user applications.

Symbol	Type	Name and Function
RST*	I	Hardware reset for SARA-S
CLK	I	Clock input.
TE	I	Scan test enable input.
SI	I	Scan input for SARA-S in scan mode.
SO	O	Scan output for SARA-S in scan mode.
TEI	I	Tri-state all outputs and bidirectional pins

Table 3-7. SARA-S Miscellaneous Signals Pin Description

3.2 Reassembly SARA Hardware Description

3.2.1 Reassembly SARA Internal Block Description

Figure 3-2 shows a block diagram of the Reassembly SARA (SARA-R) chip.

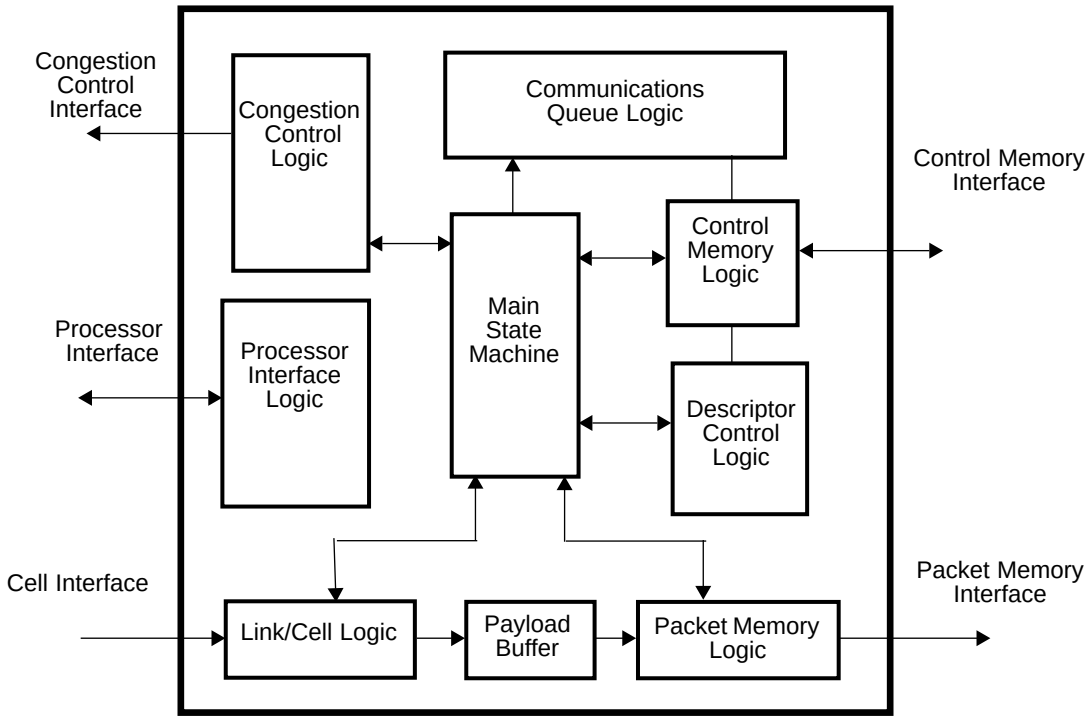


Figure 3-2. Block Diagram of Reassembly SARA

The SARA-R internally consists of several functional blocks:

- The *Processor Interface Logic* block interfaces with the processor to access the internal registers of SARA-R and to issue commands. The processor interface operates asynchronously.
- The *Control Memory Logic* block generates the bus arbitration signals (for access to the packet memory) and data transfer signals to the control memory. It can be programmed to generate these signals for both synchronous and asynchronous memory interfaces.
- The *Packet Memory Logic* block generates the bus arbitration signals (for access to the control memory) and data transfer signals to the packet memory. It also generates the packet memory address and transfers data from the cell buffer to the packet memory. It can be programmed to generate these signals for both synchronous and asynchronous memory interfaces.
- The *Link/Cell Logic* block is used to read the ATM cells from the link. It generates the signals to interface to a FIFO-like device.
- The *Payload Buffer* is an internal buffer (1 cell deep) that holds the payload ready for transfer to the packet memory interface.
- The *Descriptor Control Logic* block maintains working registers and processing logic for the buffer descriptor manipulation. It maintains the various base address registers and generates the control memory address and data bus signals. It also contains a timer for aging packets.

- The *Congestion Control Logic* block is used to communicate with the SARA-S to pass congestion control information.
- The *Communications Queue Logic* block contains all of the pointers and associated pointer logic for the four communication queues in the control memory and the two cell FIFOs in the packet memory. The pointers can be accessed by the host through the processor/host interface.
- The *Main State Machine*, consisting of 64 states, controls the seven blocks described above. It generates all of the control signals and sequences that operate the device.

3.2.2 Reassembly SARA Hardware Interfaces

The SARA-R contains the following types of hardware interfaces

- Processor Interface
- Cell Interface
- Control Memory Interface
- Packet Memory Interface
- Congestion Control Interface
- Miscellaneous Interface

Notes for tables:

1. Bit 0 on all busses is the least significant bit.
2. Parity is optional on all data busses that have parity bits.
3. An * at the end of signal name indicates that the signal is an active-low signal.
4. OD indicates an open drain output.
5. POD indicates a programmable open drain output.
6. All input and bi-directional I/O pins have internal pull-ups or pull-downs, which force an inactive state when the pin is left open.
7. All unused input pins must be tied to inactive state.

3.2.2.1 Processor Interface

The Processor interface is a slave interface that is used by the local host/processor to read and write the internal registers of the device. These registers are identified in Chapter 5. It is also used to interrupt the processor/host when an unmasked status bit is set. Table 3-8 shows the SARA-R processor interface pin descriptions.

Symbol	Type	Name and Function
CS*	I	Chip select signal to enable the processor interface for data transfers
DS*	I	Data strobe for enveloping the data transfer
WRT	I	WRITE(1), READ(0) input to indicate the direction of data transfer
RDY*	OD	SARA-R signal indicating completion of data transfer
A(7:0)	I	Address bus for internal register selection
D(15:0)	I/O	Processor interface data bus
INTR*	OD	Interrupt signal to the processor

Table 3-8. SARA-R Processor Interface Pin Descriptions

3.2.2.2 Cell Interface

The ATM cells are read by the SARA-R through the cell interface. The interface may be configured as either byte or word width and parity is selected by a mode bit. Cells may be received and read-in without boundaries or may be demarcated by either a tag bit or a parity inversion. Table 3-9 shows the SARA-R cell interface pin descriptions.

Symbol	Type	Name and Function
CLAV	I	Cell available in the FIFO(s).
FFMT	I	FIFO empty input that is used to detect data-insufficiency in a FIFO. Active when one or more FIFO(s) are empty.
FFRD(3:0)	O	FIFO Read enable. FFRD(0) is used as read signal, and FFRD(3:1) are used for testing.
FFSL(1:0)	O	These pins are used for testing and should be left open
FFD(17:0)	I	FIFO data-bus. FFD(17) & FFD(16) are the parity bits for FFD(15:8) & FFD(7:0) respectively. Either the full 16-bits or 8-bits of the data bus can be selected to transfer the data. In the 8-bit mode, the data is transferred on FFD(7:0) and uses FFD(16) as parity.
FFLUSH	O	FIFO flush signal is used to request the appropriate FIFO controller to flush the FIFO(s) when data misalignment is detected.
FLSHDONE	I	Asserted when all FIFO(s) are empty acknowledging FFLUSH.

Table 3-9. SARA-R Cell Interface Pin Description

3.2.2.3 Control Memory Interface

The control memory interface is a bus-master interface to the control memory. It is used to access the communication queues, virtual circuit/virtual path lookup tables, the reassembly table and the buffer descriptors maintained in the control memory. This interface supports both synchronous and asynchronous bus interfaces of various speeds. Table 3-10 shows the SARA-R control memory interface pin descriptions.

Symbol	Type	Name and Function
CREQ	O	Control memory bus request.
CGRT	I	Control memory bus grant.
CMULR	O	Control memory multiple request signal, indicating that more than one data transfer is pending.
CCYCST*	POD	Control memory cycle start signal marking the beginning of data transfer. Programmable as either open drain or as an active output.
CRDY*	I	Control memory ready signal indicates the end of each transfer.
CWRT	O	Control memory write signal for direction of transfer. This pin can be programmed as an early write signal to provide additional flexibility in external system design.
CA(23:1)	O	Control memory address bus (16-bit word address).
CD(17:0)	I/O	Control memory data bus. CD(17) & CD(16) are the parity bits for CD(15:8) & CD (7:0) respectively.

Table 3-10. SARA-R Control Memory Interface Pin Descriptions

3.2.2.4 Packet Memory Interface

The packet memory interface is a bus-master interface for writing cells to the packet memory where these cells are reassembled into packets. Various bus interfaces are supported. SARA-R packet memory interface pin descriptions are shown in Table 3-11.

Symbol	Type	Name and Function
PREQ	O	Packet memory bus request
PGRT	I	Packet memory bus grant
PCYCST*	POD	Packet memory cycle start (marks the beginning of data transfers). Programmable as either open drain or as an active output.
PRDY*	I	Packet memory ready, validates each data transfer.
PD(35:32)	O	PD(35), PD(34), PD(33), PD(32) are the parity bits for PD(31:24), PD(23:16), PD(15:8), PD(7:0) respectively.
PD(31:0)	O	Data path to the packet memory. Output for multiplexed address in PM_REQADR mode (Byte address PA(31:0) is presented on these pins during packet memory bus requests).

Table 3-11. SARA-R Packet Memory Interface Pin Descriptions

Symbol	Type	Name and Function
PA(15:0)	O	Packet memory address bus. This bus is driven with 16-bits of the byte address of the Packet memory location being accessed. The selection of the 16-bits is dependent upon the state of the PLWADR pin and the PM_INTRLV mode bit.
PLWADR	I	Input signal to multiplex the upper or lower part of the packet memory address on the PA bus.
PAMTCH	O	Indicates that the packet memory address matches with the contents of the address match register. Used to detect the end of the serial address memory (for VRAMs) or page boundaries.

Table 3-11. SARA-R Packet Memory Interface Pin Descriptions

3.2.2.5 Congestion Control Interface

This interface is used to transfer the VCI and congestion code of the congestion notification cell to the SARA-S. The SARA-S can use this index and the congestion notification code to modify the rate of cell generation of the particular VC source. The SARA-R congestion control interface pin descriptions are shown in Table 3-12.

Symbol	Type	Name and Function
CCDATA	O	Serial input stream of congestion control information from SARA-R to SARA-S.
CCXFER	O	Congestion control transfer enable for CCDATA.
CCHLD	I	Hold congestion control transfer.

Table 3-12. SARA-R Congestion Control Interface Pin Descriptions

3.2.2.6 Miscellaneous Signals

Miscellaneous signal pin descriptions for SARA-R are shown in Table 3-13. The test pins TE, TEI, and SI are provided for manufacturing test purposes only. They must be tied to “0” (low) in user applications.

Symbol	Type	Name and Function
CBRMT*	O	CBR FIFO empty
RST*	I	Hardware reset
CLK	I	Clock
TE	I	Scan test enable input.
SI	I	Scan input for SARA-R in scan mode.
SO	O	Scan output for SARA-R in scan mode.
TEI	I	Tri-state all outputs and bidirectional pins

Table 3-13. SARA-R Miscellaneous Signals Pin Descriptions

3.3 Hardware Interface Operation

3.3.1 Processor Interface

The processor/host interface is used by the external controller to access the internal registers of the SARA chips and to issue commands and instructions to the devices. It is comprised of a chip-select signal (CS*), data-strobe signal (DS*), write/read signal (WRT), ready signal (RDY*), an 8-bit address bus (A[7:0]) and a 16-bit data bus (D[15:0]). It also includes an interrupt signal (INTR*) to the interrupt handler.

It operates as an asynchronous slave interface. An access is triggered when both the CS* and DS* become active. The type of access (write or read) is determined by the state of the WRT signal. For the processor to write to the SARA chips, the WRT signal must be “1”; WRT must be “0” when the processor is to read from the SARA chips. The contents of the address-bus (A[7:0]) selects the register of interest or the command to be issued. Figure 3-3 shows the processor interface functional timing.

If the access is a read, the SARA chip will latch the data of the appropriate register, and then assert the RDY* signal. Upon the assertion of the RDY* signal, the processor/host can read the data on the data-bus and then terminate the access by de-asserting CS* and/or DS*.

If the access is a write, the SARA chip will load the contents of the data-bus into the appropriate register, and then assert the RDY* signal. At this time the processor can terminate the access by de-asserting the CS* and/or DS* signals.

This interface allows only a single data transfer per access. Back-to-back accesses require that either CS* or DS* be de-asserted for a duration of at least one-and-one- half CLK periods.

The INTR* signal is driven active whenever an unmasked bit is set in the status register of the device. The status register of the SARA will be automatically cleared after it is read by the processor through the processor interface. This in turn, will force the INTR* signal inactive. INTR* will not go active until an unmasked bit is set in the status register. Internal circuitry in the devices force the INTR* signal to be inactive for at least four CLK cycles even though an unmasked bit gets set in the status register.

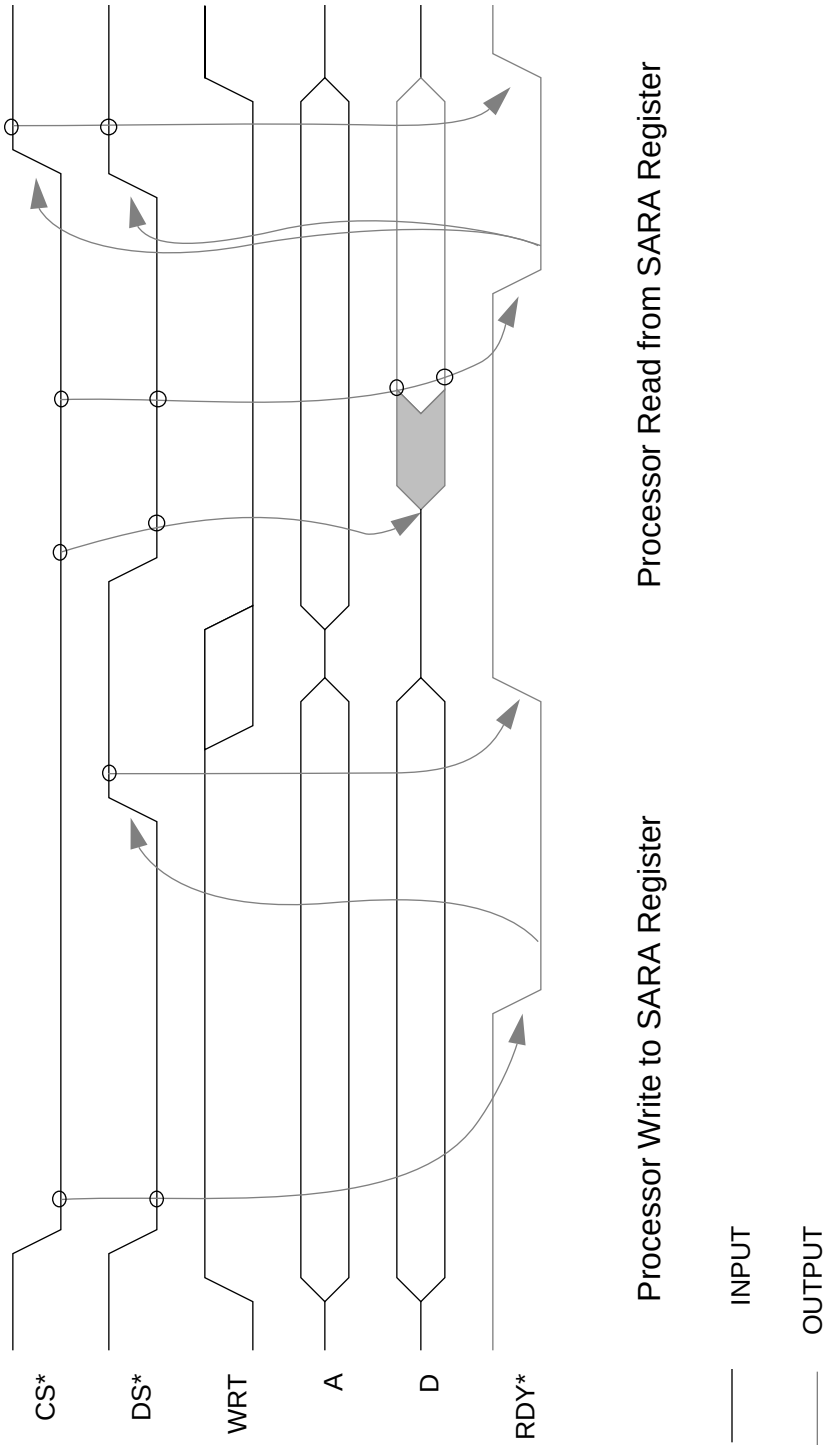


Figure 3-3. Processor Interface Functional Timing

3.3.2 Control Memory Interface

The Control memory interface is used by the SARA chips to access the control data structures and communication queues that are maintained in the control memory. This is a bus-master interface and can be programmed in a synchronous mode of operation, whereby all signals are synchronous to CLK and multiple data transfers are allowed per access, or in an asynchronous mode of operation where the input signals are synchronized inside the device and only single data transfers are allowed per access. This mode is controlled by the mode bit CM_IF_ASYNC in the mode register(s) of the appropriate device.

3.3.2.1 Address Bus CA(23:1)

The address bus is 23-bits wide (16-bit word addresses) and is driven active during the entire duration of the control memory data transfers, i.e. when CCYCST* is active.

3.3.2.2 Data Bus CD(17:0)

The upper 2-bits (17:16) of the data bus are the parity bits of the corresponding bytes. The PARITY bit in the mode register selects the type of parity: 0 - Odd parity, 1 - Even parity. The SARA chips always generate the parity bits during a write operation, but will only check the parity bits if the mode bit CM_PAR_EN bit is set high (“1”).

3.3.2.3 Control Interface

The control memory control is composed of two sets of signals. The signals CREQ, and CGRT are used to arbitrate for access to the bus. The signals CMULR, CWRT, CCYCST* and CRDY* are used for data transfer. These signals are further described in detail for both the synchronous and the asynchronous modes of operation.

3.3.2.4 Synchronous Mode

Bus Access

CREQ is driven active when SARA-R or SARA-S needs access to the control memory bus for data transfers to or from the control memory. CREQ remains active until the CGRT input signal becomes active; CREQ is driven inactive on the next clock cycle after CGRT becomes active. CGRT must remain active for the duration of the entire memory access and must be forced inactive after memory access to complete the bus handshake. The SARA will not make a new request until CGRT becomes inactive. Figure 3-4 through Figure 3-8 show the various timings of the control memory interface. Figure 3-5 and Figure 3-8 show examples of an early write cycle, where CWRT terminates one clock cycle before the end of the last write.

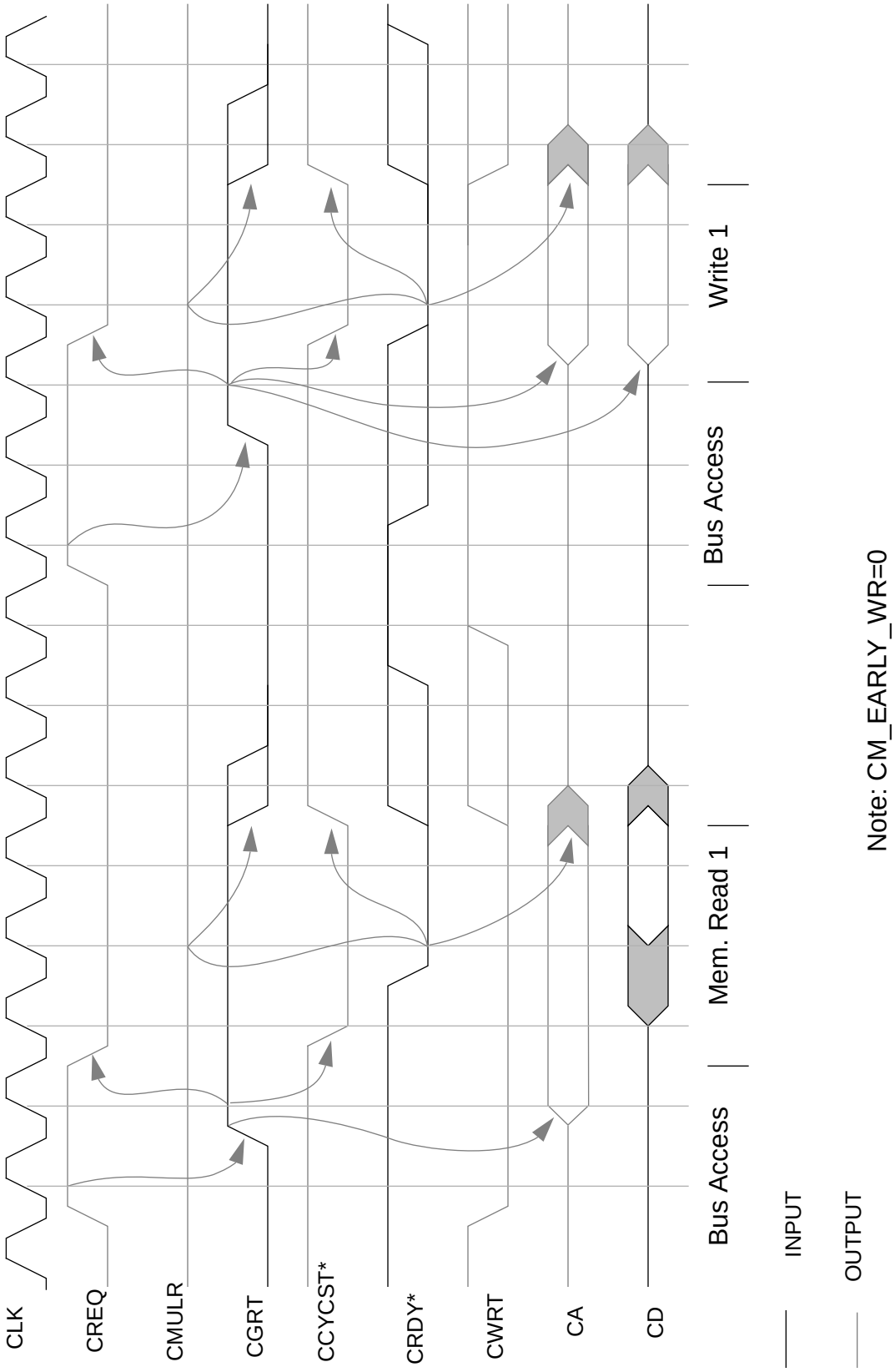


Figure 3-4 Control Memory Interface—Single Access Read and Write—Normal Write Mode

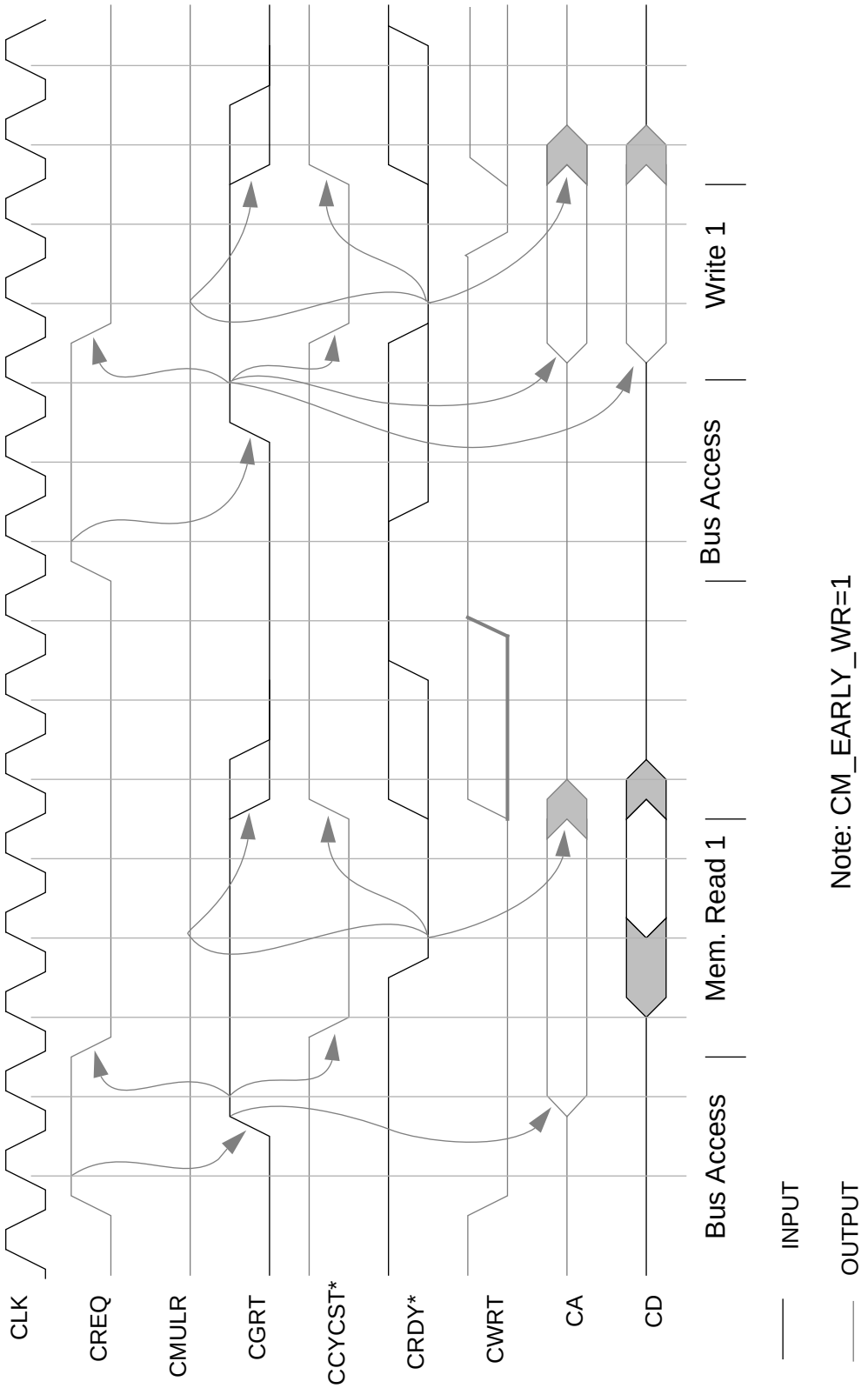


Figure 3-5.Control Memory Interface—Single Access Read and Write—Early Write Mode

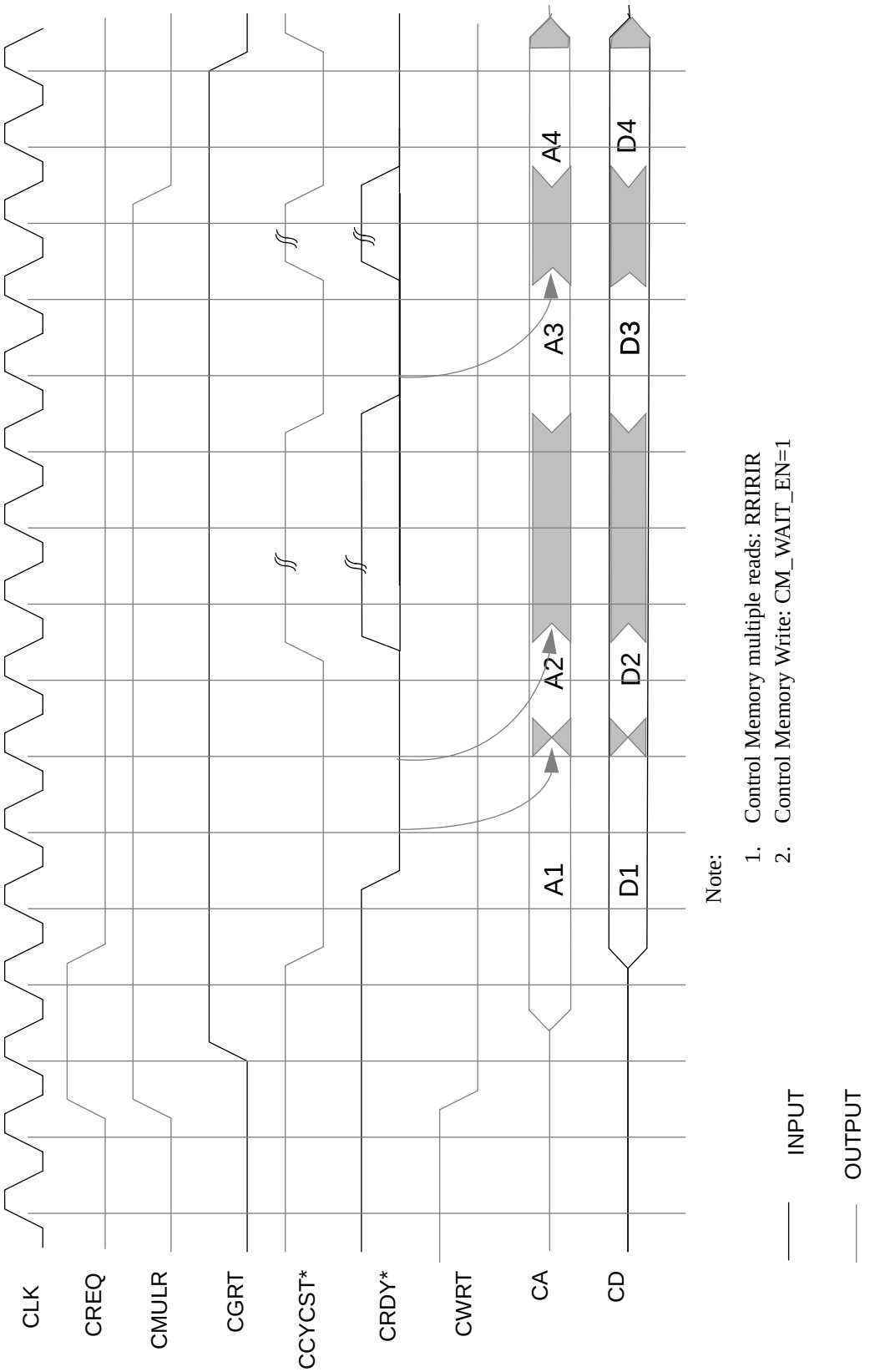


Figure 3-6. Control Memory Interface—Multiple Reads

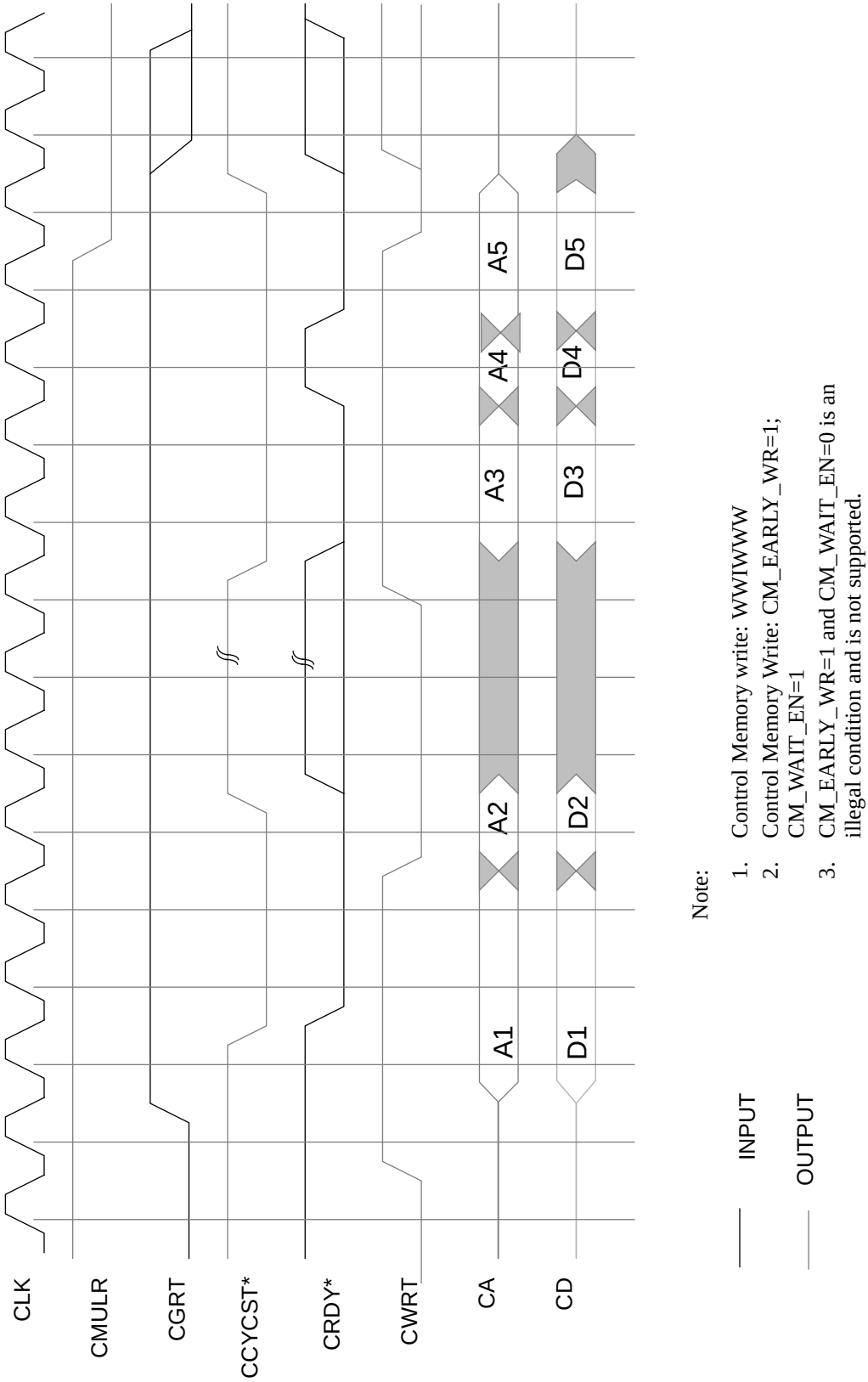


Figure 3-7. Control Memory Interface—Multiple Writes

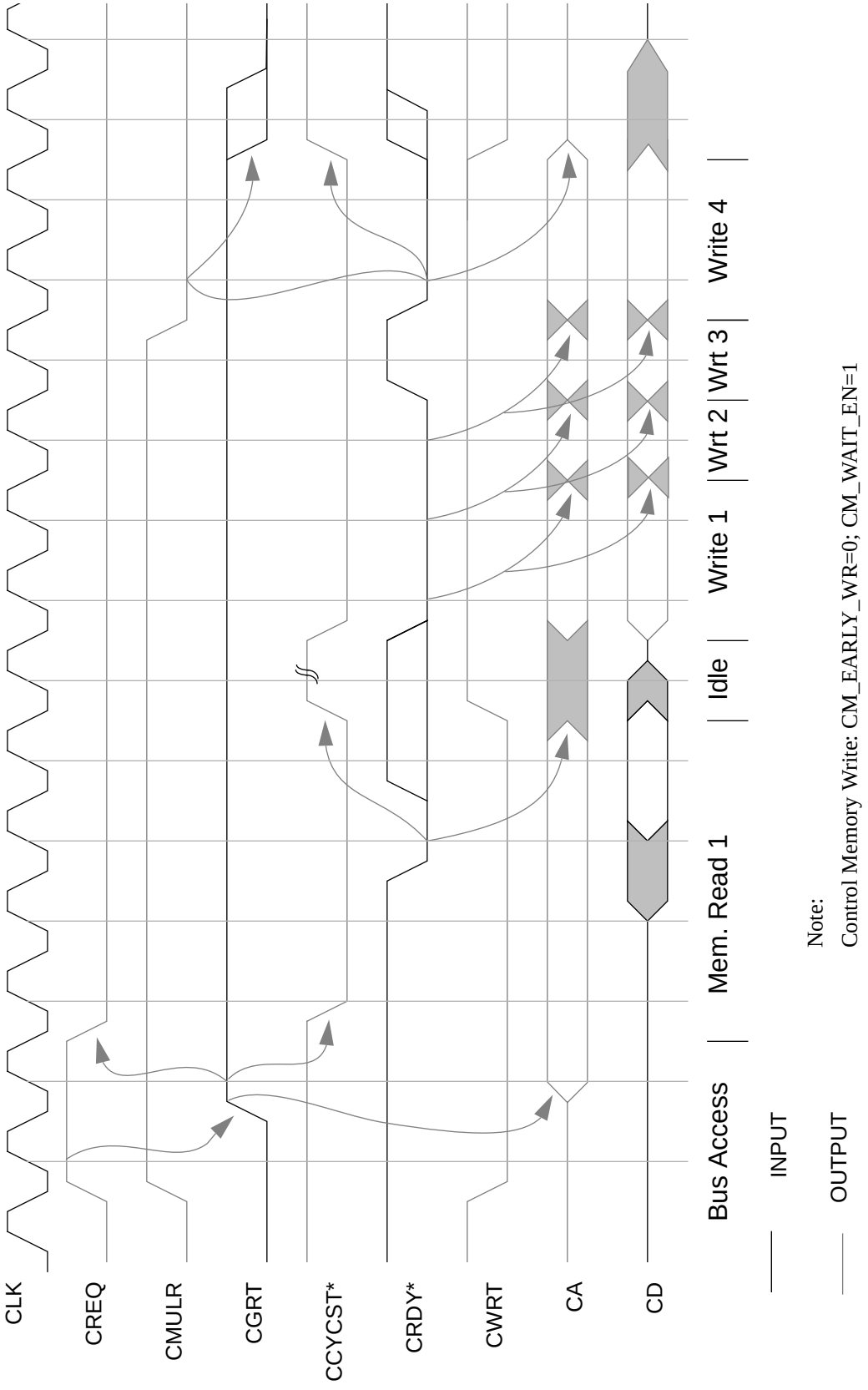


Figure 3-8. Control Memory Interface—Single Read/ Multiple Writes—Normal Write Signal

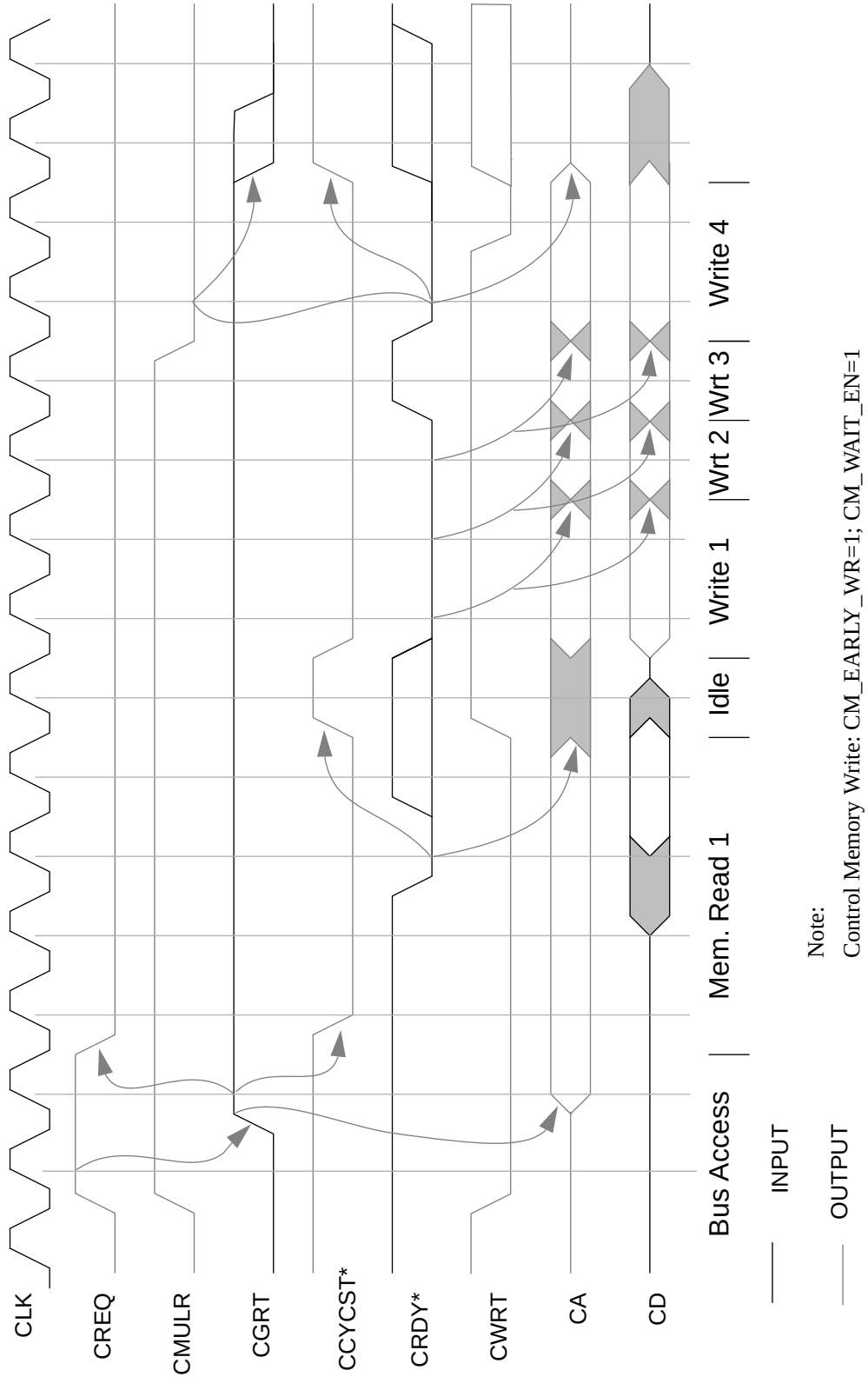


Figure 3-9. Control Memory Interface—Single Read/ Multiple Writes—Early Write Signal

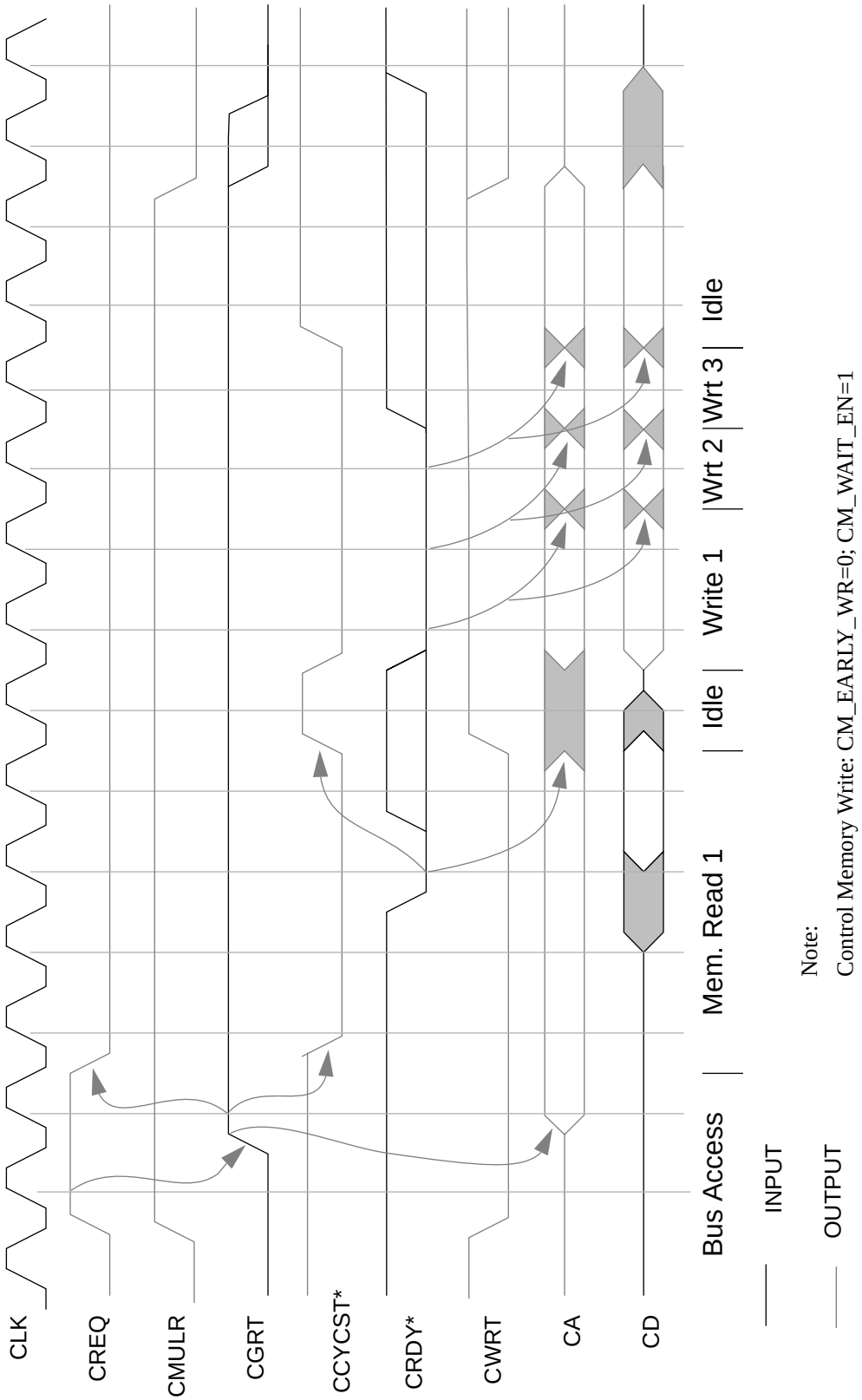


Figure 3-10. Control Memory Interface—Without Last Access

Data Transfer

The CGRT signal going active marks the beginning of the data transfer(s). The signal CCYCST* is driven active in the same cycle when CREQ becomes inactive. If the number of data transfers is expected to be greater than one, the CMULR signal is also driven active. This signal remains active until the end of the penultimate data transfer. The address bus CA(23:1) is driven active (it is 3-stated when CCYCST* is inactive). While CCYCST* is active, the device reads from the data-bus when CWRT is “0” (read data transfer) or drives the data onto the data bus when the CWRT signal is “1” (write data transfer). During a burst of data transfers, if the access changes from a read to a write transfer or from a write to a read transfer or if there are no transfers but the SARA wants to hold the bus, the CCYCST* signal will be driven inactive for at least one cycle, and the CMULR signal will remain active indicating that the SARA is still active on this interface.

Note that if the SARA is holding the control memory interface but not performing any data transfers, it will hold the CMULR signal active and the CCYCST* signal inactive. On the last access the SARA will force the CMULR signal inactive. If the SARA does not need to access control memory after the idle cycle, then it will release the interface by deasserting the CMULR signal, as shown in Figure 3-10.

The duration of each data transfer is controlled by the CRDY* input signal. The CRDY* signal (when CCYCST* is active) is delayed internally by one CLK cycle, then used to latch the data and generate the next address. By holding the CRDY* input signal inactive the duration of the transfer can be extended, since each transfer is completed only when CRDY* is active. On the completion of the last word transfer, the signal CCYCST* is driven inactive and the address and data busses are 3-stated. The CRDY* signal must be forced inactive after the completion of the access.

When the CM_WAIT_EN bit is set to 1, connecting CRDY* to CCYCST* will result in a two cycles for the first data transfer and single clock cycle for the remaining data transfers.

The SARA chips can be preempted on this interface by deactivating the CGRT signal. The data transfers will be terminated in the cycle following the cycle when the CGRT signal becomes inactive. If CM_WAIT_EN = 1, the CRDY* signal must also be forced inactive when CCYCST* is driven inactive. The SARA chips will re-request for the bus and continue from the last valid data transfer.

If the mode bit CM_EARLY_WR is set to a “1” (Figure 3-5, Figure 3-7, and Figure 3-8), then the CWRT pin will be forced to a ‘0‘in the clock cycle before CCYCST* becomes inactive. This allows for external hardware to generate synchronous write signals to the control memory.

3.3.2.5 Asynchronous Mode

Bus Access

CREQ is driven active when the SARA-R or SARA-S needs access to the control memory bus for data transfer to/from the control memory. CREQ is driven inactive after CGRT becomes active. CGRT must remain active for the duration of the entire memory access and must be forced inactive after memory access to complete the bus handshake.

Data Transfer

The CGRT signal going active marks the beginning of data transfer. The signal CCYCST* is driven active in the same cycle when CREQ becomes inactive. (The CMULR and CM_EARLY_WR signals have no meaning in the asynchronous mode of operation.) The address bus CA(23:1) is driven active (it is 3-stated when CCYCST* is inactive). When the CCYCST* is active, the device reads from the data-bus when CWRT is “0” or drives the data onto the data bus when the CWRT signal is “1”.

The CRDY* signal (when CCYCST* is active) should be driven active after the data has been written to the memory (write transfer) or when the data is available on the data bus (read transfer). CRDY* is synchronized internally, then used to latch the data. By holding the CRDY* input signal inactive, the duration of the transfer can be extended, since the transfer is completed only when CRDY* becomes active. Once CRDY* is sensed active by the SARA chips, the signal CCYCST* is driven inactive, and the address and data busses are 3-stated. The CRDY* signal must be forced inactive for the completion of the access.

Forcing CGRT inactive before the completion of the access will invalidate the data transfer. The SARA chips will re-request for the bus and continue from the last valid data transfer.

3.3.3 Packet Memory Interface

The packet memory interface is used by the SARA chips to transfer packet data to and from the packet memory. This is a bus-master interface that can be used with a wide variety of memories. It can be programmed in a synchronous mode of operation where all signals are synchronous to CLK and multiple data transfers are allowed per access, or in an asynchronous mode of operation where the input signals are synchronized inside the device and only single data transfers are allowed per access. This mode is controlled by the bit PM_IF_ASYNC in the mode register(s) of the appropriate device.

The interactions of the packet memory interface are also controlled by the contents of the mode registers and the address match register. The effect of the mode bits is described in relation to the appropriate interface signals in the following paragraphs.

3.3.3.1 Data Bus PD(35:0)

The upper 4-bits PD(35:32) of the data bus are the parity bits of the corresponding bytes. The PARITY bit in the mode register of the appropriate SARA selects the type of parity: 0 - odd parity, 1 - even parity. The SARA-R always generates the parity bits, and the SARA-S will only check the parity bits if the mode bit PM_PAR_EN is set to a “1”. When the mode bit PM_IF_WORD is set to “1”, indicating a 16-bit data-interface, then only the data-bits {PD(33:32), PD(15:0)} of the data bus are used.

If the mode bit PM_REQADR is set to “1” in the mode register of the appropriate SARA, the 32-bit packet memory address is multiplexed onto the data-bus bits PD(31:0) when the SARA is requesting access to the bus (PREQ is active and PGRT is inactive), and when PLWADR signal input is a “1.” Table 3-14 shows the state of the 32 bits of the data bus as a function of PREQ, PGRT, and PLWADR.

PREQ	PGRT	PLWADR	PD(31:0)
1	0	0	3-state
1	0	1	Address
1	1	X	Data

Table 3-14. Interactions of PREQ, PGRT, and PLWADR with the Packet Data Bus PD(31:0)

3.3.3.2 Address Bus PA(15:0)

While the PA address bus is 16-bits wide, the packet memory internal to the SARA devices can have 24 bits of address, allowing 16 MBytes of packet data. When the PLWADR signal is “0”, the SARA-R and SARA-S multiplexes the upper bits of the packet address onto the address bus PA(15:0). The lower bits of the address (byte address) are correspondingly multiplexed

when the PLWADR signal is a “1”. Note that this does not depend on the CLK signal. The upper and lower part of the address overlap on 8-bits (PA(15:8)). This allows a range of DRAMs with different RAS and CAS address-widths.

The actual bits on the address bus also depend upon the state of the bits PM_INTRLV and PM_IF_WORD in the mode registers. When the PM_INTRLV mode bit is set and when the upper address bits are selected, the second or first address bit (rather than the eighth address bit) is driven on PA(0). This is particularly useful if the packet memory is organized as a 2-way interleaved memory, since this allows the selection of the appropriate memory bank when the upper address bits are selected (e.g. RAS address).

Table 3-15 lists the interactions of the mode bits and the state of the PLWADR pin.

PLWADR	PM_INTRLV	PM_IF_WORD	PA(15:0)
(Input pin)	(Mode bit)	(Mode bit)	(Output pins)
0	0	X	Address bits 23 through 8.
0	1	0	Address bits 23 through 9 and address bit 2.
0	1	1	Address bits 23 through 9 and address bit 1.
1	X	X	Address bits 15 through 0

Table 3-15. Mode Bit/PLWADR Interactions

The SARA chips also contain circuitry to support detection of packet memory address boundaries. The packet memory address, PA(13:2) bits are constantly compared with the contents (11:0) programmed in the address match register. The upper four bits are only compared when the corresponding bits (15:12) in the address match register are set to “0”. When the packet memory address matches the contents of the address match register, the PAMTCH signal becomes active. This feature can be used by external circuitry to detect page-boundaries, end of SAM (serial address memory) in a VRAM (Video DRAM), etc., and stop the SARA data-transfer to change the page or SAM.

The SARA chips always interrupt the data transfers when the packet address rolls over the 16-bit boundary, by releasing the bus. The SARA chips will re-request the bus and then restart from the next address. This is useful if the upper bits of the address are latched and are used to drive the address lines of memory.

3.3.3.3 Control Signals

The packet memory control is comprised of two sets of signals. PREQ and PGRT are used for the packet memory bus handshake. PCYCST*, PRDY*, PLWADR are used for data transfers. These signals are further described in detail for both modes.

3.3.3.4 Synchronous Mode

Bus Access

The PREQ signal is driven active when the SARA-R or SARA-S needs access to the bus for data transfers to or from the packet memory. The PREQ signal remains active until the PGRT input signal becomes active. The PREQ signal is driven inactive on the next clock cycle after PGRT becomes active. PGRT must remain active for the duration of the entire data transfer, and must be driven inactive after the access to complete the handshake.

Data Transfer

The PGRT signal becoming active marks the beginning of data transfer. The signal PCYCST* becomes active in the same cycle when PREQ becomes inactive. The address bus PA(15:0) is driven active (it is 3-stated when PCYCST* is inactive). There are no packet memory read or write signals on the SARA chips since the SARA-S only reads from and the SARA-R only writes to the packet memory. When the PCYCST* becomes active, the SARA-R drives the data onto the data bus PD(35:0). The signal PCYCST* will remain active through the entire data transfer.

The duration of each data transfer is controlled by the PRDY* signal. The PRDY* signal (when PCYCST* is active) is synchronized internally and is used to generate the next address and latch the input data (SARA-S) or the output data (SARA-R). By driving the PRDY* signal inactive, the access time of the transfer can be extended. On the completion of the last word transfer, the signal PCYCST* is forced inactive and the address and data bus are 3-stated.The PRDY* and PGRT signals must also be driven inactive at the completion of the access. Figure 3-11 and Figure 3-12 show the packet memory interface timings.

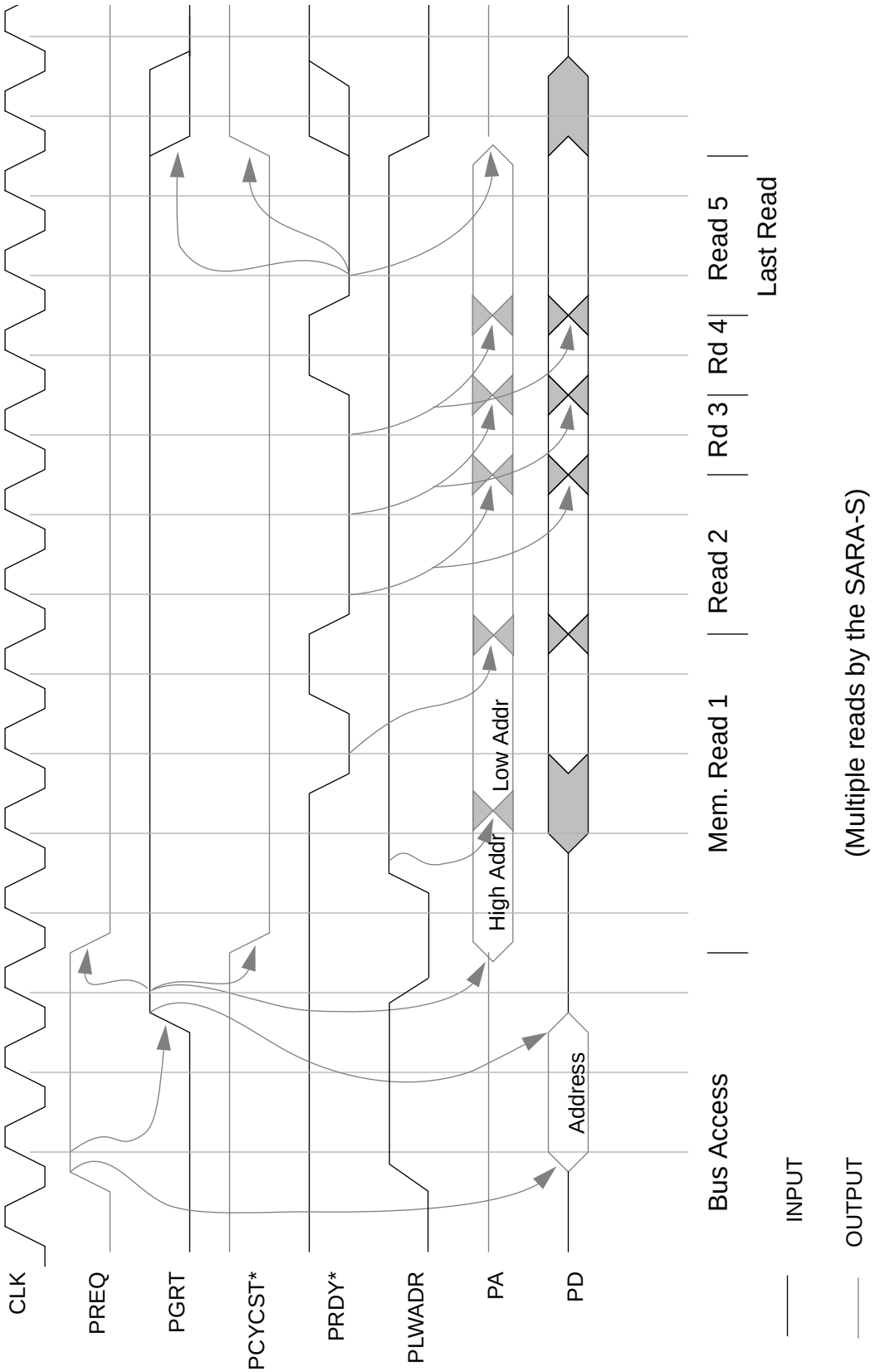


Figure 3-11. Packet Memory Interface—Address Driven on Data Bus During Bus Access

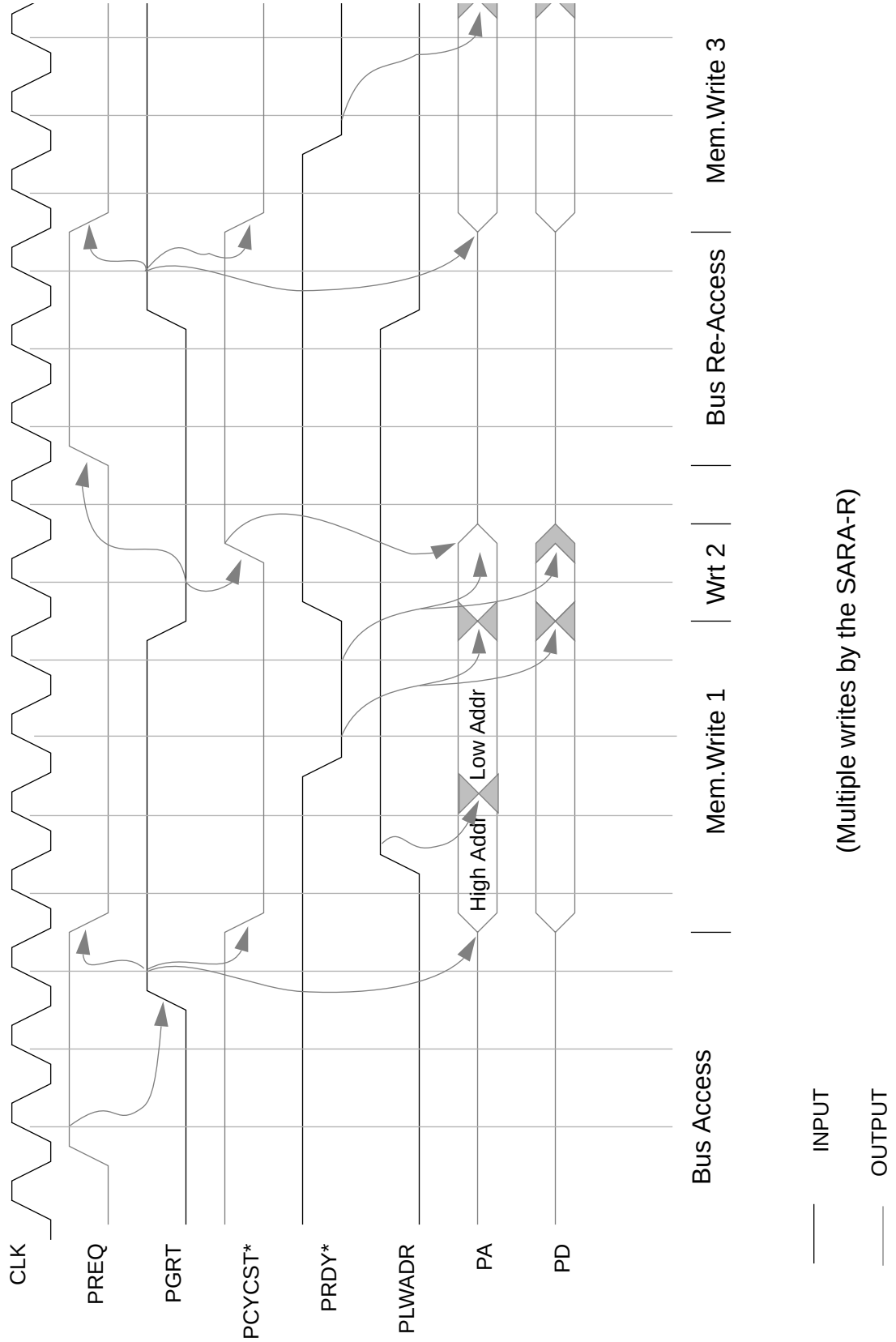


Figure 3-12. Packet Memory Interface—Bus Preempted by De-assertion of PGRT

The SARA chips can be preempted on this interface by de-asserting the PGRT signal. The data transfer will be terminated in the following cycle after the PGRT signal becomes inactive. The SARA chips will re-request for the bus and continue from the last data transfer.

3.3.3.5 Asynchronous Mode

Bus Access

PREQ is driven active when either SARA chip needs access to the bus for data transfer to or from the packet memory. PREQ is driven inactive after PGRT becomes active. PGRT must remain active for the duration of the entire memory access and must be forced inactive after to access to complete the bus handshake.

Data Transfer

The PGRT signal going active marks the beginning of data transfer. The signal PCYCST* is driven active in the same cycle when PREQ becomes inactive. The address bus PA(15:0) is driven active (it is 3-stated when PCYCST* is inactive). When the PCYCST* becomes active, the SARA-R drives the data onto the data bus.

The PRDY* signal (when PCYCST* is active) should be driven active either after the data has been written to the memory (SARA-R), or when the data is available on the data bus (SARA-S). PRDY* is synchronized internally, then is used to latch the data. By holding the PRDY* input signal inactive, the duration of the transfer can be extended, since the transfer is completed only when PRDY* becomes active. Once PRDY* is sensed active by the SARA chips, the signal PCYCST* is driven inactive and the address and data bus are 3-stated. The PRDY* and PGRT signals must be forced inactive for the completion of the access.

Forcing PGRT inactive before the completion of the access will invalidate the data transfer. The SARA chips will re-request for the bus and continue from the last valid data transfer.

3.3.4 Link/Cell Interface

The SARA devices connect to the transmission logic through the link/cell interface. This interface is designed to work synchronously and can be configured to have either an 8- or a 16-bit data path. This interface can optionally be protected by parity. On the receive path, the SARA-R supports demarcation of cells. The beginning of a cell can be indicated by inverting the parity on the first word of the cell. The cell boundary can also be indicated by setting the parity bits to “1” if parity is not used on this interface. The byte ordering of the data on this interface is shown in Table 3-16.

8-bit mode	
FFD16	**FFD7-FFD0**
t	hdr0
p	hdr1
p	hdr2
p	hdr3
p	hec
p	pyld0
p	pyld1
p	pyld2
p	.
p	.
p	.
p	pyld46
p	pyld47

16-bit mode			
FFD17	**FFD16**	**FFD15 - FFD8**	**FFD7 - FFD0**
t	t	hdr0	hdr1
p	p	hdr2	hdr3
p	p	hec	pyld0
p	p	pyld1	pyld2
p	p	pyld3	pyld4
p	p	pyld5	pyld6
p	p	pyld7	pyld8
p	p	.	.
p	p	.	.
p	p	.	.
p	p	pyld43	pyld44
p	p	pyld45	pyld46
p	p	pyld47	--

Table 3-16. Byte Ordering of Data for Link/Cell Interface

For SARA-S, ‘t’ & ‘p’, are parity bits of the corresponding bytes; for SARA-R, these bits depend on whether boundary check (mode bit CI_BNDRY_CHK) is enabled or disabled and whether parity check (mode register bit CI_PAR_EN) is enabled or disabled on this interface.

CI_BNDRY_CHK = 1, CI_PAR_EN = 1: p => parity, t => inverted parity

CI_BNDRY_CHK = 1, CI_PAR_EN = 0: p => 0, t => 1

CI_BNDRY_CHK = 0, CI_PAR_EN = 1: p => parity, t => parity

CI_BNDRY_CHK = 0, CI_PAR_EN = 0: p => 0, t => 0

3.3.4.1 SARA-S

For SARA-S, the link/cell interface is designed to emulate a clocked FIFO interface. The interface works on the clock signal RDCLK, which is independent of the SARA-S CLK signal and is typically based on the link clock. When a cell is available for transmission in the payload buffer, the SARA-S sets the signal CELAVL* active and drives the first word (8-bit or 16-bit depending upon the CI_WIDTH16 mode bit) on the FFD(17:0) bus. The external logic asserts the RDEN signal when it is ready to transmit the data. The SARA-S drives the next data word on the FFD(17:0) bus on the rising transition of the RDCLK when RDEN and CELAVL* are both active. CELAVL* will be active for the entire cell transfer duration. When the last word is read out, the SARA-S de-asserts the CELAVL* signal. There is a minimum of four clock cycles before CELAVL* goes active again.

SARA-S responds to the XON input by transferring cells to the cell interface as long as the XON input is always true. If the XON input is false (low), then cell transfers of selected rate queues (and optionally the CBR traffic) are temporarily suspended while the XON signal is false. The suspension occurs gracefully on cell boundaries if the XON signal goes false during rate queue processing.

3.3.4.2 SARA-R

The SARA-R link/cell interface is designed to communicate with the read port of a clocked FIFO. This interface works on the SARA-R clock signal CLK. The CLAV input signal is driven active when a cell is available in the external FIFO (all 53-bytes of the cell must be available). The SARA-R reads either eight- or 16-bit data as determined by the CI_WIDTH16 on the FFD(17:0) bus by asserting the FFRD(0) signal. If the boundary check is enabled, the SARA-R reads the data until it finds the start of cell and then reads the complete cell. The CLAV needs to be valid until the first byte/word is read in. The CLAV may be deasserted any time after that. The SARA-R reads the complete cell before it senses the CLAV signal for the next cell. The SARA-R de-asserts the FFRD(0) signal after reading the first seven bytes (four words) of a cell. The SARA-R then re-asserts the FFRD(0) to continue reading the rest of the cell.

If the external FIFO becomes empty (FFMT goes active) before the entire cell is read in, the SARA-R will terminate the processing on that cell and assert the FFLUSH signal. The FFLUSH signal is also asserted when boundary check is disabled and the SARA-R sees two cells back-to-back with HEC errors. This signal remains active until the external logic asserts FLSHDONE. The external logic must hold-off further writes to the FIFO until a new start of cell is detected. This will prevent the generation of a runt cell in the FIFO after the FLUSH signal is asserted. Figure 3-13 and Figure 3-14 show the cell interface timings.

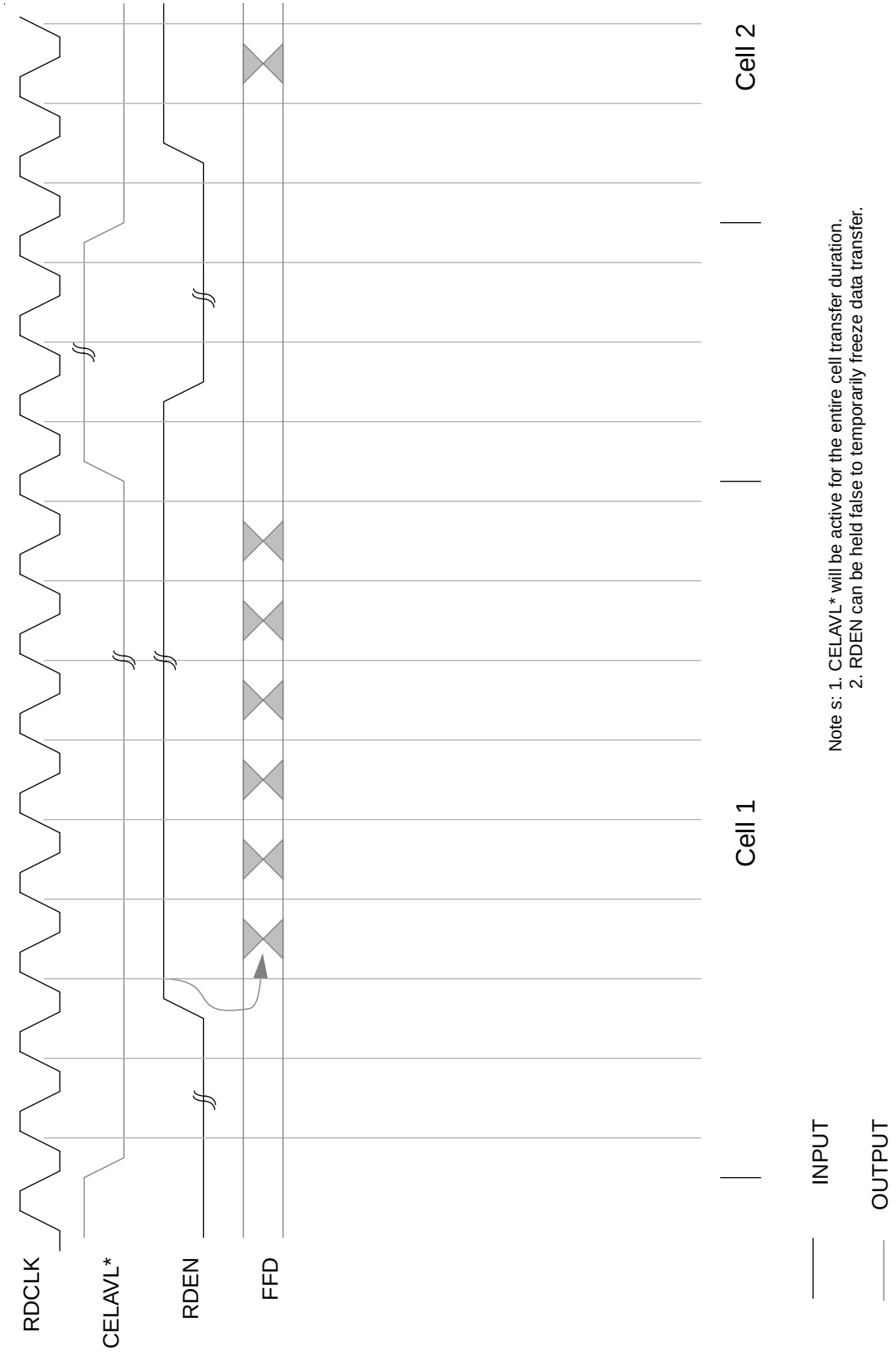


Figure 3-13. Cell Interface—SARA-S

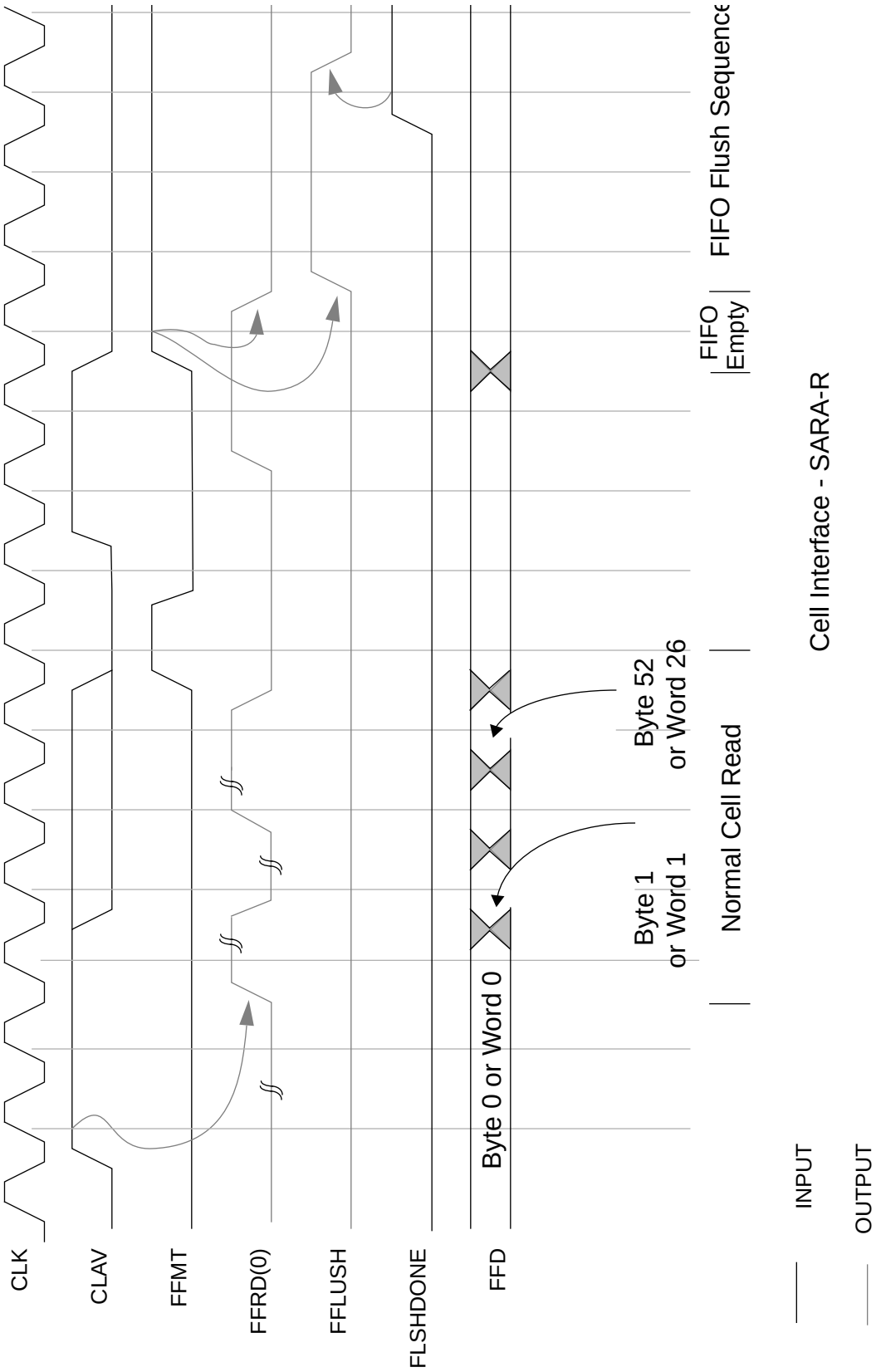


Figure 3-14. Cell Interface—SARA-R

3.3.5 Congestion Control Interface

The congestion control interface is used to transfer the congestion control information from the SARA-R to the SARA-S upon the receipt of a congestion notification cell by the SARA-R. The format of congestion notification cells is defined in Chapter 5.

This interface is comprised of two control signals, CCXFER and CCHLD, and a serial data signal, CCDATA. When a congestion notification cell is received, if CCHLD is inactive, the SARA-R drives CCXFER active and shifts out the 26 bits of serial data on the CCDATA pin. Once the SARA-S receives the congestion control information, it forces CCHLD active until it has processed the information and is ready to receive new congestion control information.

The bits that are transferred from the SARA-R to the SARA-S are shown in Figure 3-15. The most significant bit is shifted out first.

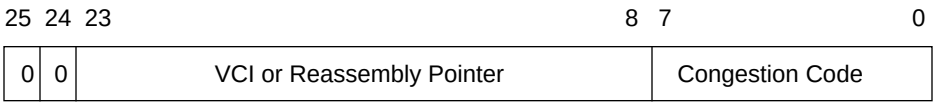


Figure 3-15. CCDATA Format

If the cell is received on a circuit that is setup to reassemble packets using the VCI and MID, then the lower 16 bits of the VCI are shifted out, otherwise if the circuit is setup to reassemble packets using the VPI and VCI fields, then the 16-bit reassembly pointer from the VP table is shifted out.

The congestion code is that which is carried in the payload of the congestion control cell, i.e. the second byte of the 48-byte payload. If the congestion code is 01H, the congestion is considered moderate and the segmentation peak rate on that virtual circuit is throttled by one notch. If the congestion code is 02H, the congestion is considered extreme and the segmentation of packets on that particular virtual circuit is immediately stopped (drops down to fifth notch).

3.3.6 Constant Bit Rate Interface

3.3.6.1 SARA-S

The SARA-S has a pair of signals CBRXMIT and CBRDONE for controlling the transmission of constant bit rate (CBR) traffic. These signals cycle through a complete handshake for every CBR cell transmission. When the external CBR source has loaded the CBR data into a predefined cell buffer in the packet memory, (the location of which is programmed in the SARA-S registers CBR_ADDR_HI and CBR_ADDR_LO), it drives the signal CBRXMIT active. The SARA-S senses this signal and at the next opportunity drives the CBRDONE signal low. The external logic may de-assert the CBRXMIT signal after the CBRDONE signal is driven high by SARA-S. After the cell has been loaded into the payload buffer, the SARA-S asserts the CBRDONE signal. The interface is then ready for the next CBR transmission. Figure 3-16 shows the constant bit rate interface timings.

CBR traffic can also be initiated using a software accessible mode bit (SEND_CBR) in the SARA-S. When the SEND_CBR bit is set, a CBR cell is transferred. The SEND_CBR bit gets automatically cleared after a successful CBR cell transfer (the software may poll if required). In addition, a maskable interrupt bit will be set.

The CBR buffer location in the packet memory could be mapped onto a real FIFO. The CBR data is stored in the buffer as a VC table entry pointer followed by the 48-bytes of payload. The VC pointer is used to fetch the four-byte cell header information from the VC table in the control memory. The 4-byte header, the header checksum, and the 48-byte payload are used to compose the 53-byte cell, which is then transferred to the cell interface

3.3.6.2 SARA-R

Any virtual circuit can be set up for CBR traffic. Cells received on these VCs are treated as CBR cells. The VC setup is described in section 7.3. When the SARA-R receives a cell on a CBR VC, the cell is loaded (header and payload) into a programmable circular chain of buffers in the packet memory. The circular chain of buffers is constructed by programming the SARA-R registers CBR_FIF_ST_ADR, CBR_FIF_ED_ADR, CBR_FIF_RD_PTR and CBR_FIF_WR_PTR. While this circular queue is not empty, the SARA-R has an output signal CBRMT*, which is driven high. This signal could be used to trigger the CBR receive logic to read the data from the circular buffers.

The destination of the CBR data could also be a real FIFO. To accomplish this, all the above defined CBR registers could be loaded with the location FIFO in the packet memory address space. This will result in the full and empty conditions to be active at same time. To prevent the SARA-R from responding to the full condition and dropping CBR cells, the IGNCBRFL bit in the mode register of the SARA-R could be set to a “1”, thus ignoring the full condition of the circular buffer.

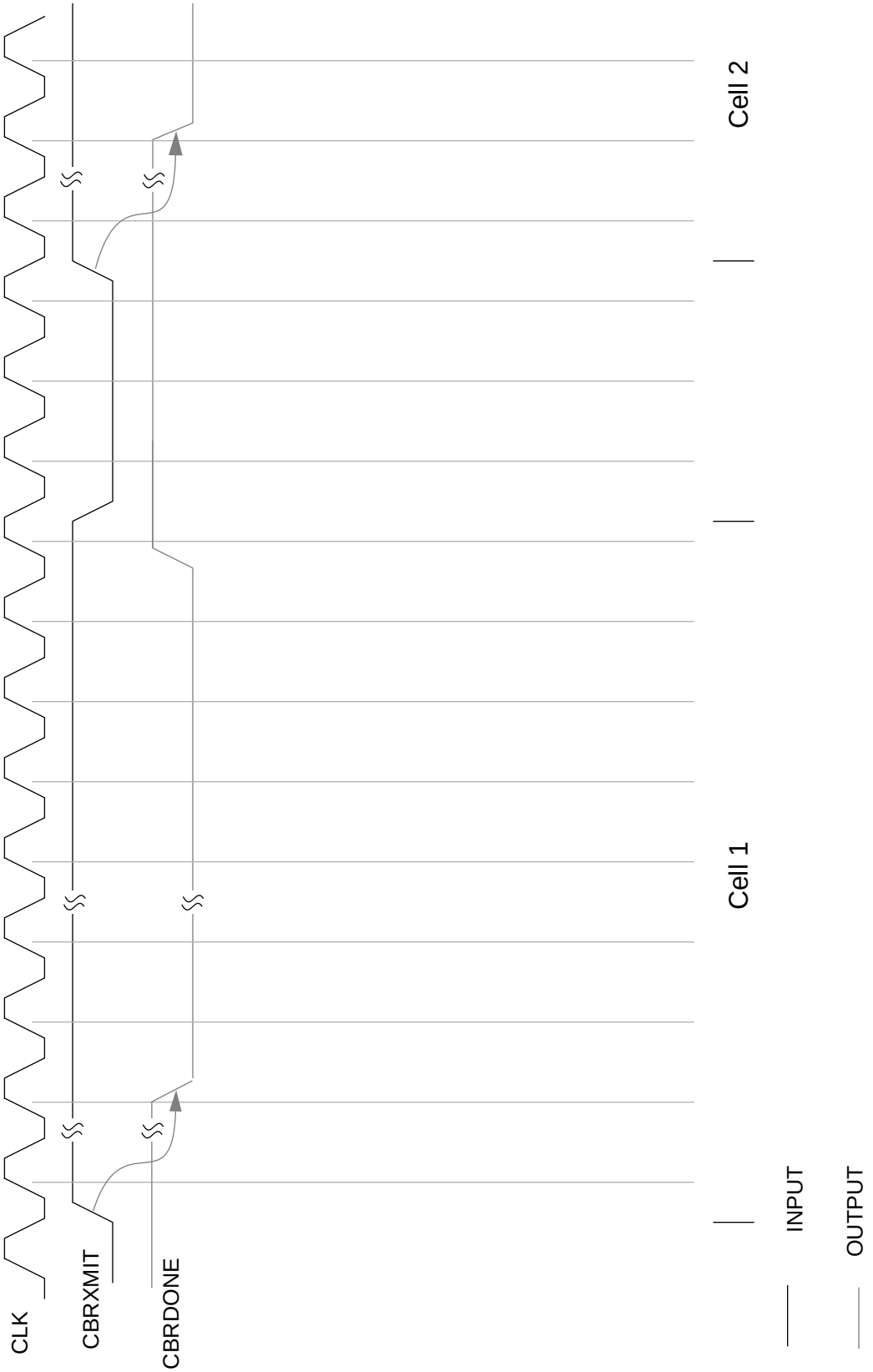


Figure 3-16. Constant Bit Rate Interface—SARA-S

3.3.7 Cross Reference by Pin Name (SARA-S)

Processor Interface		Control Memory Interface		Control Memory Interface (cont'd)		Packet Memory Interface (cont'd)		Cell Interface	
Signal	Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal	Pin
D15	130	CD17	32	CA5	75	PD9	198	FFD17	95
D14	131	CD16	33	CA4	76	PD8	199	FFD16	96
D13	132	CD15	34	CA3	77	PD7	200	FFD15	97
D12	133	CD14	36	CA2	78	PD6	201	FFD14	98
D11	134	CD13	37	CA1	79	PD5	202	FFD13	99
D10	135	CD12	38	CMULR	29	PD4	203	FFD12	100
D9	136	CD11	39	CREQ	31	PD3	205	FFD11	101
D8	139	CD10	40	CGRT	23	PD2	206	FFD10	102
D7	140	CD9	41	CWRT	28	PD1	207	FFD9	103
D6	141	CD8	42	CCYCST*	30	PD0	208	FFD8	105
D5	142	CD7	43	CRDY*	24	PA15	1	FFD7	106
D4	143	CD6	46			PA14	2	FFD6	107
D3	144	CD5	47	**Packet Memory Interface**		PA13	3	FFD5	108
D2	145	CD4	48			PA12	4	FFD4	109
D1	146	CD3	49	PD35	166	PA11	5	FFD3	110
D0	147	CD2	50	PD34	167	PA10	6	FFD2	111
A7	149	CD1	51	PD33	168	PA9	8	FFD1	112
A6	150	CD0	52	PD32	169	PA8	9	FFD0	113
A5	151	CA23	54	PD31	170	PA7	10	CELAVL*	94
A4	152	CA22	55	PD30	171	PA6	11	RDCLK	91
A3	153	CA21	56	PD29	173	PA5	12	RDEN	89
A2	154	CA20	57	PD28	174	PA4	13	XON	80
A1	155	CA19	58	PD27	175	PA3	14	**CBR Interface**	
A0	156	CA18	59	PD26	176	PA2	15		
WRT	119	CA17	60	PD25	177	PA1	16	CBRXMIT	117
DS*	120	CA16	61	PD24	178	PA0	17	CBRDONE	114
CS*	121	CA15	64	PD23	179	PREQ	162		
INTR*	129	CA14	65	PD22	180	PGRT	158	**Congestion Control Interface**	
RDY*	128	CA13	66	PD21	181	PCYCST*	163		
		CA12	67	PD20	182	PRDY*	159	CCDATA	87
		CA11	68	PD19	186	PLWADR	160	CCXFER	86
		CA10	69	PD18	187	PAMTCH	164	CCHLD	88
		CA9	70	PD17	188				
		CA8	71	PD16	189			**Miscellaneous Signals**	
		CA7	73	PD15	190				
		CA6	74	PD14	191			RST*	118
				PD13	192			CLK	124
				PD12	193			TE	21
				PD11	196			SI	20
				PD10	197			SO	157
								TEI	22

Table 3-17. Cross Reference by Pin Name for SARA-S

Table 3-17 gives a cross reference by pin name for SARA-S. The signals have been classified by the nature of the hardware interface with which they are associated, following the organization of the previous sections.

3.3.8 SARA-S Pin Configuration

Figure 3-17 shows a diagram of the 208-pin PQFP configuration for SARA-S

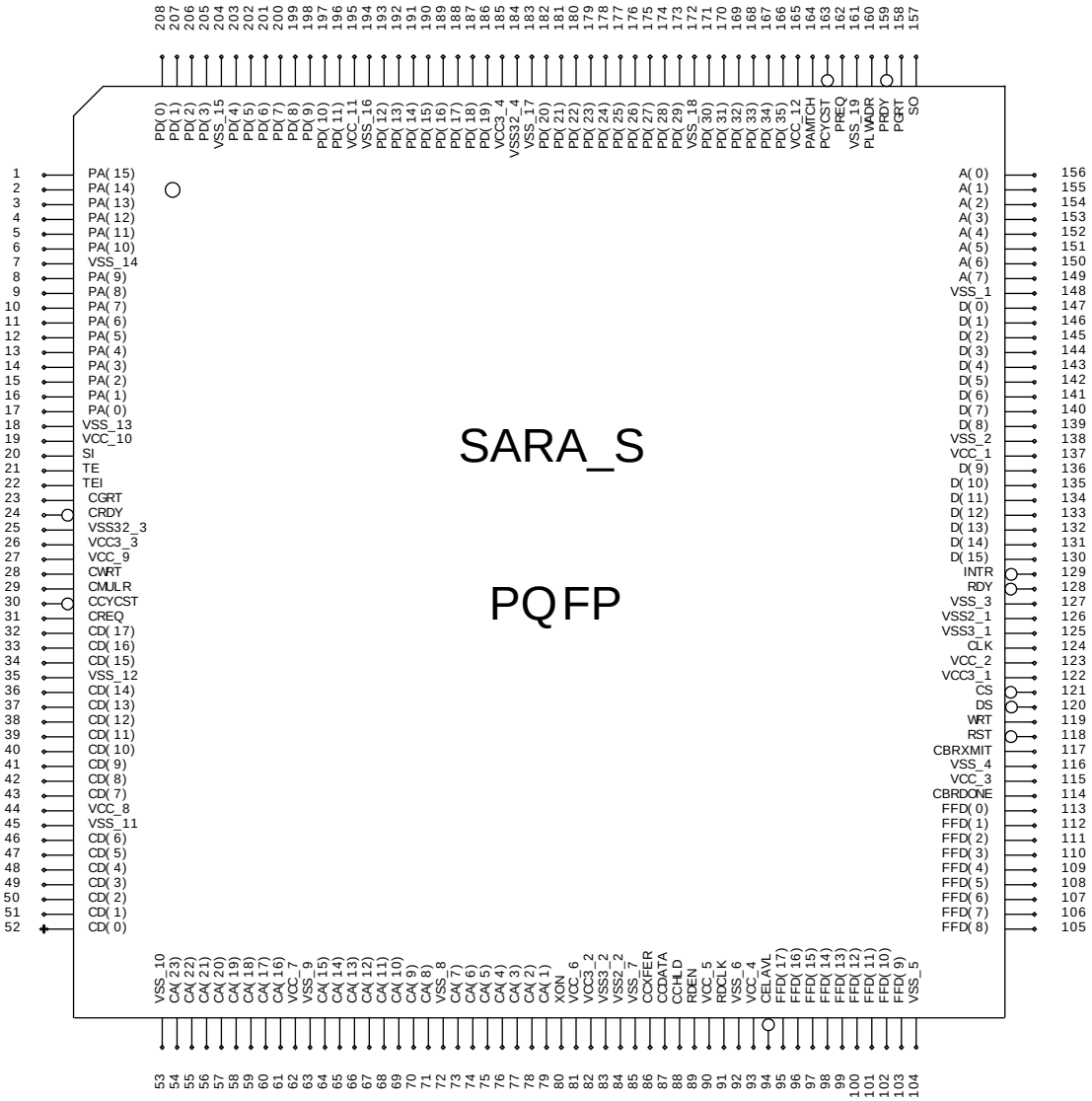


Figure 3-17. 208-pin PQFP Pin Configuration—SARA-S

Processor Interface		Control Memory Interface		Control Memory Interface (cont'd)		Packet Memory Interface (cont'd)		Cell Interface	
Signal	Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal	Pin
CS*	36	CREQ	95	CD12	89	PD35	200	FFRD3	110
DS*	37	CMULR	96	CD11	87	PD34	199	FFRD2	109
WRT	38	CGRT	101	CD10	86	PD33	198	FFRD1	108
RDY*	29	CCYCST*	97	CD9	85	PD32	197	FFRD0	107
A7	8	CRDY*	102	CD8	84	PD31	196	FFSL1	112
A6	7	CWRT	98	CD7	83	PD30	195	FFSL0	111
A5	6	CA23	71	CD6	82	PD29	192	FFD17	131
A4	5	CA22	70	CD5	81	PD28	191	FFD16	130
A3	4	CA21	69	CD4	80	PD27	190	FFD15	129
A2	3	CA20	68	CD3	79	PD26	189	FFD14	128
A1	2	CA19	67	CD2	78	PD25	188	FFD13	127
A0	1	CA18	66	CD1	73	PD24	187	FFD12	126
D15	27	CA17	65	CD0	72	PD23	186	FFD11	125
D14	26	CA16	64	**Packet Memory Interface**		PD22	185	FFD10	124
D13	25	CA15	62			PD21	184	FFD9	123
D12	24	CA14	61	PREQ	203	PD20	180	FFD8	122
D11	23	CA13	60	PGRT	207	PD19	179	FFD7	121
D10	22	CA12	59	PCYCST*	202	PD18	178	FFD6	120
D9	21	CA11	58	PRDY*	206	PD17	177	FFD5	119
D8	18	CA10	57	PLWADR	205	PD16	176	FFD4	118
D7	17	CA9	56	PAMTCH	201	PD15	175	FFD3	117
D6	16	CA8	55	PA15	156	PD14	174	FFD2	116
D5	15	CA7	54	PA14	155	PD13	173	FFD1	115
D4	14	CA6	53	PA13	154	PD12	172	FFD0	114
D3	13	CA5	50	PA12	153	PD11	169	CLAV	103
D2	12	CA4	49	PA11	152	PD10	168	FFMT	104
D1	11	CA3	48	PA10	151	PD9	167	FFLUSH	106
D0	10	CA2	47	PA9	149	PD8	166	FLSHDONE	105
INTR*	28	CA1	46	PA8	148	PD7	165	**Congestion Control Interface**	
Miscellaneous Signals		NC	45	PA7	147	PD6	164		
		CD17	94	PA6	146	PD5	163	CCHLD	40
CBRMT*	42	CD16	93	PA5	145	PD4	162	CCXFER	44
RST*	39	CD15	92	PA4	144	PD3	160	CCDATA	43
CLK	33	CD14	91	PA3	143	PD2	159		
TE	136	CD13	90	PA2	142	PD1	158		
SI	137			PA1	141	PD0	157		
SO	208			PA0	140				
TEI	135								

Table 3-18. Cross Reference by Pin Name for SARA-R

3.3.9 Cross Reference by Pin Name (SARA-R)

Table 3-18 gives a cross reference by pin name for SARA-R. The signals have been classified by the nature of the hardware interface with which they are associated, following the organization of the previous sections

3.3.10 SARA-R Pin Configuration

Figure 3-18 shows a diagram of the 208-pin PQFP configuration for SARA-R

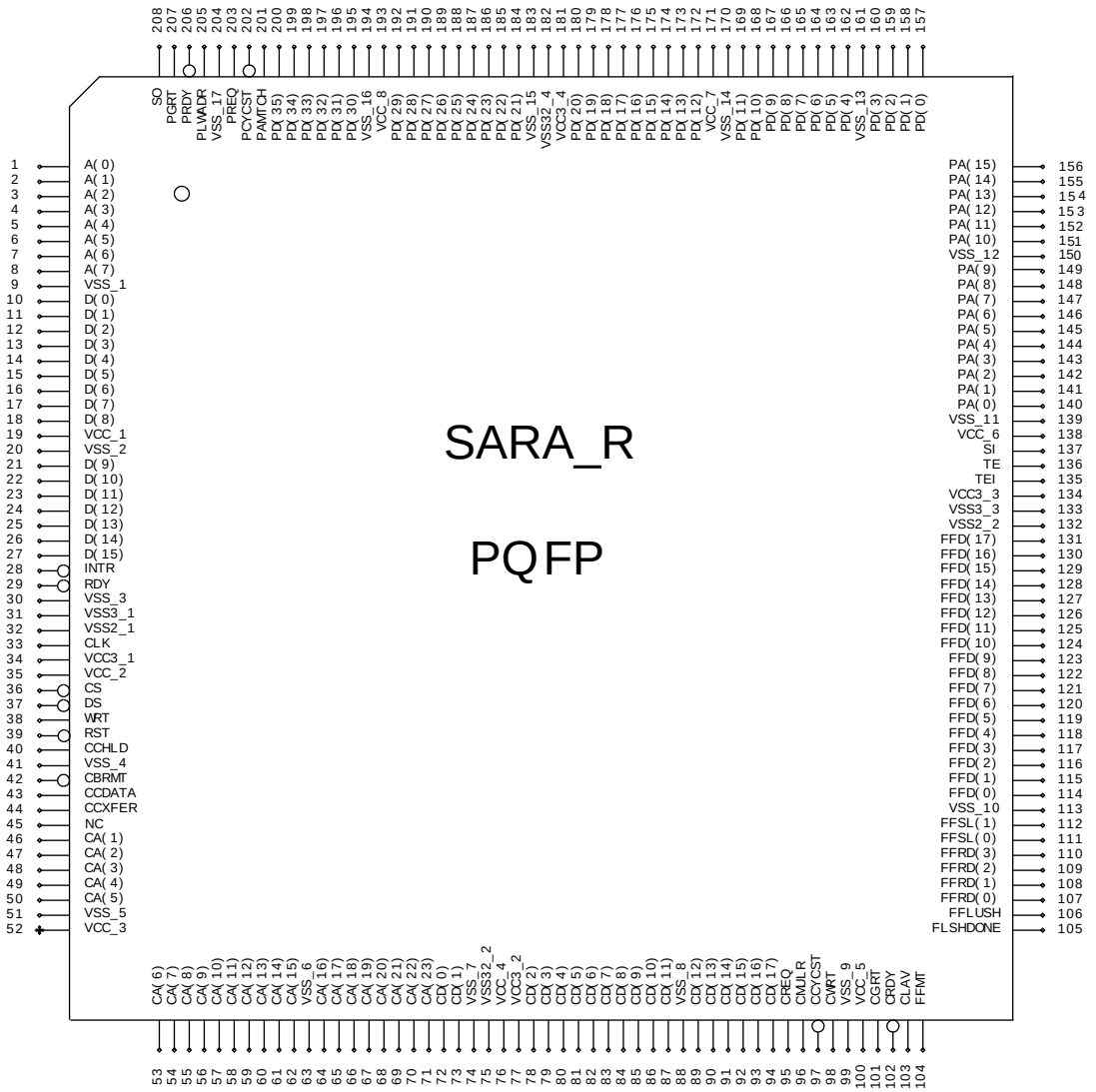
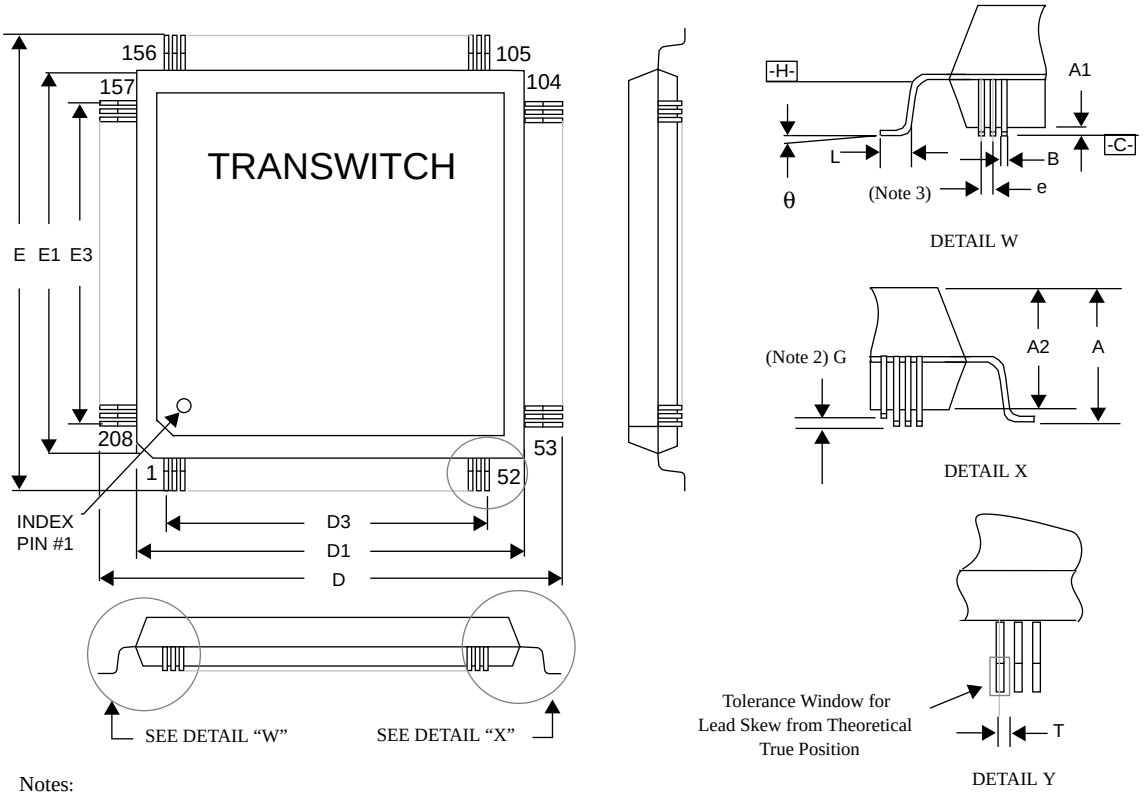


Figure 3-18. 208-pin PQFP Pin Configuration—SARA-R

3.3.11 SARA Packaging Dimensions

Figure 3-19 shows the dimensions of the SARA packaging.



Notes:

1. Controlling dimension - millimeter.
2. Coplanarity of all leads shall be within 0.1 mm (difference between the highest and lowest lead with seating plane -C- as reference).
3. Lead pitch determined at datum -H-.

Figure 3-19. SARA Packaging

A	Max	3.86	D1	Min	27.90	θ	Min	0º
A1	Min	0.25		Max	28.10		Max	5º
	Max	0.36	D3	Ref	25.50	L	Min	0.40
A2	Min	3.30	E	Min	30.40		Max	0.60
	Max	3.50		Max	30.80	G	Max	0.076
B	Min	0.18	E1	Min	27.90	T	Max	0.10
	Max	0.28		Max	28.10			
D	Min	30.40	E3	Ref	25.50			
	Max	30.80	e	BSC	0.50			

Table 3-19. SARA Packaging Dimensions

3.4 DC Characteristics

Based on the temperature and supply voltage given below, Table 3-20 shows the DC Characteristics which are applicable to the “A” and “B” functional versions of each of the SARA-S and SARA-R devices. Table 3-21 shows the corresponding capacitance characteristics.

Temperature, T_A 0 °C to 70 °C

Supply Voltage, V_{CC} 4.75 V to 5.25 V

Parameter Symbol	Parameter Description	Test Conditions	Functional Version	Min.	Max.	Unit
V_{IL}	Input Low Voltage		“A” and “B”		0.8	V
V_{IH}	Input High Voltage		“A” and “B”	2.0		V
V_{OL}	Output Low Voltage	I_{OL}=Max	“A” and “B”		0.4	V
V_{OH}	Output High Voltage	I_{OH}=I_{OL}	“A” and “B”	2.4		V
I_{OL}(open drain outputs)	Output Low Current		“A” and “B”		8.0	mA
I_{OL}(all other outputs)	Output Low Current		“A” and “B”		4.0	mA
I_{OH}	Output High Current		“A” and “B”		I_{OL}	mA
I_{OZ1}	Output Leakage Current	0.4V <V_{OUT} <V_{CC}	“A” and “B”	-10	10	µA
I_{OZ2}	Bi-Directional Tri-State I/O Output Leakage Current	V_{OH} =V_{SS} or V_{CC}	“A” “B”	-350 -200	10 10	µA µA
I_{IX}	Input Leakage Current	0V <V_{IN} <V_{CC}	“A” “B”	-35 -10	350 200	µA µA
I_{CC}	Power Supply Current	V_{CC}=Max., f(CLK) as shown for Version	“A” @ 20 MHz “B” @ 33 MHz		400 280	mA mA

Table 3-20. SARA DC Characteristics Over Commercial Operating Ranges

Parameter Symbol	Parameter Description	Test Conditions	Functional Version	Min.	Max.	Unit
C_{IN}	Input Pins[a]		“A” and “B”		10	pF
$C_{I/O}$	Bidirectional pins		“A” and “B”		10	pF
Output Drive Capability	Microprocessor Outputs		“A” and “B”		100	pF
	All Other Outputs		“A” and “B”		50	pF

Table 3-21. SARA Capacitance Characteristics

a. Unused input pins must be tied to inactive state.

3.5 AC Characteristics

3.5.1 Switching Characteristics

Tables 3-22A and 3-22B show the switching characteristics of both the SARA-R and SARA-S timings for the “A” and “B” functional version devices, respectively, over commercial operating conditions. These similar tables include “A” and “B” background overlays to help avoid inadvertent use of the table that does not correspond to the device version under consideration. The significance of the timing parameter symbols is illustrated in the signal timing diagrams provided in section 3.5.2. The notation NA indicates “Not Applicable”.

Symbol	Parameter	Mln.	Max	Units
T1	Clock Period	50		ns
T2	Clock High	20		ns
T3	Clock Low	20		ns
T4	CCHLD to CLK Setup (SARA-R)	9		ns
T5	CLK to CCHLD Hold (SARA-R)	2		ns
T6	CLK to CCHLD Valid (SARA-S)		24	ns
T7	CCXFER to CLK Setup (SARA-S)	0		ns
T8	CLK to CCXFER Hold (SARA-S)	4		ns
T9	CLK to CCXFER Valid (SARA-R)		22	ns
T10	CCDATA to CLK Setup (SARA-S)	0		ns
T11	CLK to CCDATA Hold (SARA-S)	5		ns
T12	CLK to CCDATA Valid (SARA-R)		22	ns
T13	CBRXMIT to CLK Setup (SARA-S)	Async.		ns
T14	CLK to CBRXMIT Hold (SARA-S)	Async.		ns
T15	CLK to CBRDONE Valid (SARA-S)		19	ns
T16	CLK to CBRMT* Valid (SARA-R)		19	ns
T21	WRT, A(7:0), D(15:0) to CS*, DS* Setup	0		ns
T22	CS*, DS* to WRT, A(7:0), D(15:0) Hold	0		ns
T23	CS*, DS* to RDY* Low		(3*T1 + 21)	ns
T24	CS*, DS* to RDY* Deasserted		14	ns
T25	CS*, DS* to D(15:0) Enabled (Read)	0		ns
T26	D(15:0) to RDY* Setup		(T1 - 32)	ns

Table 3-22A. Switching Characteristics for “A” Versions over Commercial Operating Conditions

Symbol	Parameter	MIn.	Max	Units
T27	CS*, DS* to D(15:0) Deasserted (Read)		20	ns
T28	CS*, DS* High	(1.5 * T1)		ns
T29	INTR* High	(4 *T1 - 24)		ns
T30	RST* Low	(3*T1 + 5)		ns
T31	CLAV to CLK Setup (SARA-R)	0		ns
T32	CLK to CLAV Hold (SARA-R)	5		ns
T33	CLK to FFRD(0) Valid (SARA-R)		20	ns
T34	FFMT to CLK Setup (SARA-R)	8		ns
T35	CLK to FFMT Hold (SARA-R)	4		ns
T36	FFD(17:0) to CLK Setup (SARA-R)	0		ns
T37	CLK to FFD(17:0) Hold (SARA-R)	3		ns
T38	CLK to FFLUSH Valid (SARA-R)		25	ns
T39	FLSHDONE to CLK Setup (SARA-R)	4		ns
T40	CLK to FLSHDONE Hold (SARA-R)	5		ns
T41	RDCLK to CELAVL* Valid (SARA-S)		17	ns
T42	RDEN to RDCLK Setup (SARA-S)	13		ns
T43	RDCLK to RDEN Hold (SARA-S)	2		ns
T44	RDCLK to FFD(17:0) Valid (SARA-S)		25	ns
T45	RDCLK Period	T1		ns
T46	RDCLK Low	T3		ns
T47	RDCLK High	T2		ns
T51	CLK to CREQ Valid		24	ns
T52	CLK to CMULR Valid		24	ns
T53	CGRT to CLK Setup	1		ns
T54	CLK to CGRT Hold	4		ns
T55	CLK to CCYCST* Valid		20	ns
T56	CRDY* to CLK Setup	0		ns
T57	CLK to CRDY* Hold	4		ns
T58	CLK to CWRT Valid		24	ns

Table 3-22A. Switching Characteristics for “A” Versions over Commercial Operating Conditions

Symbol	Parameter	MIn.	Max	Units
T59	CLK to CA(23:0) Asserted	0		ns
T60	CLK to CA(23:0) Valid		19	ns
T61	CLK to CA(23:0) Deasserted		26	ns
T62	CD(17:0) to CLK Setup (Read)	0		ns
T63	CLK to CD(17:0) Hold (Read)	3		ns
T64	CLK to CD(17:0) Asserted (Write)	0		ns
T65	CLK to CD(17:0) Valid (Write)		22	ns
T66	CLK to CD(17:0) Deasserted (Write)		22	ns
T71	CLK to PREQ Valid		22	ns
T72	PGRT to CLK Setup	8		ns
T73	CLK to PGRT Hold	3		ns
T74	CLK to PCYCST* Valid		20	ns
T75	PRDY* to CLK Setup	1		ns
T76	CLK to PRDY* Hold	4		ns
T77	CLK to PA(15:0) Asserted	0		ns
T78	CLK to PA(15:0) Valid		24	ns
T79	CLK to PA(15:0) Deasserted		24	ns
T80	CLK to PD(31:0) Asserted (PM Address)		27	ns
T81	CLK to PD(31:0) Deasserted		27	ns
T82	PD(35:0) to CLK Setup (SARA-S Read)	3		ns
T83	CLK to PD(35:0) HOld (SARA-S Read)	4		ns
T84	PLWADR to PA(15:0) Valid		18	ns
T85	CLK to PAMTCH Valid		27	ns
T86	CLK to PD(35:0) Valid (SARA-R Write)		27	ns
T87	CLK to PD(35:0) Deasserted (SARA-R Write)		27	ns

Table 3-22A. Switching Characteristics for “A” Versions over Commercial Operating Conditions

Symbol	Parameter	SARA-S (TXC-05501-BCPQ)		SARA-R (TXC-05601-BCPQ)		Units
		Min	Max	Min	Max	
T1	Clock Period	30		30		ns
T2	Clock High	12		12		ns
T3	Clock Low	12		12		ns
T4	CCHLD to CLK Setup (SARA-R)	NA	NA	6		ns
T5	CLK to CCHLD Hold (SARA-R)	NA	NA	2		ns
T6	CLK to CCHLD Valid (SARA-S)		16	NA	NA	ns
T7	CCXFER to CLK Setup (SARA-S)	0		NA	NA	ns
T8	CLK to CCXFER Hold (SARA-S)	4		NA	NA	ns
T9	CLK to CCXFER Valid (SARA-R)	NA	NA		16	ns
T10	CCDATA to CLK Setup (SARA-S)	0		NA	NA	ns
T11	CLK to CCDATA Hold (SARA-S)	5		NA	NA	ns
T12	CLK to CCDATA Valid (SARA-R)	NA	NA		16	ns
T13	CBRXMIT to CLK Setup (SARA-S)	Async.		NA	NA	ns
T14	CLK to CBRXMIT Hold (SARA-S)	Async.		NA	NA	ns
T15	CLK to CBRDONE Valid (SARA-S)		16	NA	NA	ns
T16	CLK to CBRMT* Valid (SARA-R)	NA	NA		16	ns
T21	WRT, A(7:0), D(15:0) to CS*, DS* Setup	0		0		ns
T22	CS*, DS* to WRT, A(7:0), D(15:0) Hold	0		0		ns
T23	CS*, DS* to RDY* Low		(3*T1 + 21)		(3*T1 + 21)	ns
T24	CS*, DS* to RDY* Deasserted		14		14	ns
T25	CS*, DS* to D(15:0) Enabled (Read)	0		0		ns
T26	D(15:0) to RDY* Setup		(T1 - 18)		(T1 - 18)	ns
T27	CS*, DS* to D(15:0) Deasserted (Read)		20		20	ns
T28	CS*, DS* High	(1.5 * T1)		(1.5 * T1)		ns
T29	INTR* High	(4 *T1 - 24)		(4 *T1 - 24)		ns
T30	RST* Low	(3*T1 + 5)		(3*T1 + 5)		ns
T31	CLAV to CLK Setup (SARA-R)	NA	NA	2		ns
T32	CLK to CLAV Hold (SARA-R)	NA	NA	5		ns
T33	CLK to FFRD(0) Valid (SARA-R)	NA	NA		14	ns
T34	FFMT to CLK Setup (SARA-R)	NA	NA	4		ns

Table 3-22B. Switching Characteristics for “B” Versions over Commercial Operating Conditions

Symbol	Parameter	SARA-S (TXC-05501-BCPQ)		SARA-R (TXC-05601-BCPQ)		Units
		Min	Max	Min	Max	
T35	CLK to FFMT Hold (SARA-R)	NA	NA	2		ns
T36	FFD(17:0) to CLK Setup (SARA-R)	NA	NA	8		ns
T37	CLK to FFD(17:0) Hold (SARA-R)	NA	NA	3		ns
T38	CLK to FFLUSH Valid (SARA-R)	NA	NA		15	ns
T39	FLSHDONE to CLK Setup (SARA-R)	NA	NA	2		ns
T40	CLK to FLSHDONE Hold (SARA-R)	NA	NA	5		ns
T41	RDCLK to CELAVL* Valid (SARA-S)		11	NA	NA	ns
T42	RDEN to RDCLK Setup (SARA-S)	8		NA	NA	ns
T43	RDCLK to RDEN Hold (SARA-S)	2		NA	NA	ns
T44	RDCLK to FFD(17:0) Valid (SARA-S)		11	NA	NA	ns
T45	RDCLK Period	T1		T1		ns
T46	RDCLK Low	T3		T3		ns
T47	RDCLK High	T2		T2		ns
T51	CLK to CREQ Valid		13		13	ns
T52	CLK to CMULR Valid		13		13	ns
T53	CGRT to CLK Setup	4		8		ns
T54	CLK to CGRT Hold	4				ns
T55	CLK to CCYCST* Valid		11		11	ns
T56	CRDY* to CLK Setup	5		8		ns
T57	CLK to CRDY* Hold	4				ns
T58	CLK to CWRT Valid		11		12	ns
T59	CGRT to CA(23:0) Asserted		10		10	ns
T60	CLK to CA(23:0) Valid		9		10	ns
T61	CLK to CA(23:0) Deasserted		15		15	ns
T62	CD(17:0) to CLK Setup (Read)	5		8		ns
T63	CLK to CD(17:0) Hold (Read)	3		3		ns
T64	CLK to CD(17:0) Asserted (Write)		11		13	ns
T65	CLK to CD(17:0) Valid (Write)		11		13	ns
T66	CLK to CD(17:0) Deasserted (Write)		15		15	ns
T71	CLK to PREQ Valid		12		12	ns

Table 3-22B. Switching Characteristics for “B” Versions over Commercial Operating Conditions

Symbol	Parameter	SARA-S (TXC-05501-BCPQ)		SARA-R (TXC-05601-BCPQ)		Units
		Min	Max	Min	Max	
T72	PGRT to CLK Setup	8		8		ns
T73	CLK to PGRT Hold	3		3		ns
T74	CLK to PCYCST* Valid		10		10	ns
T75	PRDY* to CLK Setup	6		4		ns
T76	CLK to PRDY* Hold	4		4		ns
T77	CLK to PA(15:0) Asserted		13		11	ns
T78	CLK to PA(15:0) Valid		13		11	ns
T79	CLK to PA(15:0) Deasserted		15		15	ns
T80	CLK to PD(31:0) Asserted (PM Address)		12		11	ns
T81	CLK to PD(31:0) Deasserted		15		15	ns
T82	PD(35:0) to CLK Setup (SARA-S Read)	8		NA	NA	ns
T83	CLK to PD(35:0) Hold (SARA-S Read)	4		NA	NA	ns
T84	PLWADR to PA(15:0) Valid		13		13	ns
T85	CLK to PAMTCH Valid		17		17	ns
T86	CLK to PD(35:0) Valid (SARA-R Write)	NA	NA		11	ns
T87	CLK to PD(35:0) Deasserted (SARA-R Write)	NA	NA		15	ns

Table 3-22B. Switching Characteristics for “B” Versions over Commercial Operating Conditions

3.0.1 Interface Signal Timings

Figures 3-20A through 3-28A, and Figures 3-20B through 3-28B, for the “A” and “B” functional version devices, respectively, show the timings for the congestion control interface, processor, cell interface, control memory interface, and packet memory interface of both the SARA-R and SARA-S. These nine pairs of similar figures include “A” and “B” background overlays to help avoid inadvertent use of the figure that does not correspond to the device version under consideration.

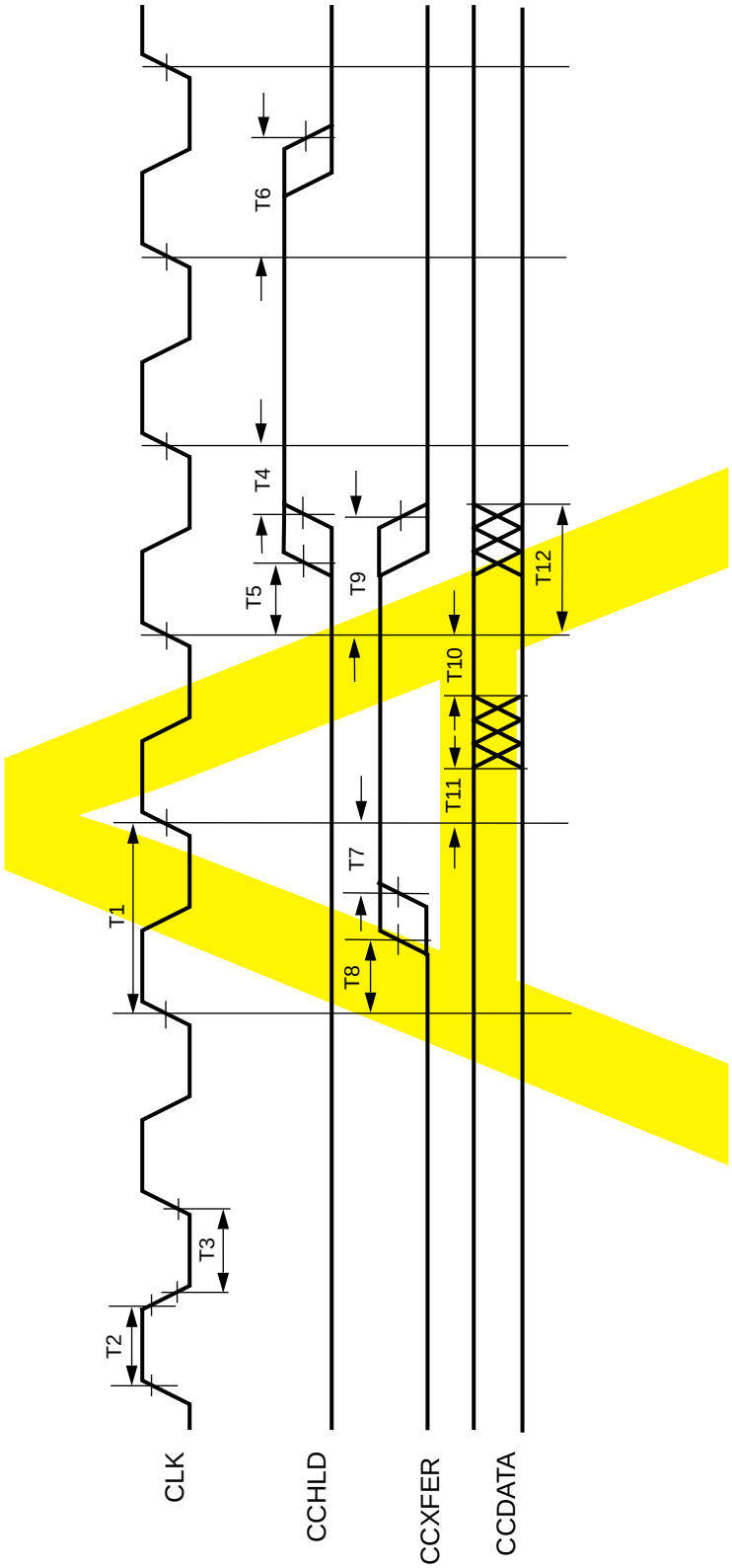


Figure 3-20A. Congestion-Control Interface Signals Between the SARA-S and SARA-R (“A” Versions)

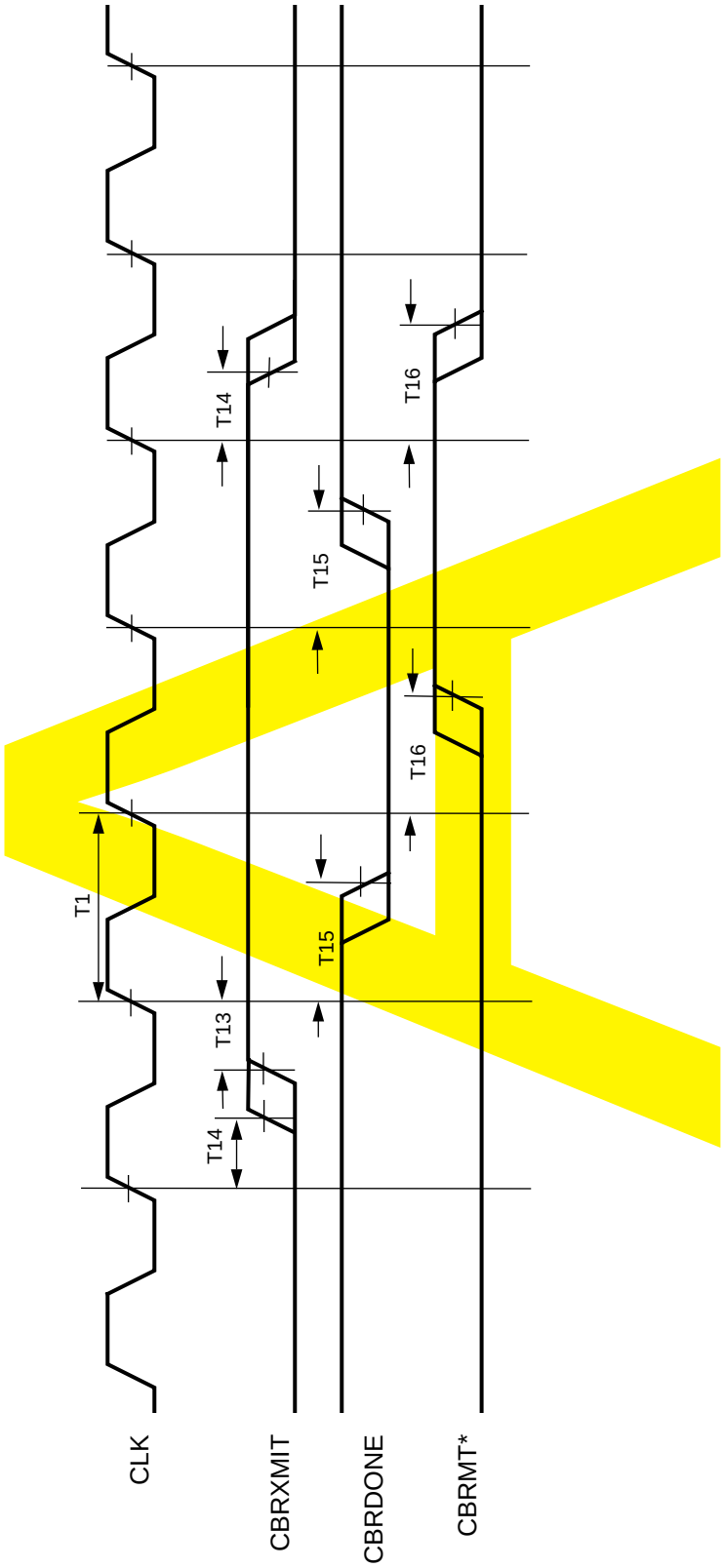


Figure 3-21A. Constant Bit Rate Interface Signals of SARA-S and SARA-R (“A” Versions)

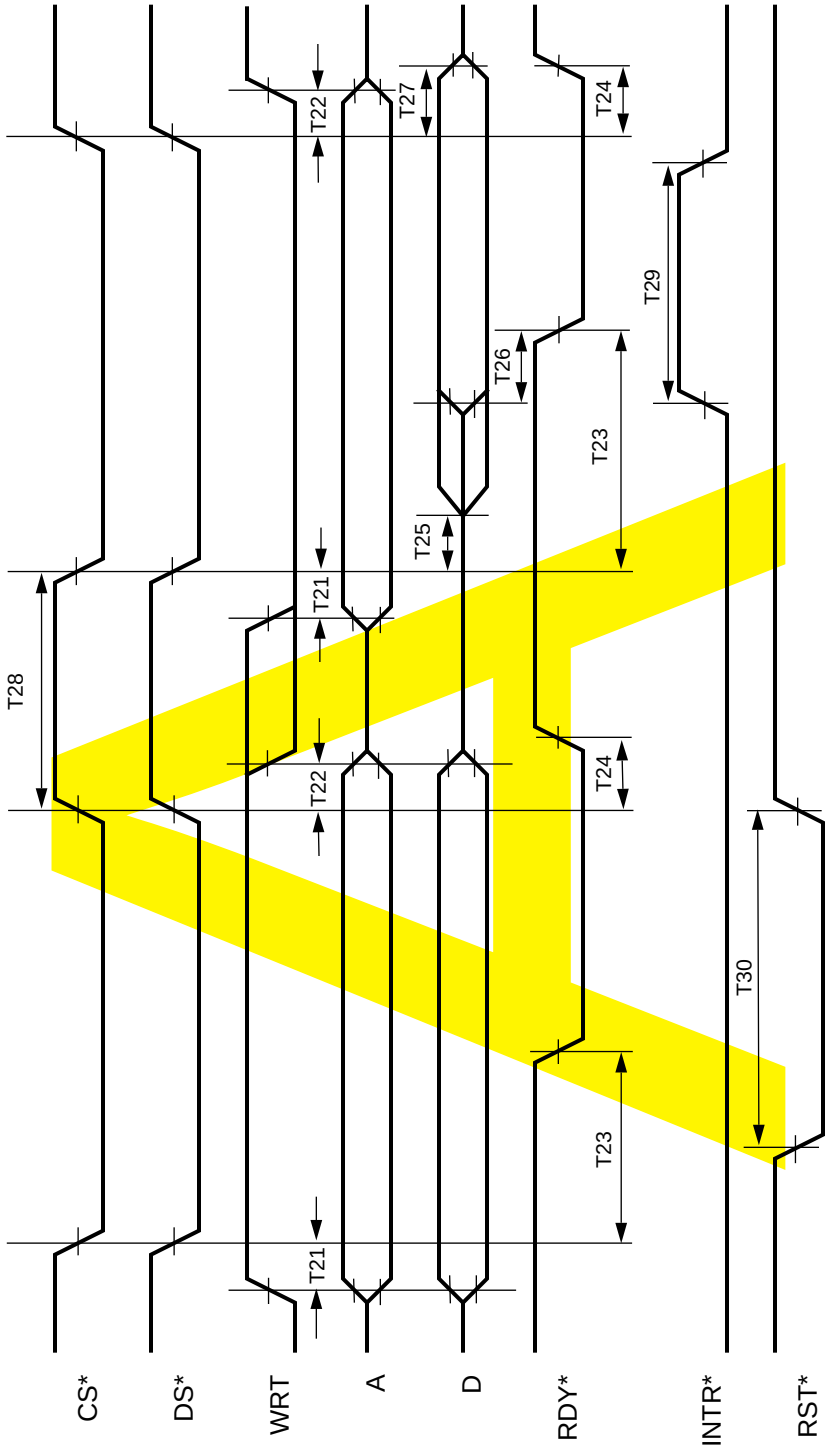


Figure 3-22A. Processor-Interface Signal Timing for “A” Versions

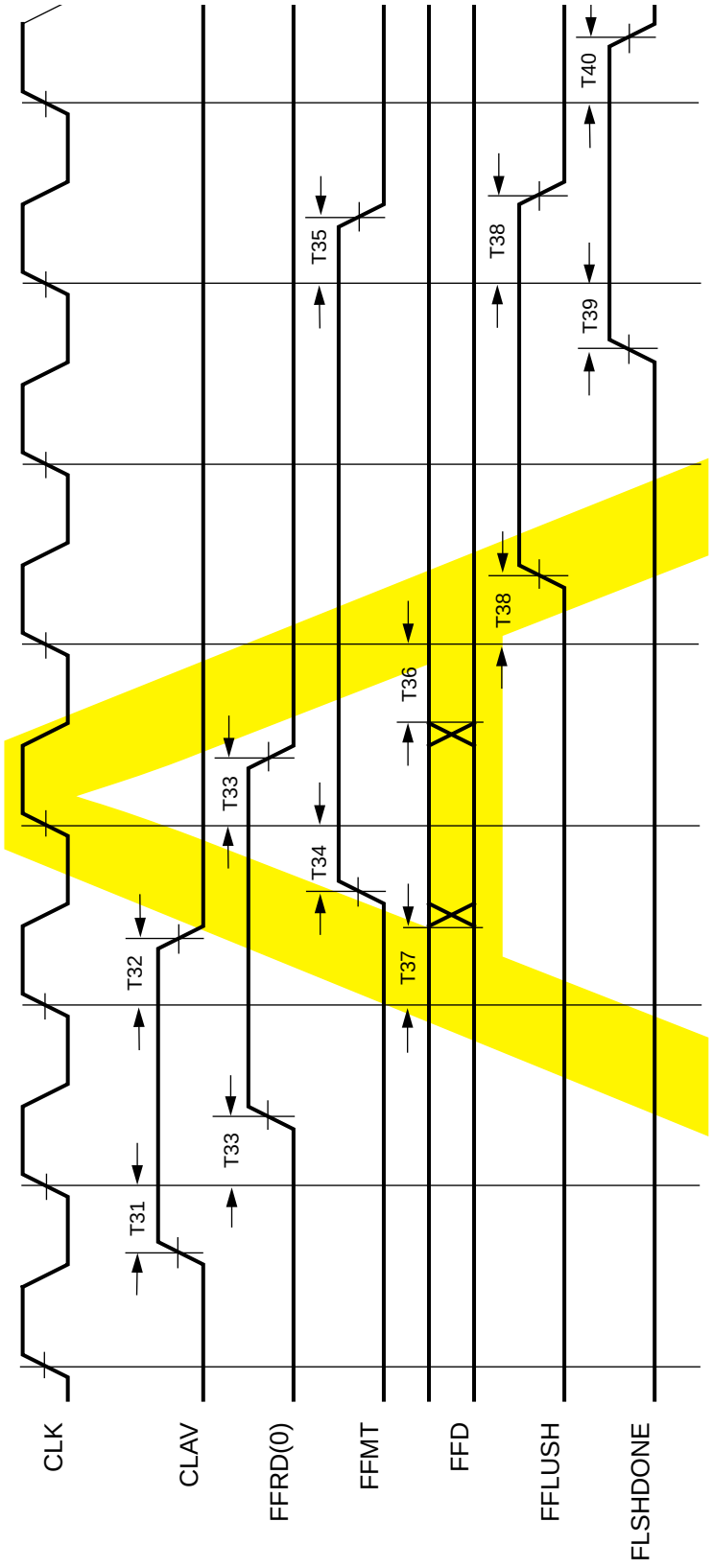


Figure 3-23A. Cell-Interface Signals for the SARA-R (“A” Version)

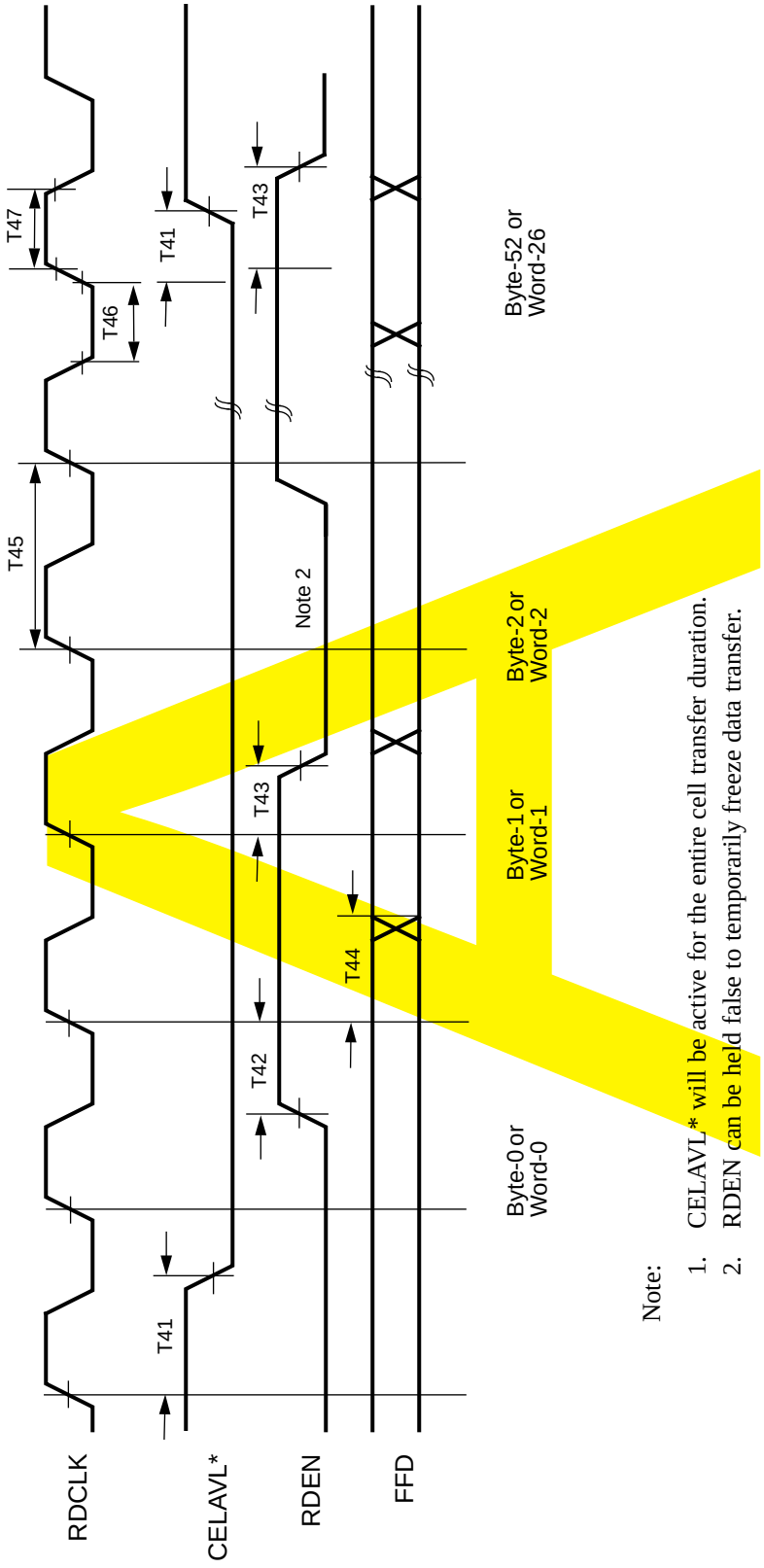
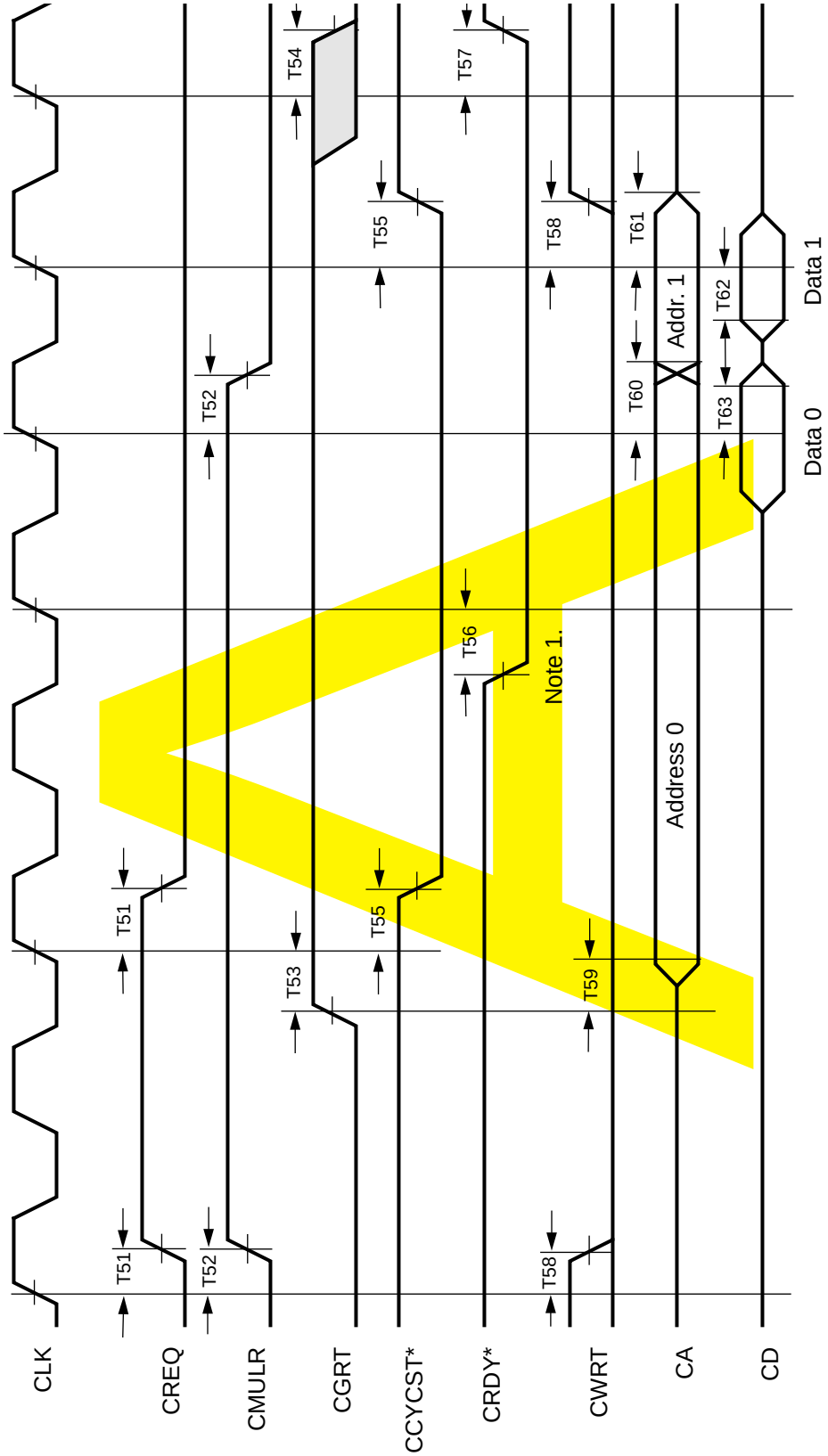


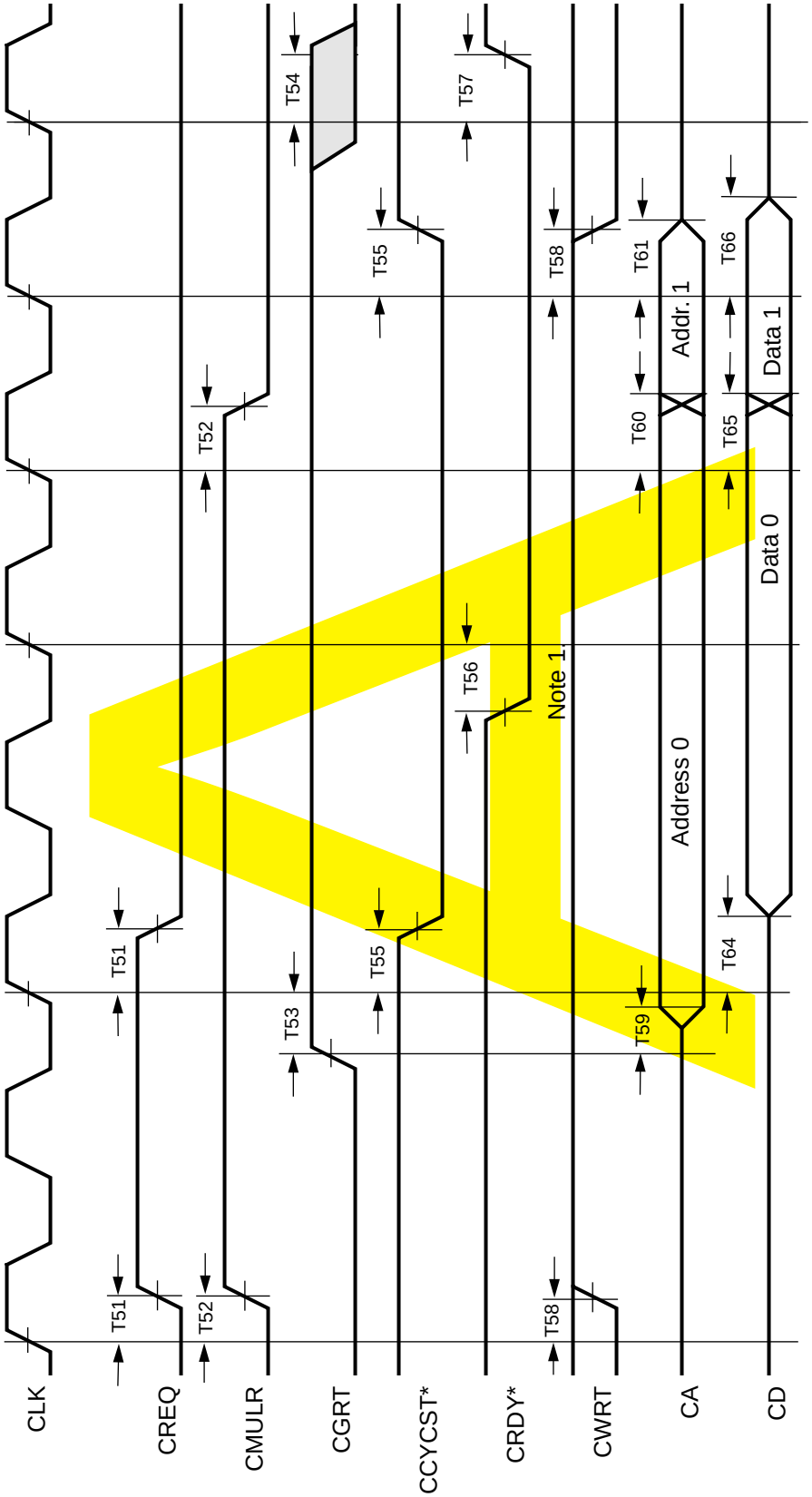
Figure 3-24A. Cell-Interface Signals for the SARA-S (“A” Version)



Note:

1. Data transfer does not occur on the very first clock cycle when CRDY becomes active.
2. Mode settings are:CM_WAIT_EN=1 and CM_EARLY_WR=0.

Figure 3-25A. Control Memory Interface Signal Timing-Read Operation for “A” Versions



Note:

1. Data transfer does not occur on the very first clock cycle when CRDY becomes active.
2. Mode settings are:CM_WAIT_EN=1 and CM_EARLY_WR=0.

Figure 3-26A. Control Memory Interface Signal Timings-Write Operation for “A” Versions

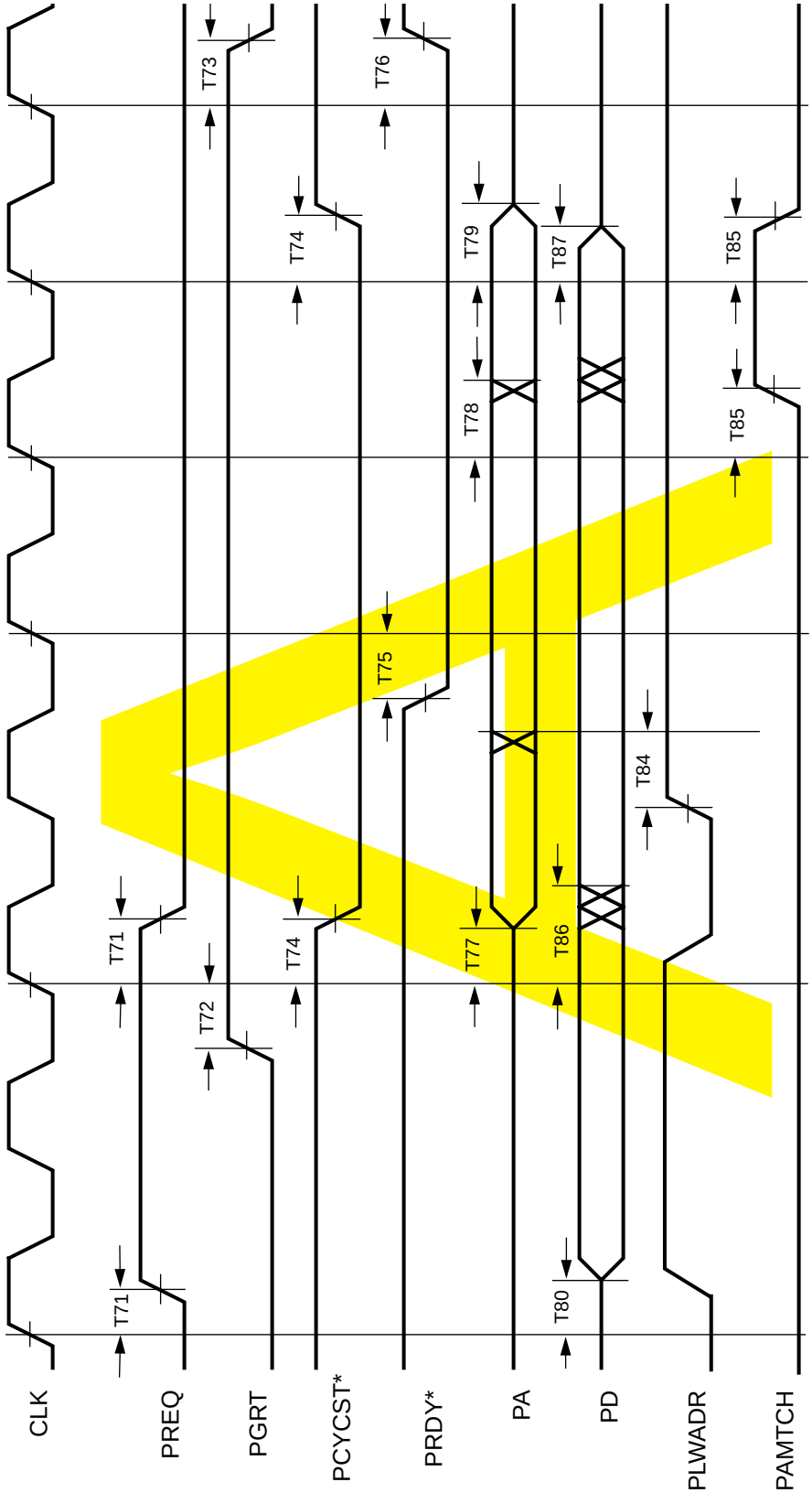


Figure 3-27A. Packet Memory Interface Signal Timings-Write Operation for the SARA-R (“A” Version)

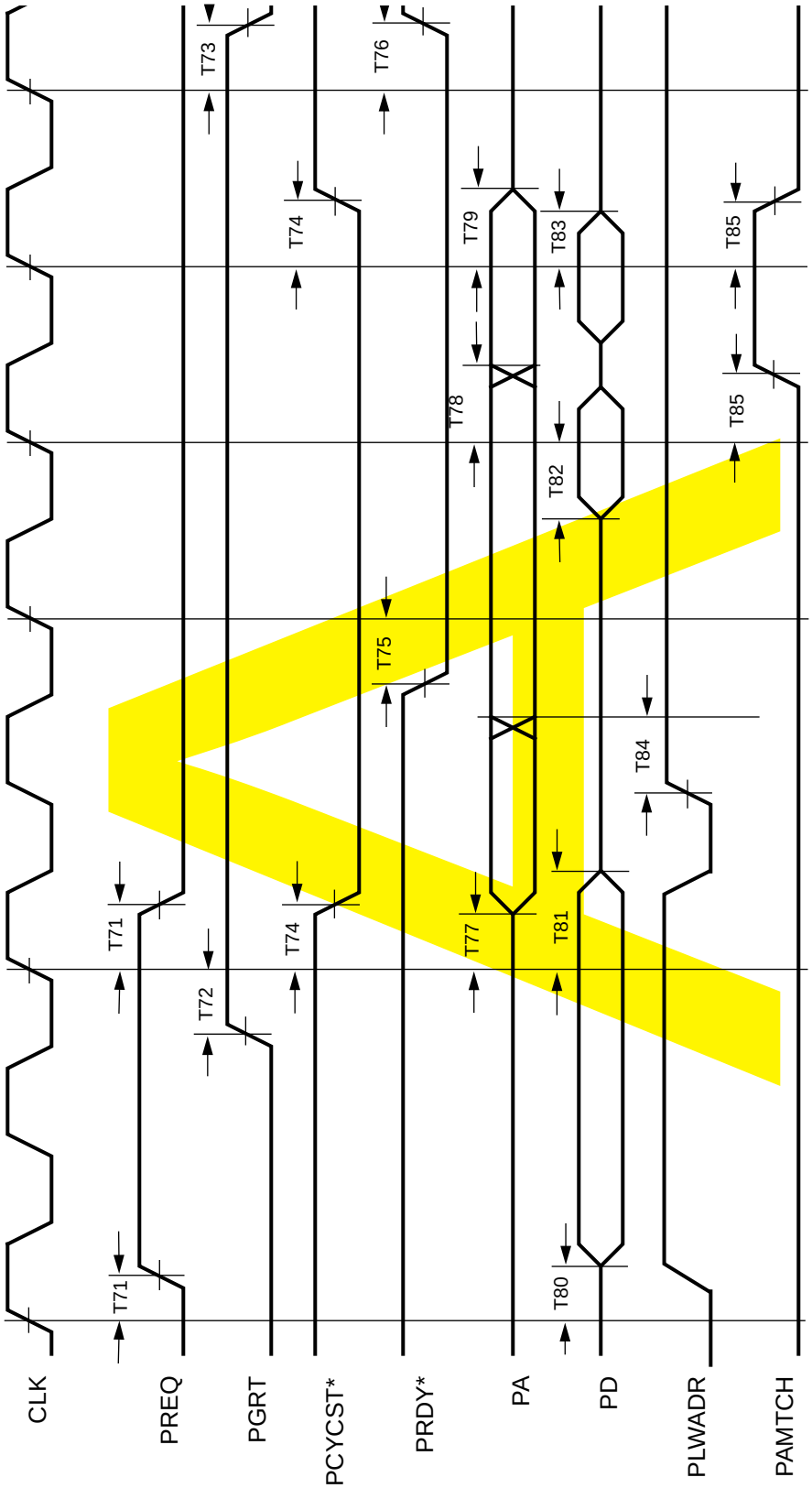


Figure 3-28A. Packet Memory Interface Signal Timings-Read Operation for the SARA-S (“A” Version)

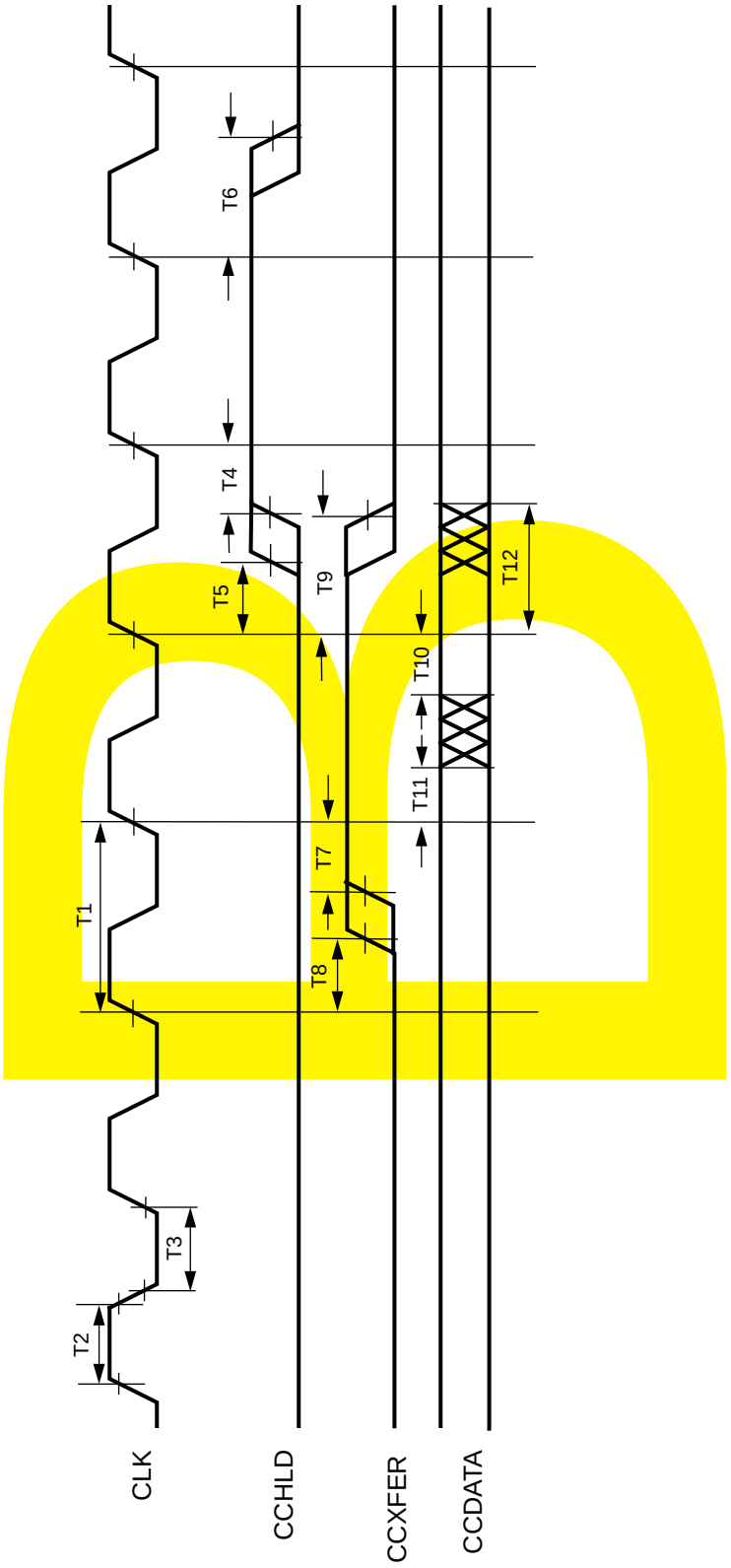


Figure 3-20B. Congestion-Control Interface Signals Between the SARA-S and SARA-R (“B” Versions)

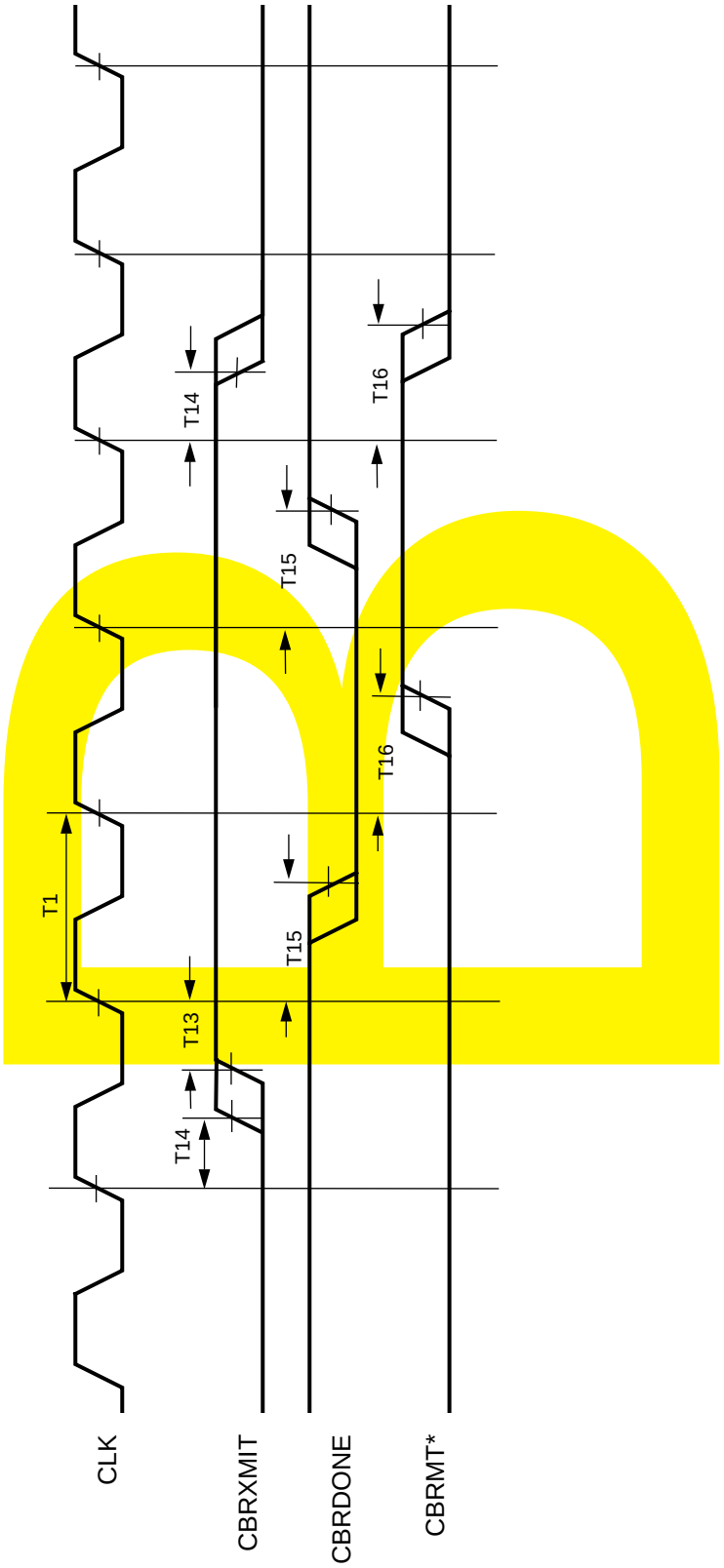


Figure 3-21B. Constant Bit Rate Interface Signals of SARA-S and SARA-R (“B” Versions)

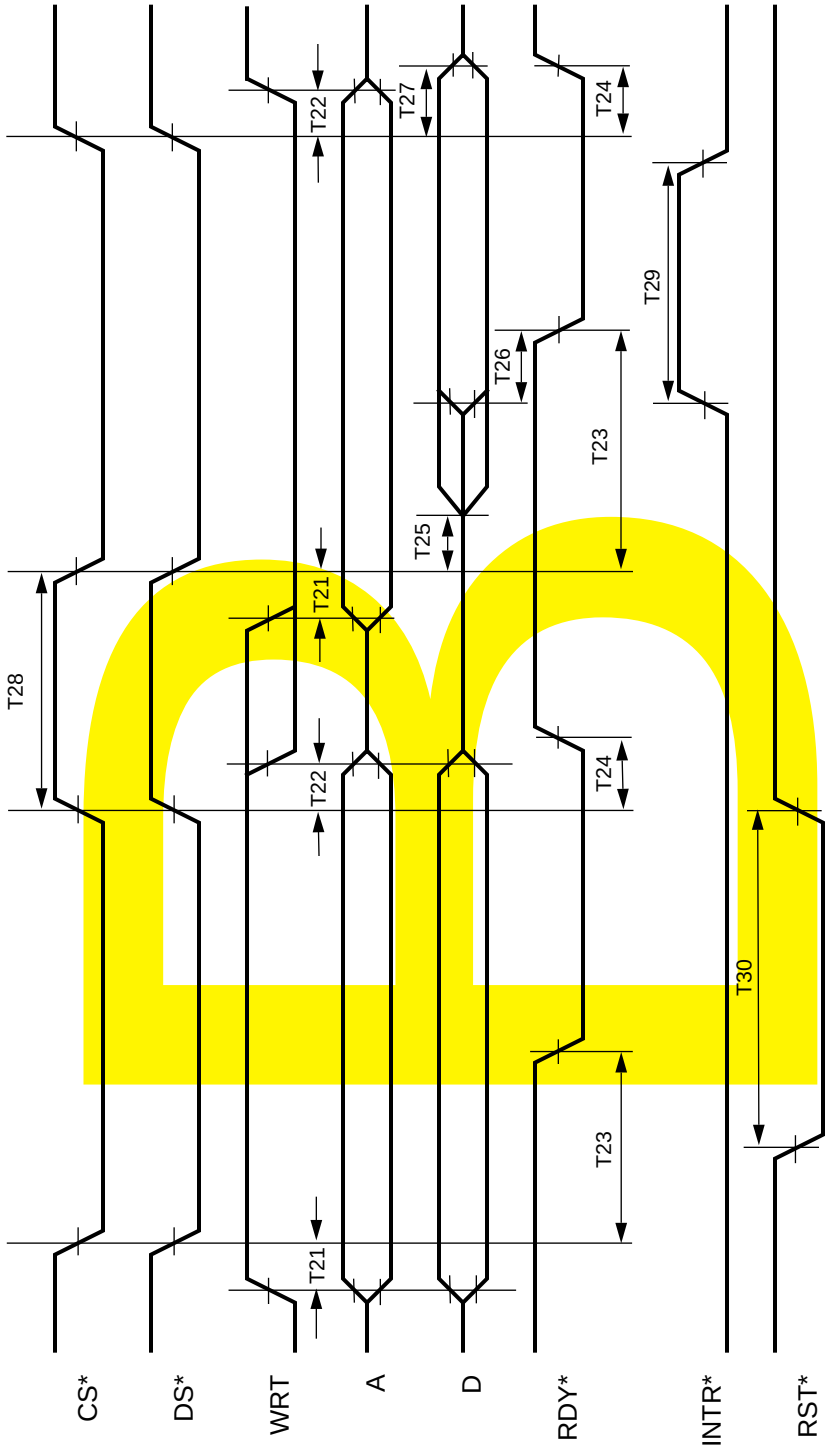


Figure 3-22B. Processor-Interface Signal Timing for “B” Versions

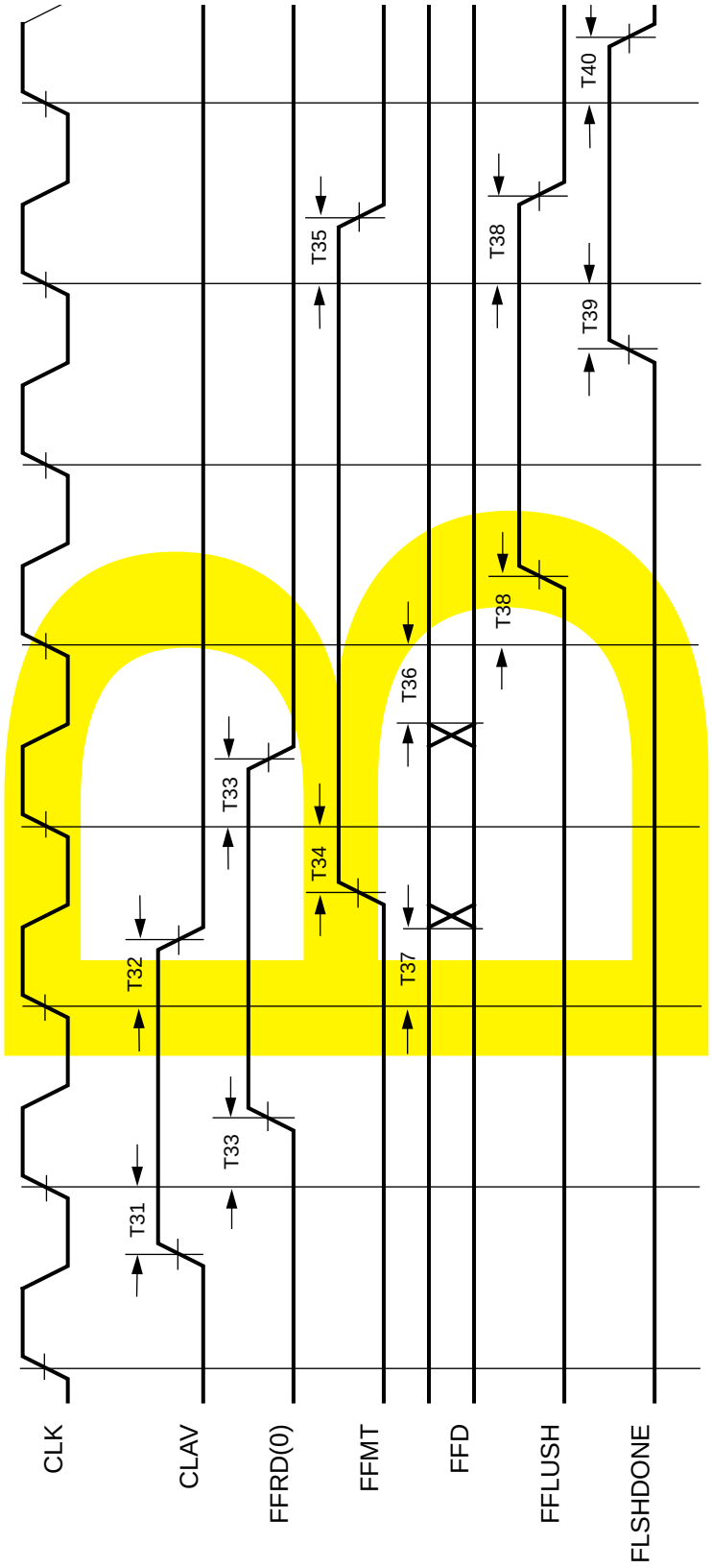


Figure 3-23B. Cell-Interface Signals for the SARA-R (“B” Version)

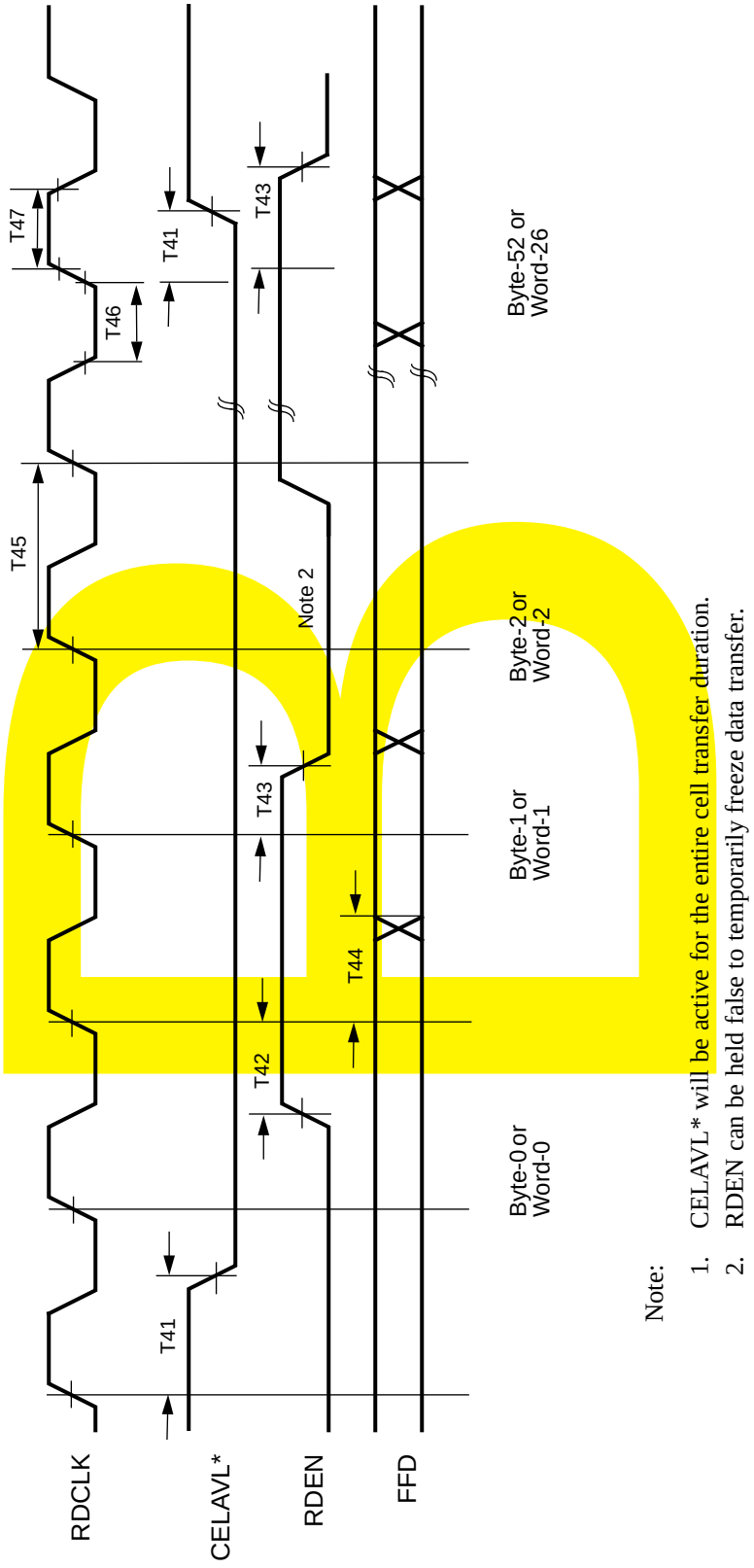
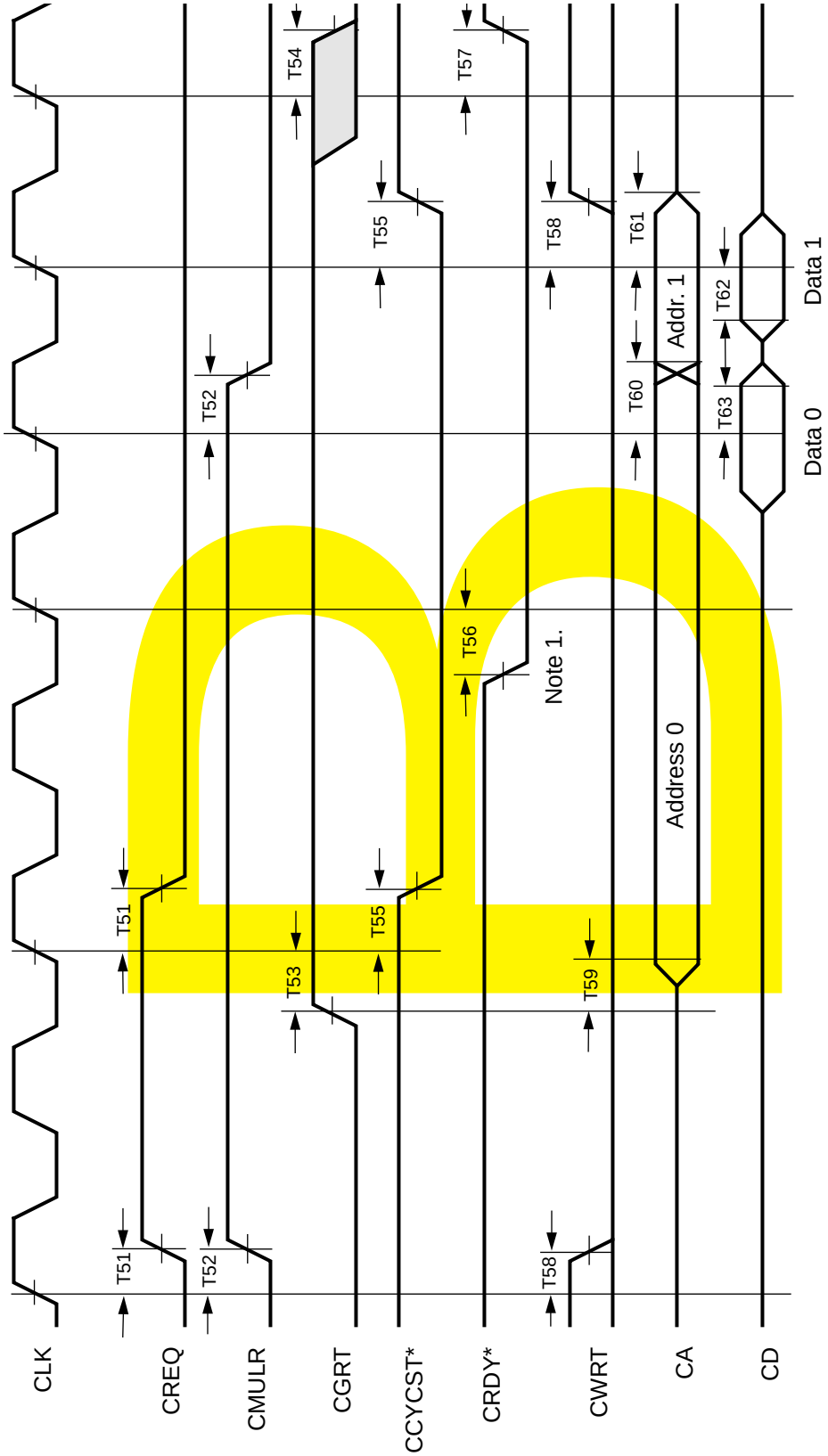
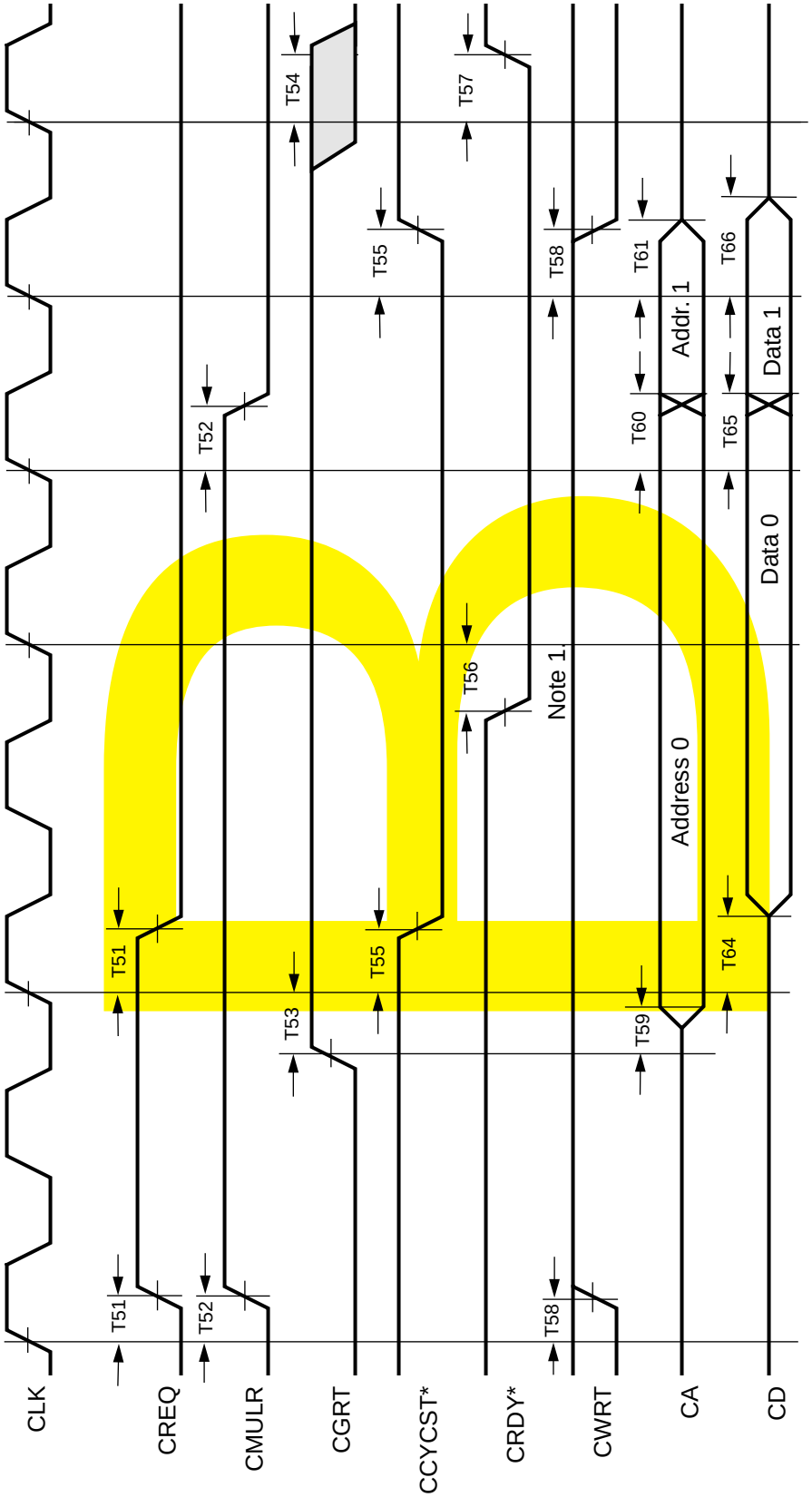


Figure 3-24B. Cell-Interface Signals for the SARA-S (“B” Version)



Note:
1. Data transfer does not occur on the very first clock cycle when CRDY becomes active.
2. Mode settings are:CM_WAIT_EN=1 and CM_EARLY_WR=0.

Figure 3-25B. Control Memory Interface Signal Timing-Read Operation for “B” Versions



Note:

1. Data transfer does not occur on the very first clock cycle when CRDY becomes active.
2. Mode settings are:CM_WAIT_EN=1 and CM_EARLY_WR=0.

Figure 3-26B. Control Memory Interface Signal Timings-Write Operation for “B” Versions

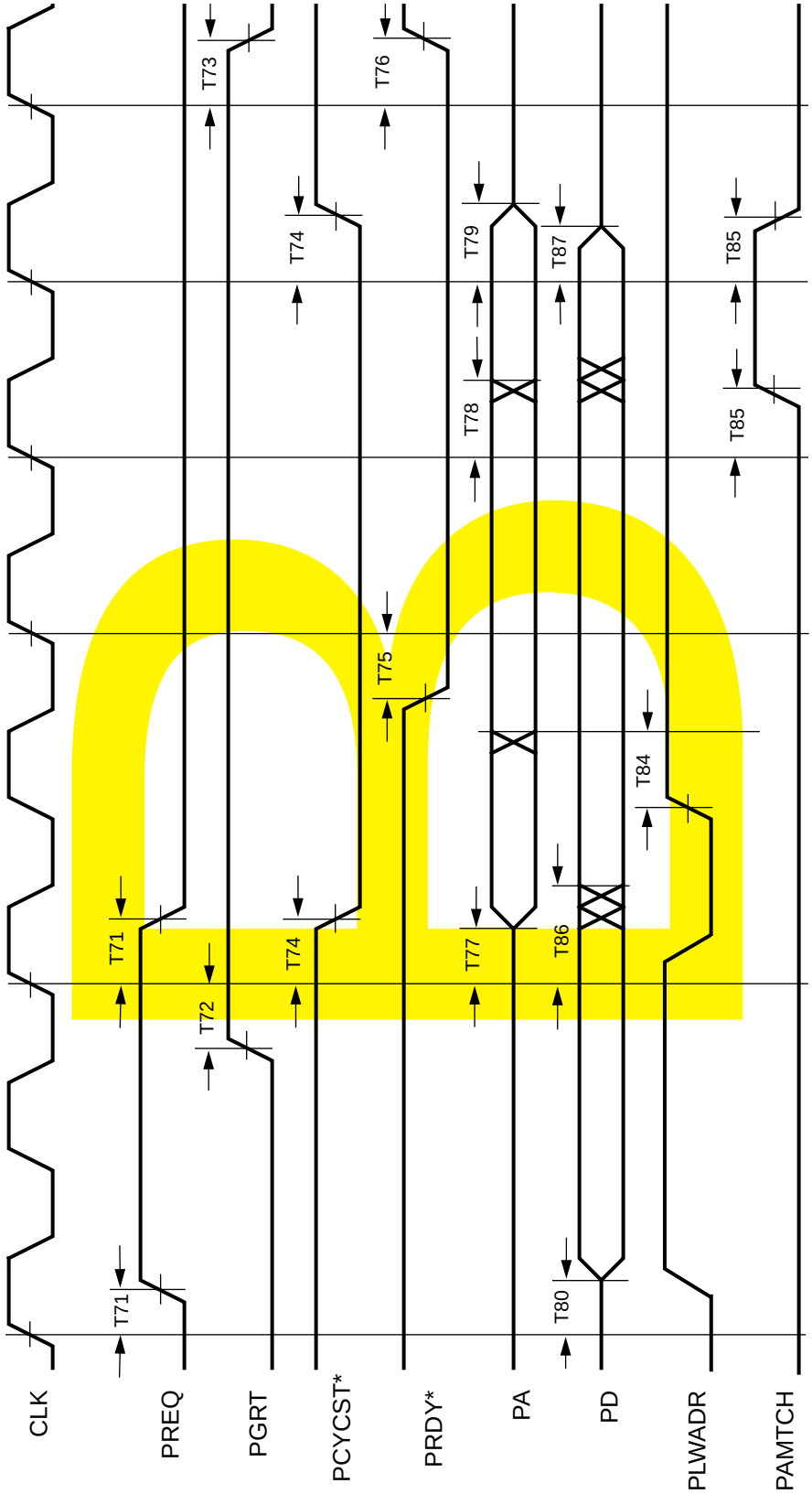


Figure 3-27B. Packet Memory Interface Signal Timings-Write Operation for the SARA-R (“B” Version)

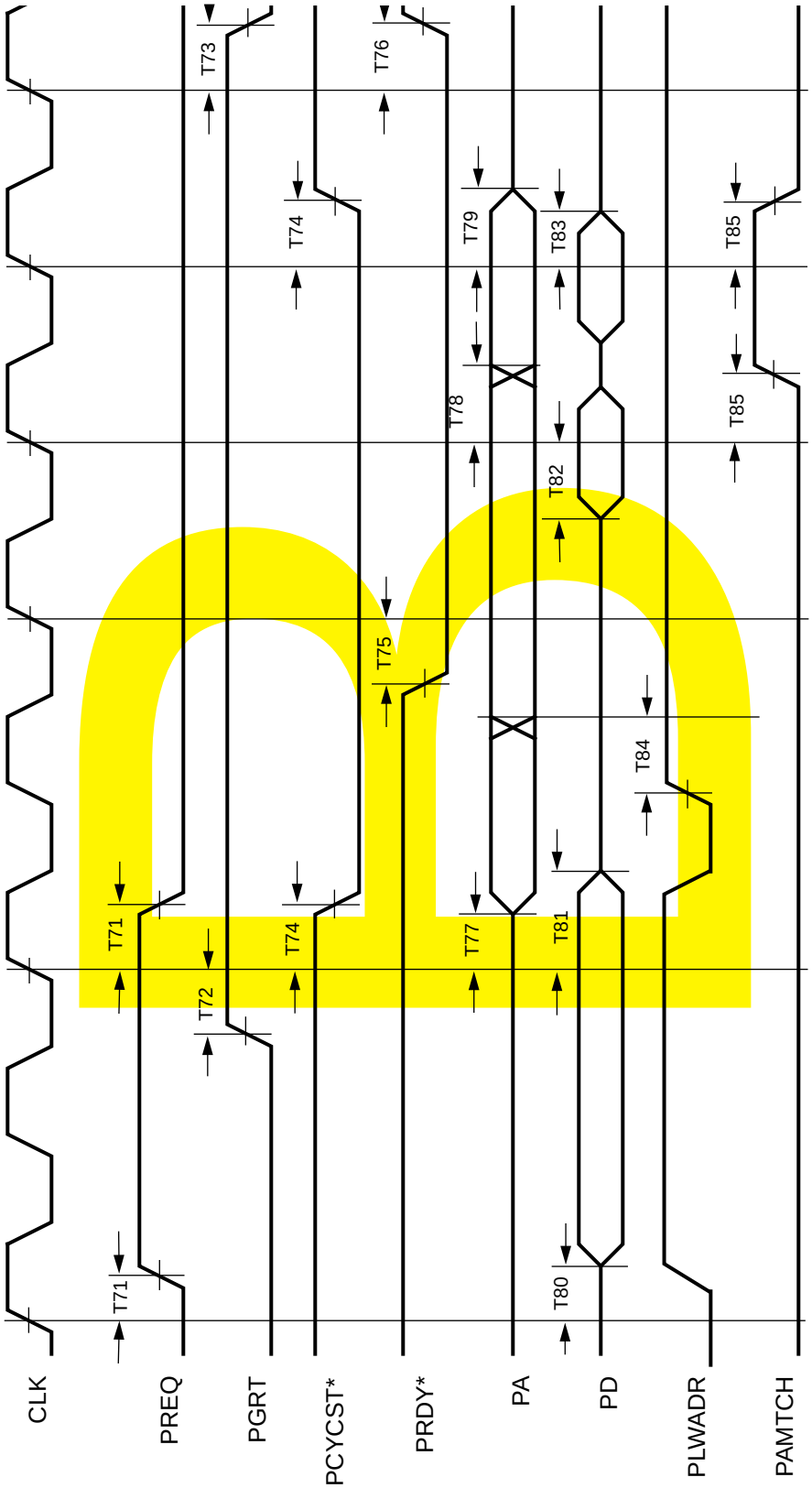


Figure 3-28B. Packet Memory Interface Signal Timings-Read Operation for the SARA-S (“B” Version)

Chapter 4. SARA-S Register Descriptions

This chapter serves as a software reference for the SARA-S chipset. Detailed descriptions of the processor/host registers and the queues are included.

All SARA-S registers are accessed as 16 bits with bit 0 as the least significant bit (LSB). All reserved and unused bits in these registers must be set to “0” for proper operation unless otherwise mentioned. Table 4-1 and Table 4-2 show the SARA-S Processor Registers used for chip setup and the SARA-S Diagnostic Registers respectively. The diagnostics registers are used mainly in debugging environments and are not required to be programmed for normal operation. The Reset Value column in the tables specify default reset/power-up value of the registers.

Register Name	Addr. (Hex)	Description	Type[1]	Reset Value
RQ_REG_B_0	00	Rate Register, Bank B, Queue 0	R/W	0000H
RQ_REG_B_1	01	Rate Register, Bank B, Queue 1	R/W	0000H
RQ_REG_B_2	02	Rate Register, Bank B, Queue 2	R/W	0000H
RQ_REG_B_3	03	Rate Register, Bank B, Queue 3	R/W	0000H
RQ_REG_A_0	04	Rate Register, Bank A, Queue 0	R/W	0000H
RQ_REG_A_1	05	Rate Register, Bank A, Queue 1	R/W	0000H
RQ_REG_A_2	06	Rate Register, Bank A, Queue 2	R/W	0000H
RQ_REG_A_3	07	Rate Register, Bank A, Queue 3	R/W	0000H
COMMAND	17	Command Register	W/O	N/A
CBR_ADDR_HI	20	DMA start address (high) for CBR traffic	R/W	0000H
CBR_ADDR_LO	21	DMA start address (low) for CBR traffic	R/W	0000H
PM_ADDR_MTCH	22	Packet Memory Address Match Register	R/CW	0000H
PRQ_ST_ADR	30	Packet Ready Queue Starting Address	R/CW	0000H
PRQ_ED_ADR	31	Packet Ready Queue Ending Address	R/CW	0000H
PRQ_RD_PTR	32	Packet Ready Queue read pointer	R/CW	0000H
PRQ_WR_PTR	33	Packet Ready Queue write pointer	R/W	0000H
TCQ_ST_ADR	34	Transmit Complete Queue Starting Address	R/CW	0000H
TCQ_ED_ADR	35	Transmit Complete Queue Ending Address	R/CW	0000H
TCQ_RD_PTR	36	Transmit Complete Queue read pointer	R/W	0000H
TCQ_WR_PTR	37	Transmit Complete Queue write pointer	R/CW	0000H

Table 4-1. SARA-S Internal Registers

Register Name	Addr. (Hex)	Description	Type[1]	Reset Value
QUEUE_BASE	40	Base address for Packet Ready and Transmit Complete Queues	R/CW	0000H
DESC_BASE	41	Base address for Descriptor Table	R/CW	0000H
VC_LKUP_BASE	42	Base address for ATM header/VC tables	R/CW	0000H
MODE_REG_0	45	Mode Register 0	R/W	0000H
MODE_REG_1	46	Mode Register 1	R/W	0000H
INTR_STATUS_REG	47	Interrupt Status Register	R/O	0800H
MASK_REG	48	Mask Register	R/W	FFFFH
CELL_CTR_HI	49	Total cells transferred counter (high); auto clear on read	R/O	0000H
CELL_CTR_HI_NC	C9	Total cells transferred counter (high); Does not auto clear on read.	R/O	0000H
CELL_CTR_LO	4A	Total cells transferred counter (low); auto clear on read.	R/O	0000H
CELL_CTR_LO_NC	CA	Total cells transferred counter (low); Does not auto clear on read.	R/O	0000H
STATE_REG	4B	State Register	R/O	F100H

Table 4-1. SARA-S Internal Registers

1. R/W Read and Write
   R/O Read Only
   W/O Write Only
   R/CW Read and Controlled Write. Writable only when the mode register of the chip has control write mode enabled.

Register Name	Addr. (hex)	Description	R/W[1]	Reset Value
RQ_B_0_SDN	50	Bank B, rate queue-0 start desc. number	R/CW	0000H
RQ_B_1_SDN	51	Bank B, rate queue-1 start desc. number	R/CW	0000H
RQ_B_2_SDN	52	Bank B, rate queue-2 start desc. number	R/CW	0000H
RQ_B_3_SDN	53	Bank B, rate queue-3 start desc. number	R/CW	0000H
RQ_A_0_SDN	54	Bank A, rate queue-0 start desc. number	R/CW	0000H
RQ_A_1_SDN	55	Bank A, rate queue-1 start desc. number	R/CW	0000H
RQ_A_2_SDN	56	Bank A, rate queue-2 start desc. number	R/CW	0000H
RQ_A_3_SDN	57	Bank A, rate queue-3 start desc. number	R/CW	0000H
CURR_DESC_NUM	58	Contains the current descriptor number being accessed	R/CW	0000H
CC_LABEL	59	Congestion control cell VCI	R/CW	0000H

Table 4-2. SARA-S Diagnostic Registers

1. R/CW Read and Controlled Write. Writable only when the mode register of the chip has control write mode enabled.

4.1 Mode Registers

Figure 4-1 and Figure 4-2 show the SARA-S Mode Register 0 and Mode Register 1 data bits. These two registers are used to setup the operating modes of the chip. Reserved and unused bits must be programmed as “0” for proper device operation, unless otherwise mentioned.

4.1.1 Mode Register 0

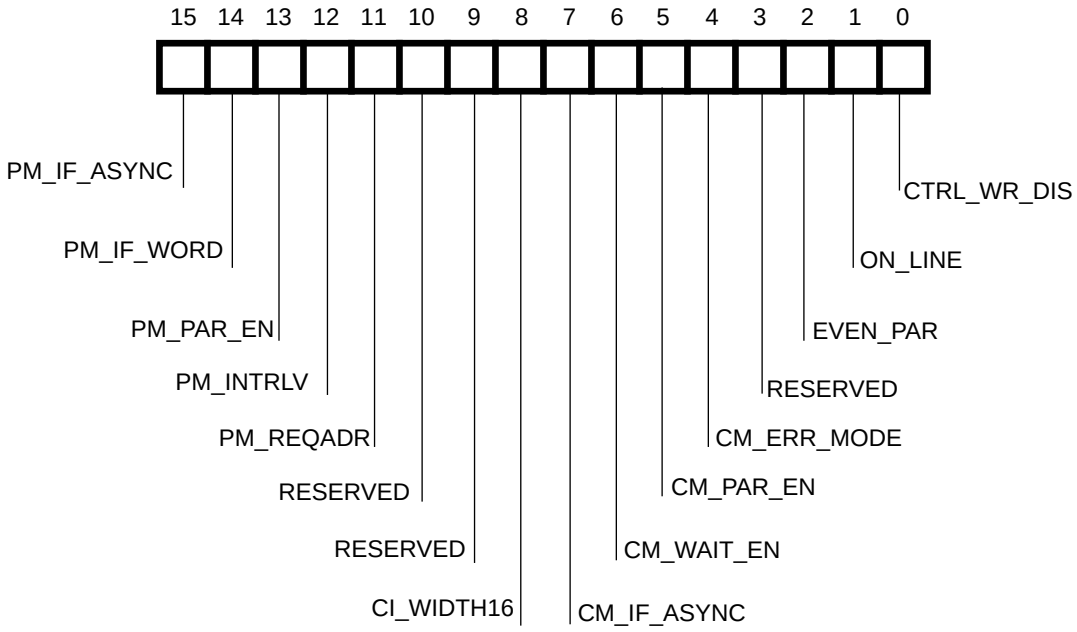


Figure 4-1. Mode Register 0 Data Bits

The following describes the individual bits in SARA-S Mode Register 0:

Bit 15: PM_IF_ASYNC:

0 - Packet memory interface is synchronous

1 - Packet memory interface is asynchronous

In synchronous mode, data is transferred during packet memory cycles on every clock (when PRDY* input pin is TRUE). Either a burst of data or a single cycle data can be transferred under external PRDY* input control.

In asynchronous mode, the bus is relinquished after every transfer and will be re-requested. Data transfer is effectively through a complete PREQ/PGRT/PCYCST*/ PRDY* handshake.

Bit 14: PM_IF_WORD:

0 - Packet memory interface is 32 bits wide

1 - Packet memory interface is 16 bits wide

This bit programs the size of the packet memory interface, selecting between a 16- or a 32-bit wide data bus.

Bit 13: PM_PAR_EN:

0 - Disable packet memory parity checking

1 - Enable packet memory parity checking

This bit is used to enable parity checking on packet memory data. When set to “1”, either odd or even parity is checked based on the EVEN_PAR bit.

Bit 12: PM_INTRLV:

0 - Packet memory is not interleaved

1 - Packet memory is interleaved

This bit is used to select the packet memory address bits asserted on PA(15:0) pins. This bit can be programmed appropriately to ease the memory system design. Note that the input pin PLWADR determines whether the high or low bits of the packet memory address are presented on the bus.

When the low bits are being requested (PLWADR = 1), address (15:0) is always presented on the address bus. When the high bits are requested (PLWADR = 0), the address bits presented on the address bus depend on PM_INTRLV bit and the width of the packet memory interface as follows:

- If the PM_INTRLV bit is 0, the high order address bits (23:8) are presented on the address bus.
- If the PM_INTRLV bit is 1 and the packet memory interface is 32 bits wide, the address bits (23:9,2) are presented on the address bus.
- If the PM_INTRLV bit is 1 and the packet memory interface is 16-bits wide, the address bits (23:9,1) are presented on the address bus.

Bit 11: PM_REQADR:

0 - Do not present packet memory address on the data lines PD(31:0) during packet memory bus requests.

1 - Present packet memory address on the datalines PD(31:0) during packet memory bus requests.

If this bit is set to “1”, the packet memory address, bits 31:0 are presented on PD(31:0) data lines during memory bus request cycles for burst memory accesses.

These interactions are described inTable 4-3.

PREQ	PGRT	PLWADR	PD(31:0)
1	0	0	3-state
1	0	1	Address
1	1	X	Data

Table 4-3. Interactions of PREQ, PGRT, and PLWADR with Packet Data Bus PD(31:0)—SARA-S

Bit (10:9): RESERVED:

These bits must be programmed to “0” for proper device operation.

Bit 8: CI_WIDTH16:

0 - Cell interface is eight bits wide.

1 - Cell interface is 16 bits wide.

This bit selects between an eight-bit or a 16-bit wide cell interface.

Bit 7: CM_IF_ASYNC:

0 - Control memory interface is synchronous.

1 - Control memory interface is asynchronous.

In synchronous mode, data is transferred during control memory cycles on every clock (when CRDY* input pin is true).

In asynchronous mode, the bus is relinquished after every transfer and will be re-requested. Data transfer is effectively through a complete CREQ/CGRT/CCYCST*/ CRDY* handshake.

Bit 6: CM_WAIT_EN:

0 - Ignore CRDY* input.

1 - Data transfer subject to CRDY* protocol.

This bit determines if the external CRDY* pin is sensed by the SARA-S chip. This bit must be set to '1' when the control memory is in synchronous interface mode and is ignored when the control memory is programmed to be in asynchronous mode.

Bit 5: CM_PAR_EN:

0 - Disable control memory parity checking

1 - Enable control memory parity checking

This bit is used to enable parity checking on control memory data. When set to “1”, either odd or even parity is checked based on the EVEN_PAR bit.

Bit 4: CM_ERR_MODE:

0 - Stop the state machine gracefully on control memory parity error

1 - Continue cell processing on control memory parity error

This bit determines the action taken by the SARA-S chip when a parity error is encountered in control memory data. When set to “0”, the SARA-S state machine stops further segmentation pending a reset (hard reset, soft reset or state machine reset). This will prevent possible actions by the chip that may mask any soft memory errors from being located.

If the bit is set to “1”, the SARA-S will report control memory parity error and continue further processing.

Note: This option is only available on the SARA-S. The SARA-R always flags the error but continues processing.

Bit 3: RESERVED:

This bit must be set to “0” for proper device operation.

Bit 2: EVEN_PAR:

0 - Odd parity.

1 - Even parity.

This bit determines the parity attribute across the control memory interface, packet memory interface, and the cell interfaces.

When set to “0”, odd parity is generated across control memory and cell interfaces; odd parity is checked across the packet memory and control memory interfaces if respective parity checking is enabled.

When set to “1”, even parity is generated across control memory and cell interfaces; even parity is checked across the packet memory and control memory interfaces if respective parity checking is enabled.

Bit 1: ON_LINE:

0 - Schedule chip to enter off-line gracefully.

1 - Chip is on-line.

This bit is used to control the segmentation process. Upon hard reset or soft reset, the chip defaults to the off-line mode so that the host can set up the chip properly.

Programming this bit to “0” during the middle of a segmentation cycle will force SARA-S to gracefully enter the off-line state after completing the current segmentation cycle(s) and will set the OFF_LINE bit in the state register. Further segmentation and control memory accesses are inhibited during the off-line state and will resume only after SARA-S is reprogrammed to be on-line.

Bit 0: CTRL_WR_DIS:

0 - Controlled registers writable.

1 - Controlled registers write-protected.

Programming this bit to a “0” permits the processor to modify the contents of all controlled registers. This is the reset state of the chip. During normal operation, this bit may be set to “1” as an added protection to prevent inadvertent modification of the controlled registers.

4.1.2 Mode Register 1

The following describes the individual bits in SARA-S Mode Register 1**:**

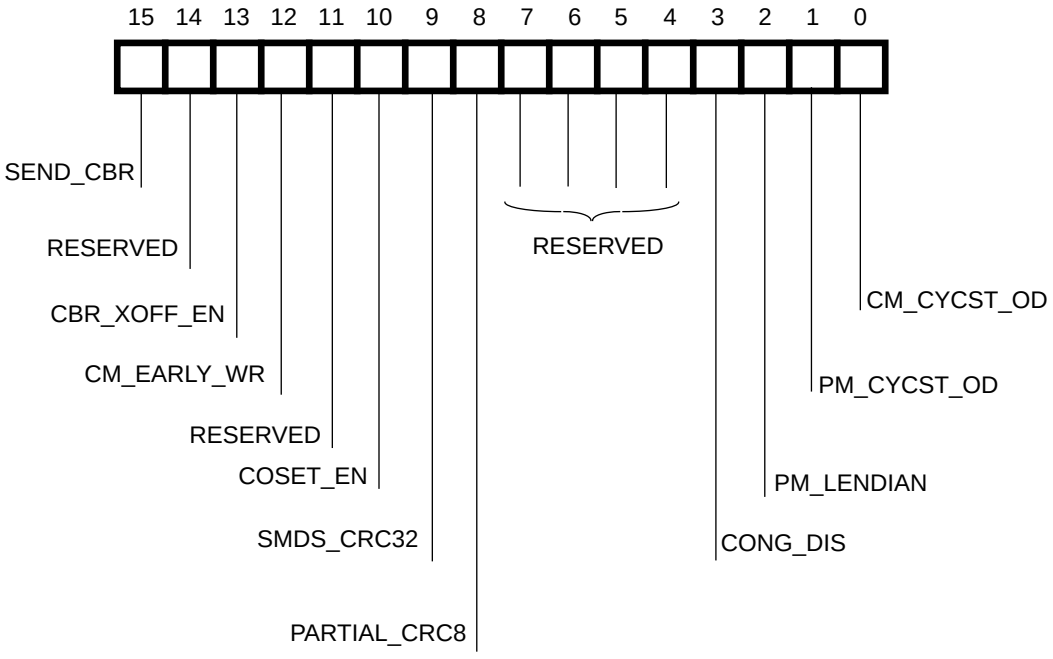


Figure 4-2. Mode Register 1 Data Bits

Bit 15: SEND_CBR:

SARA-S will send a constant bit rate (CBR) cell on every rising edge of this bit. This bit will get auto cleared on successful transfer of a CBR cell. Software can detect this either by an interrupt generated from the interrupt status register or by polling this bit.

Bit 14: RESERVED:

Must be programmed to “0” for proper device operation.

Bit 13: CBR_XOFF_EN:

0 - CBR traffic is not subject to XON/XOFF.

1 - CBR traffic is subject to XON/XOFF protocol.

If this bit is set, CBR traffic is suspended when XON signal is not active.

Bit 12: CM_EARLY_WR:

0 - CWRT pin is not pipelined to indicate early write.

1 - CWRT pin is pipelined to indicate write on clock earlier.

If this bit is set, the CWRT pin will become active one cycle early so that the external logic can generate synchronous write signals to the control RAMs.

Bit 11: RESERVED:

Must be programmed to “0” for proper device operation.

Bit 10: COSET_EN:

0 - Disable Header Checksum exclusive ORed with COSET.

1 - Enable Header Checksum exclusive ORed with COSET.

If this bit is programmed to a “1”, the eight-bit header error checksum generated by polynomial ($x^8 + x^2 + x + 1$) is XORed with the COSET polynomial ($x^6 + x^4 + x^2 + 1$). This mode bit must be set to a “1” for T1S1.5 compatibility.

Bit 9: SMDS_CRC32:

0 - CRC32 is optionally calculated and appended to AAL3/4 frames in a transparent manner.

1 - CRC32 is calculated and appended as per the IEEE 802.6 requirement.

CRC32 is optionally calculated and appended to frames segmented by the SARA-S. SARA-S calculates and appends CRC32 if the APP_CRC32 bit is set in either the descriptor table or in the VC table. The polynomial used is ($x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$). For AAL5, this bit must be programmed to a ‘0’.

Bit 8: PARTIAL_CRC8:

0 - Header checksum is calculated on bytes 0, 1, 2 and 3 of the ATM cell header.

1 - Header checksum is calculated only on bytes 1, 2 and 3 of the ATM cell header.

SARA-S can calculate the header checksum on either the full four bytes of the ATM header or only on the last three bytes of the ATM header. In SMDS/802.6 applications, the header byte-0 is used as the access control field and is not used for computing the headed check sum. In such applications, this bit must be set to a “1”.

Bit (7:4): RESERVED:

These bits must be programmed to “0”for proper device operation.

Bit 3: CONG_DIS:

0 - Enable congestion control mechanism

1 - Disable congestion control mechanism

This bit controls whether congestion control information from the SARA-R is processed by SARA-S. When set to “0”, the congestion control processing is enabled. When set to “1”, the congestion control processing is disabled. While disabled (CONG_DIS=1), the CCHLD signal is held active.

Bit 2: PM_LENDIAN:

0 - Packet memory is organized as Big-endian

1 - Packet memory is organized as Little-endian

This bit indicates how the SARA-S should interpret the data in the packet memory. Both the little-endian and the big-endian formats are supported for increased system design flexibility.

Bit 1: PM_CYCST_OD:

0 - Packet memory PCYCST* signal has an output that has an internal active pull-up

1 - Packet memory PCYCST* signal is an open drain output without internal active pull-up

With this bit, the PCYCST* pin can be programmed as a regular output with an active pull up and pull down or as an open drain output. It should be noted that the active levels of this bit are opposite to the equivalent bit in SARA-R.

Bit 0: CM_CYCST_OD:

0 - Control memory CCYCST* signal has an output that has an internal active pull-up

1 - Control Memory CCYCST* signal is an open drain output without internal active pull-up

With this bit, the CCYCST* pin can be programmed as a regular output with an active pull up and pull down or as an open drain output.

It should be noted that the active levels of this bit are opposite to the equivalent bit in SARA-R.

4.2 Interrupt Status Register

Figure 4-3 shows the SARA-S interrupt status register bits. The interrupt status register contains “sticky” bits which are set by certain events. Every bit in the status register has an associated bit in the mask register. If a particular mask bit is set to a “0”, then the corresponding status bit, when set, will result in the generation of an interrupt. If a particular mask bit is set to a “1”, then the corresponding status bit will not result in generating an interrupt, but will be set when the event occurs. The status bits are cleared when read by the processor.

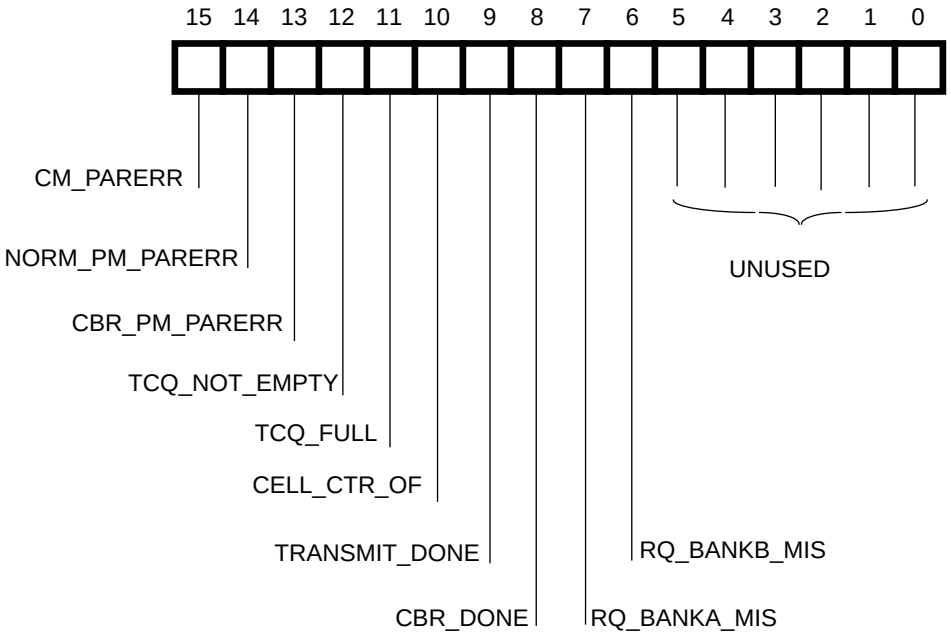


Figure 4-3. Interrupt Status Register Bit Positions

Bit 15: CM_PARERR:

This bit, when set, indicates that a parity error has been detected during control memory read operations. Further actions of the chip depend on the state of CM_ERR_MODE in Mode Register 0.

Bit 14: NORM_PM_PARERR:

This bit, when set, indicates that a parity error has been detected in the AAL-3/4 packet data when segmenting a packet. Further segmentation of the packet in error is inhibited and the corresponding buffer descriptor is freed by queueing it in the transmit complete queue. The completion code with such descriptors in the transmit complete queue will indicate a packet memory error.

Bit 13: CBR_PM_PARERR:

This bit, when set, indicates that a parity error has been detected in the CBR or AAL-5 packet data when segmenting a packet. Further segmentation of the constant bit rate traffic will, however, continue.

Bit 12: TCQ_NOT_EMPTY:

After segmentation of any packet is completed, the buffer descriptor is delinked from SARA-S internal data structures and queued in the transfer complete queue (TCQ). When the TCQ first goes non-empty, this bit is set indicating that there are descriptors queued in the TCQ. The processor can then choose to process the completed descriptor. Note: This bit is set only when TCQ state changes from empty to non-empty.

Bit 11: TCQ_FULL:

This bit when set, indicates that the transfer complete queue was full. The SARA-S will continue to keep the descriptors of the transmitted packets linked in its data structures as long as the queue is full.

Bits 10: CELL_CTR_OF:

When set, this bit indicates that the 32-bit transmit cell counter has overflowed.The counter will then wrap around and continue counting from zero.

Bits 9: TRANSMIT_DONE:

When a descriptor with XD_INTT bit set is delinked after segmentation, this status bit will be set by SARA-S.

Bits 8: CBR_DONE:

This bit is set after every successful cell transfer associated with constant bit rate traffic.

Bit 7: RQ_BANKA_MIS:

This bit, when set, indicates that one of the rate queues in bank A missed getting serviced. This could be due to one of the following conditions:

- The rate queues are over subscribed (not enough bandwidth is available to service all rate queues).
- The link/cell interface is not accepting the cells from SARA-S.

Bit 6: RQ_BANKB_MIS:

This bit, when set, indicates that one of the rate queues in bank B missed getting serviced. This could be due to one of the following conditions:

- The rate queues are over subscribed (not enough bandwidth is available to service all rate queues).
- The link/cell interface is not accepting the cells from SARA-S.

Bit (5:0): UNUSED:

These bits are unused.

4.3 State Register

Figure 4-4 shows the SARA-S State register bits. These bits do not generate an interrupt. They are read-only status bits indicating the dynamic state of the chip. Values written to these locations are ignored. All bits in this register except the CM_ERROR bit get cleared automatically when the condition causing them to be set goes away. The CM_ERROR bit is a sticky bit that will be cleared only by a reset (soft reset, state machine reset or hard reset).

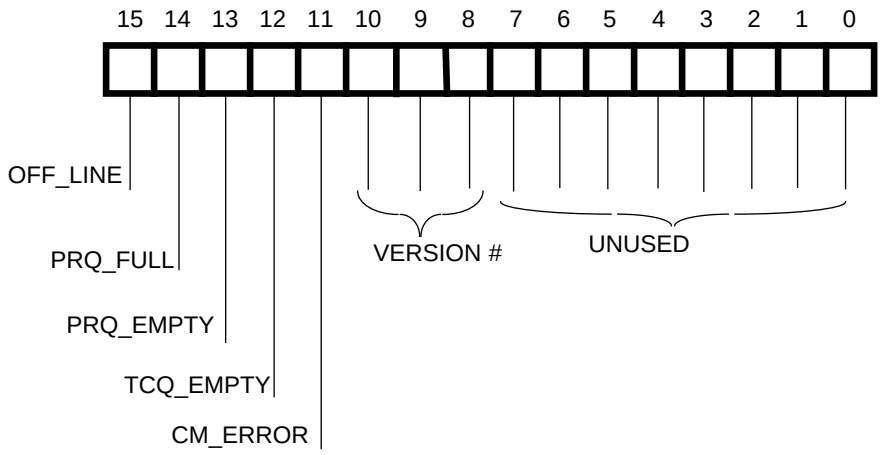


Figure 4-4. State Register Bit Positions

Bit 15: OFF_LINE:

This bit indicates the state of the chip and is set to “1” immediately after a hard/soft reset (chip is in off-line state). Packet segmentation and control memory accesses will not be performed until the chip is put back on-line (ON_LINE of Mode Register 0).

When the processor forces the chip to enter the off-line state, this bit must be polled to see if the chip has gracefully entered off-line. If this bit still indicates on-line (“0”), then the chip is in the middle of a segmentation cycle and has not yet entered the off-line state (“1”).

During on-line mode, this bit will be in the “0” state.

Bit 14: PRQ_FULL:

This bit indicates the packet ready queue full state. When the queue is full, this bit will read “1”. Software may want to read this bit before queueing a new descriptor for segmentation. This bit will be a “0” when the queue is not full.

Bit 13: PRQ_EMPTY:

This bit indicates the packet ready queue empty state. When the queue is empty, this bit will read a “1”. This can be used as a quick indication to check if the descriptors queued for segmentation have been linked by SARA-S into its internal data structures.

Bit 12: TCQ_EMPTY:

This bit indicates the transmit complete queue empty state. When the queue is empty, this bit will read a “1”. This can be used as a quick indication to check if SARA-S has delinked and freed up descriptors after packet segmentation. This bit will be a “0” when the queue is not empty.

Bit 11: CM_ERROR:

This bit indicates if SARA-S has detected a control memory parity error since the previous reset (hard or soft or state machine reset).

- If CM_ERR_MODE of Mode Register 0 is set to a “1”, then SARA-S must be reset (hard or soft or state machine reset) for further segmentation or control memory accesses to occur.
- If CM_ERR_MODE of Mode Register 0 is set to a “0”, then SARA-S continues segmentation. This bit will however, be set to indicate that a control memory error was noticed.

Bits (10:8): VERSION:

The version number of SARA-S is hard-coded in these bit locations.

Bits (7:0): UNUSED

These bits are unused.

4.4 Interrupt Mask Register

Figure 4-5 shows the SARA-S interrupt mask register bits. Each bit in the mask register corresponds to an associated bit in the interrupt status register. If a particular mask bit is set to a “0”, then the corresponding status bit when set will result in an interrupt generation. Setting a mask bit to a “1” will only suppress interrupt generation due to the associated status condition.

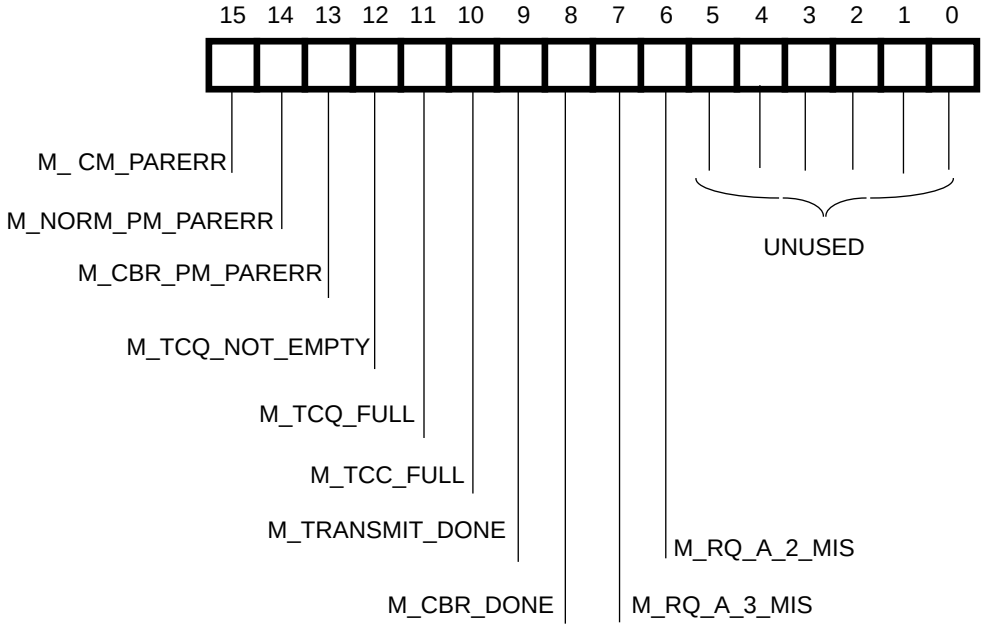


Figure 4-5. Bit Positions for SARA-S Interrupt Mask Register

Bit 15: M_CM_PARERR

0 - Unmask interrupt generation because of CM_PARERR

1 - Mask interrupt generation because of CM_PARERR

Bit 14: M_NORM_PM_PARERR

0 - Unmask interrupt generation because of NORM_PM_PARERR

1 - Mask interrupt generation because of NORM_PM_PARERR

Bit 13: M_CBR_PM_PARERR

0 - Unmask interrupt generation because of CBR_PM_PARERR

1 - Mask interrupt generation because of CBR_PM_PARERR

Bit 12: M_TCQ_NOT_EMPTY

0 - Unmask interrupt generation since TCQ_NOT_EMPTY

1 - Mask interrupt generation since TCQ_NOT_EMPTY

Bit 11: M_TCQ_FULL:

0 - Unmask interrupt generation since TCQ_FULL

1 - Mask interrupt generation since TCQ_FULL

Bit 10: M_TCC_FULL:

0 - Unmask interrupt generation because of transmit cell counter overflow

1 - Mask interrupt generation because of transmit cell counter overflow

Bit 9: M_TRANSMIT_DONE:

0 - Unmask interrupt generation because of transmit done on specially marked descriptors

1 - Mask interrupt generation because of transmit done on specially marked descriptors

Bit 8: M_CBR_DONE:

0 - Unmask interrupt generation because of CBR cell transfer

1 - Mask interrupt generation because of CBR cell transfer

Bit 7: M_RQ_A_MIS:

0 - Unmask interrupt generation because of service misses to bank A

1 - Mask interrupt generation because of service misses to bank A

Bit 6: M_RQ_B_MIS

0 - Unmask interrupt generation because of service misses to bank B

1 - Mask interrupt generation because of service misses to bank B

Bits (5:0): UNUSED

These bits are unused.

4.5 Rate Queue Registers

There are eight rate queues in SARA-S. Each rate queue is programmed for the peak rate at which the packets on that queue are segmented. The rate queues are organized as two banks as shown in Table 4-4. All rate queues within a bank are serviced on a sequential or “round robin” basis.

High Priority Bank Rate Queue Registers	Low Priority Bank Rate Queue Registers
RQ_REG_A0	RQ_REG_B0
RQ_REG_A1	RQ_REG_B1
RQ_REG_A2	RQ_REG_B2
RQ_REG_A3	RQ_REG_B3

Table 4-4. Rate Queue Registers

Each rate queue has an internal eight-bit rate counter. When this rate counter overflows, it “rolls over” to the preload value contained in its rate queue register and generates a service request for that queue. Continuous service requests from queues in the high priority rate queue can keep the low priority rate queue service requests from being serviced. When a rate queue is being serviced, one cell from each actively linked descriptor is transferred to the cell interface before another rate queue service request is processed.

Descriptors are linked in the rate queue so that only one packet is actively segmented for each virtual circuit at any given time. Segmentation of other packets to the same virtual circuit occur only after the current packet to that virtual circuit has been completely segmented, thus maintaining an ordered segmentation.

In addition to the eight bits of “rate” preload value, the rate register contains a prescaler (two bits), an enable (one bit) and an XOFF enable (one bit) as shown in Figure 4-6. The prescaler is used to pre-scale CLK input before using it as a clock to the rate counter. The prescaler and the preload value together provide a choice of frequencies at which the rate queue overflows, thus determining the peak segmentation rates of each rate queue. The enable bit is used to enable the rate counter. The RQ_XOFF_EN bit determines if the rate queue processing is suspended when the XON input is false. If this bit is set, the rate queue will not generate any service requests. Also, if the rate queue is being processed when the XON input becomes false, the processing of the rate queue will stop gracefully at the next cell boundary.

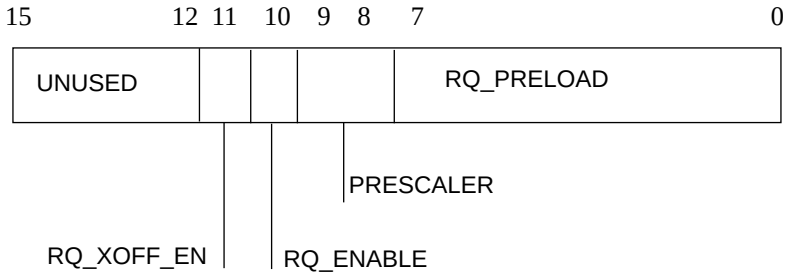


Figure 4-6. Rate Queue Register

Bits in the rate queue registers are described below:

Bits (15:12): Unused

Bit 11:RQ_XOFF_EN

0 - Ignore XON input while processing this rate queue

1 - Rate queue is processed only if XON input is true

Bit 10: RQ_ENABLE

0 - Disable servicing of the rate queue

1 - Enable servicing of the rate queue

Bits (9:8): PRESCALER:

00 - Divide CLK by 4 and use as rate counter clock

01 - Divide CLK by 16 and use as rate counter clock

10 - Divide CLK by 64 and use as rate counter clock

11 - Divide CLK by 256 and use as rate counter clock

Bits (7:0): RQ_PRELOAD

This is the internal rate counter “preload” value.

Values written to the lower eight-bits of the rate queue registers determine the peak segmentation rate of frames linked to that particular rate queue. The programming value is determined as follows:

R = | (255 - (424/Peak_rate)*(1000/Tclk)*(1/Prescaler)) |

where:

- R is the integral “pre-load” value programmed into the rate register
- Peak_rate is the Peak segmentation rate desired in Mbps
- Tclk is the SARA-S “CLK” period in ns.
- Prescaler is a choice of 4, 16, 64, 256, programmed into the rate register.

The rates programmed into the rate queue registers are system parameters. These rates must be based on the number of rate queues active at any given time and the number of descriptors of different VCs that have been associated with each rate queue.

The aggregate sum of all peak rates of active rate queues should not exceed either the link or the SARA-S bandwidth to avoid over-subscription. Over-subscription can result in blocking the low priority bank rate queues and starving rate queues in the same bank. Starvation due to oversubscription translates into reduced peak rate on each of the rate queues.

Table 4-5 contains the segmentation rates possible for SARA-S with a 20 MHz (50 ns) clock for various prescaler values.

Prescaler	Peak Rate (Mbps) Range
4	155 Mbps down to 8.3 Mbps
16	155 Mbps down to 2.1 Mbps
64	132 Mbps down to 520 kbps
256	33 Mbps down to 130 kbps

Table 4-5. Peak Segmentation Rates for Different Prescaler Values

Table 4-6 contains representative preload values in decimal for programming into the rate queue registers for indicated peak rates. The values are determined using the formula described earlier. A 20 MHz (50 ns) CLK is assumed.

Desired peak segmentation rate	PROGRAM VALUE			
	Prescaler=00	Prescaler=01	Prescaler=10	Prescaler=11
1 Mbps	INVALID	INVALID	122	221
5 Mbps	INVALID	149	228	248
10 Mbps	43	202	241	251
20 Mbps	149	228	248	INVALID
45 Mbps	207	243	252	INVALID
50 Mbps	212	244	INVALID	INVALID
100 Mbps	233	249	INVALID	INVALID
155 Mbps	241	251	INVALID	INVALID

Table 4-6. Rate Queue Register Programming Values

4.6 Queue Base Address Register

This register contains the base address of the packet ready queue and the transmit complete queue in control memory. This base address, when concatenated with the packet ready or the transmit complete queue read or write pointer, creates the corresponding queue‘s memory address (byte address). This is illustrated in Figure 4-7.

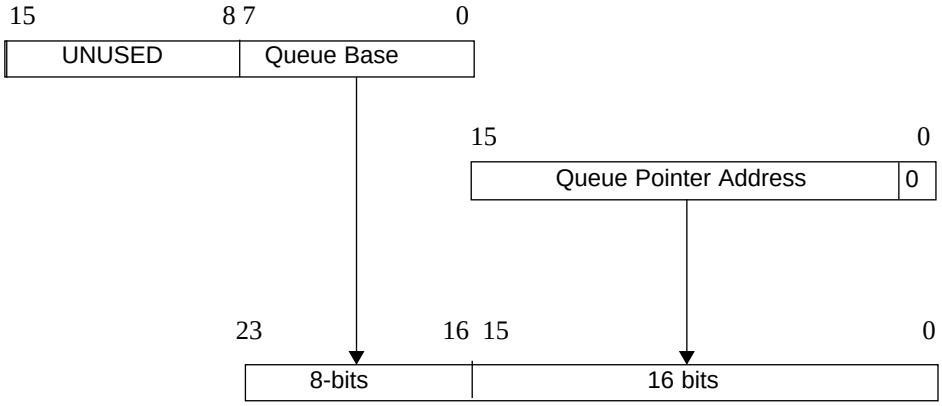


Figure 4-7. SARA-S Queue Address Generation

4.7 Descriptor Table Base Address Register

This register contains the base address of the buffer descriptor table. The value in the descriptor base address when concatenated with the descriptor number and the table offset, points to the buffer descriptor entry in control memory. Each buffer descriptor table entry is 32-bytes long. The table offset determines the specific 16-bit word within the table entry being accessed. Figure 4-8 shows how the descriptor table byte address is formed.

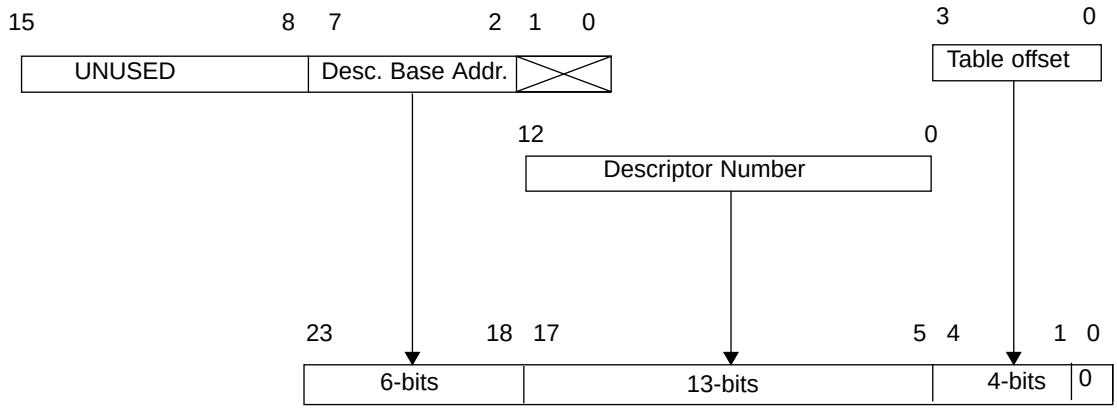


Figure 4-8. SARA-S Descriptor Table Address Generation

4.8 Virtual Circuit (VC) Table Base Address Register

This register contains the base address of the VC table. The value in this VC_LKUP_BASE register is concatenated with the lower bits of the VC index to generate the VC table entry address. The VC index is obtained either from the descriptor table during normal processing or from the congestion control information passed by the SARA-R during congestion control processing.

The least significant three bits of this base register determines the number of bits of the base register used in generating the address. This is useful in configuring the control memory for various table sizes and is illustrated in Figure 4-9. Table 4-7 shows the number of maximum VC Table entries possible for various values in the least significant 3-bits of the VC_LKUP_-BASE register. The lower order bits of the VC index are used to determine the memory address when the LSB of VC_LKUP_BASE is programmed to use less than 16 bits of the VC index.

Least Significant Bits of VC Base Register	Memory address		Max number of VCs possible
	Number of bits used from VC_LKUP_BASE	Number of bits used from VC_Index	
000	4	16	65,536
001	5	15	32,768
010	6	14	16,384
011	7	13	8,192
100	8	12	4,096
101	9	11	2,048
110	10	10	1,024
111	11	9	512

Table 4-7. Number of VC Table Entries for Various Values of VC_LKUP_BASE Register

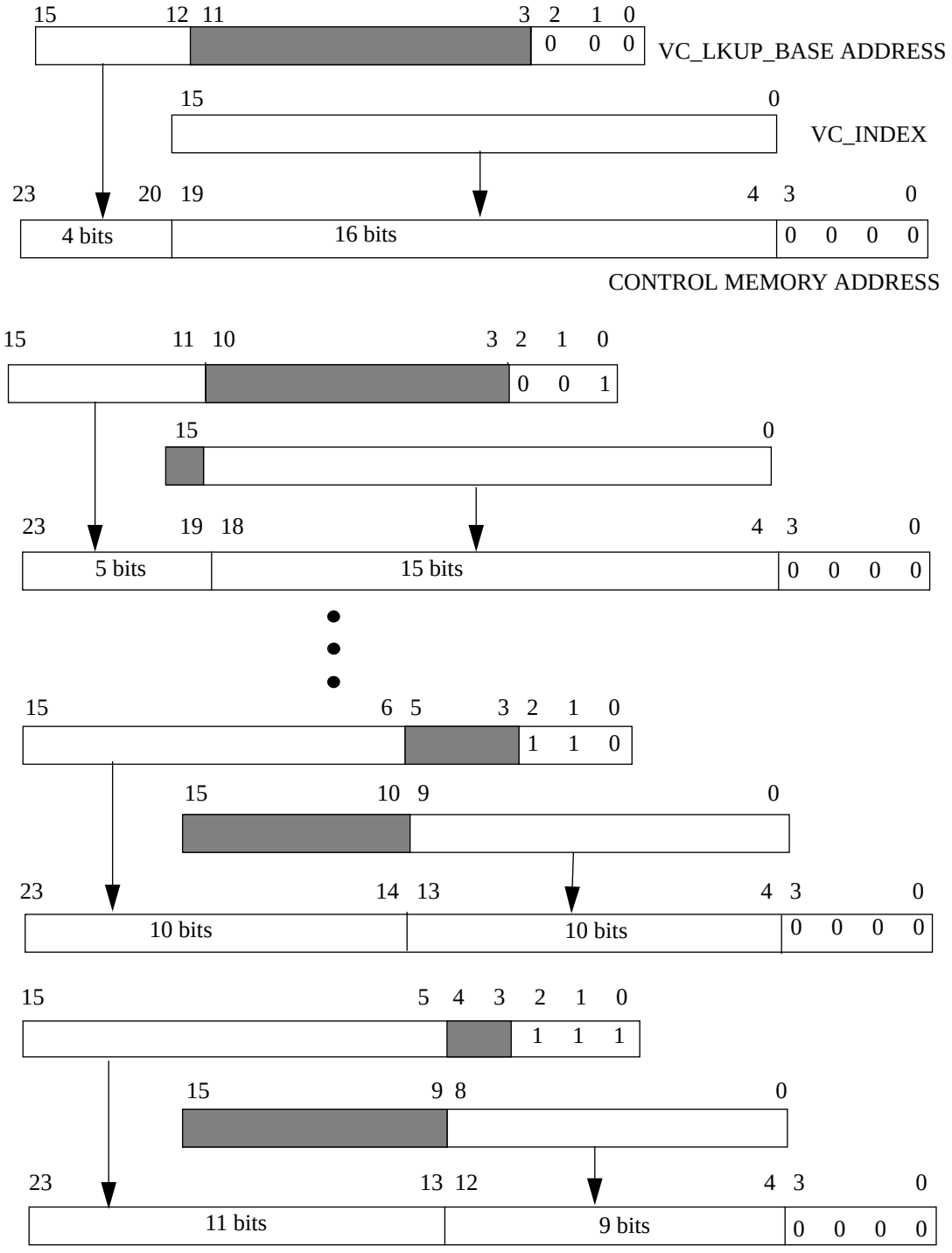


Figure 4-9. SARA-S VC Table Entry Address Generation

4.9 Packet Memory Address Match Register

During packet memory cycles the packet memory long word address is compared to the value programmed in the packet memory address register. Upon match, the signal PAMTCH is forced active. The external memory system can use this signal to terminate the memory cycles. This is helpful when memory cycle termination is required when crossing a page boundary (Example: DRAM page boundary).

Figure 4-10 describes the operation of this register. Bits (7:0) of this register are always compared with bits (9:2) of the packet memory address. The contents of bits (15:12) are used as mask bits to determine whether bits (11:8) are also used in the comparison. Packet memory address bits (1:0) are always ignored for generating the PAMTCH output.

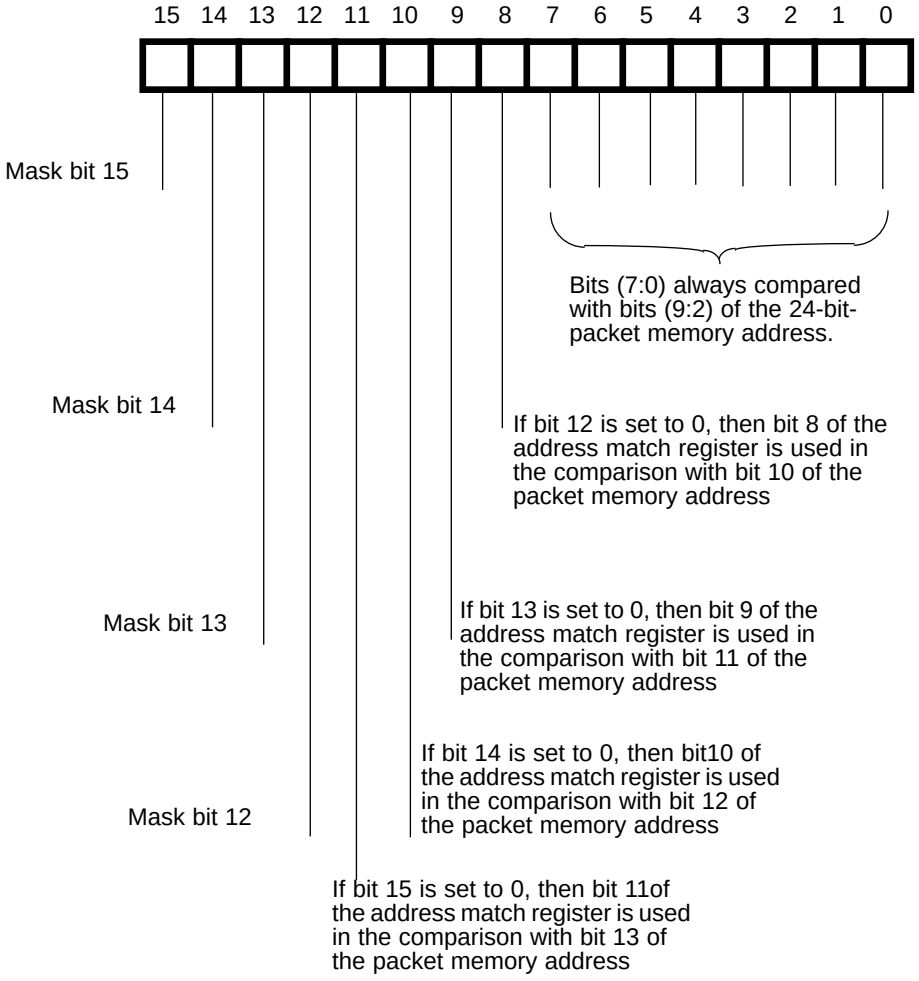


Figure 4-10. SARA-S Packet Memory Address Match Register

4.10 Command Register

The command register is a write-only register used to issue software commands to SARA-S as follows:

- reset SARA-S state machine, registers and counters when xx55 is written to this register. The chip goes off-line.
- reset SARA-S state machine (but not registers and counters) when xxAA is written to

this register
- reset SARA-S transmit cell counters when xxCC is written to this register

Values other than xxAA, xx55 or xxCC written to this register will be ignored.

4.11 Communication Queues

SARA-S uses two queues to communicate with the host/processor for passing descriptors—the packet ready queue and transmit complete queue. The packet ready queue consists of:

- PRQ_ST_ADR register which contains the packet ready queue start address
- PRQ_ED_ADR register which contains the packet ready queue end address
- PRQ_RD_PTR which contains the packet ready queue read pointer
- PRQ_WR_PTR which contains the packet ready queue write pointer.

The transmit complete queue consists of:

- TCQ_ST_ADR register which contains the transmit ready queue start address
- TCQ_ED_ADR register which contains the transmit ready queue end address
- TCQ_RD_PTR which contains the transmit ready queue read pointer
- TCQ_WR_PTR which contains the transmit ready queue write pointer

The queue length (difference between end address and start address) should always be programmed greater (at least by one) than the maximum number of descriptors used. This prevents any possible overflow condition from occurring.

The packet ready queue is used by the processor to pass the buffer descriptor number of the frames ready for segmentation to SARA-S.

The transmit complete queue is used by the SARA-S to return descriptors to the processor after segmentation has been completed. Besides the descriptor number, it also contains the descriptor completion code in its most significant 3-bits as shown in Figure 4-11. The following completion codes are valid:

- 000 - Normal completion.
- 001 - Descriptor was released as a result of the flush condition in the VC tables mode bits.
- 111 - Descriptor was released as a result of packet memory parity error.
- Other codes: Reserved for future definition.

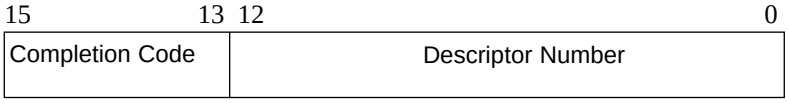


Figure 4-11. Transmit Complete Queue

Figure 4-12 shows the operational states of the queues during various conditions. The queues operate as follows: After reset, read pointer (RP) = write pointer (WP) = starting address (SA). Whenever RP=WP, the queue is empty. Descriptors are added to the queue (the processor adds descriptors to the packet ready queue and SARA-S adds descriptors to the transmit complete queue) by writing to the entry location pointed to by the WP and then incrementing WP by one entry length (two bytes). The WP wraps around when it exceeds the end address (EA). The SARA-S automatically wraps the pointer (WP or RP) as it increments this pointer. Software is responsible for detecting the EA boundary and wrapping the pointer to SA. When the WP is one entry short of the RP, the queue full condition is reached, preventing new descriptor numbers from being added to the queues.

The SARA-S will not write into a full queue. Software must check for the full condition by checking the pointer before writing into the packet ready queue. The same is true for software reading from the transmit complete queue, when empty. Descriptors are retrieved from the queue (the processor reads descriptors from the TCQ; SARA-S reads from the PRQ) when a non-empty queue condition is detected, by reading from the entry location pointed to by the RP and then incrementing the RP by one entry length (two bytes).

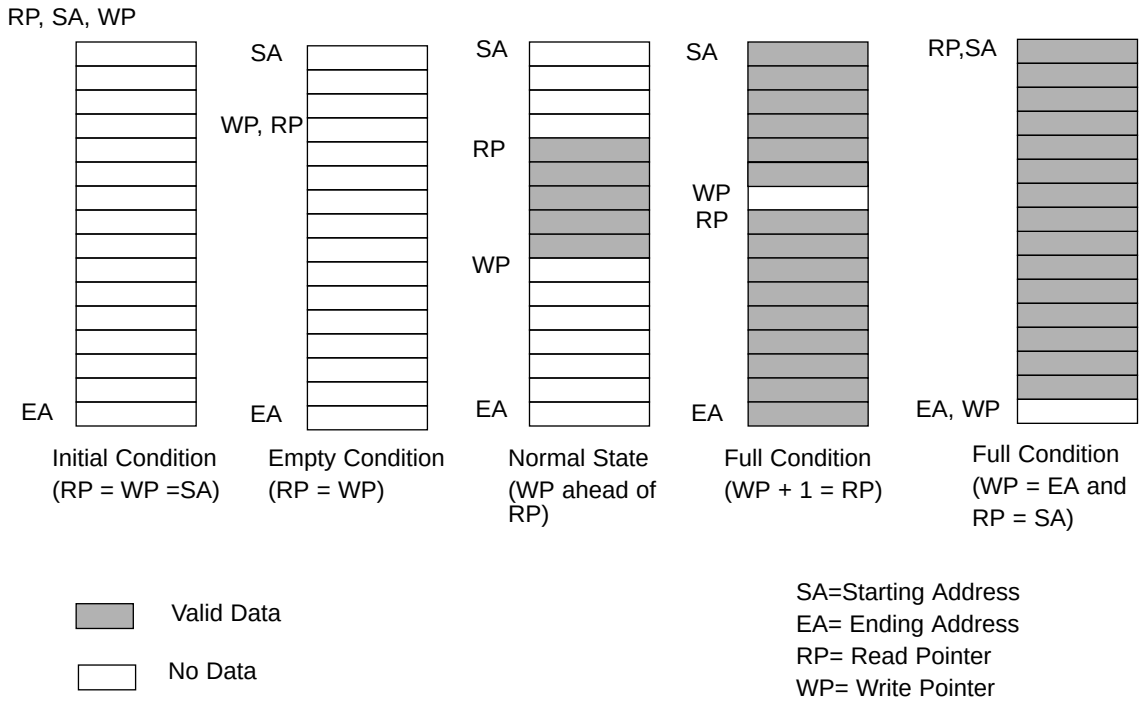


Figure 4-12. SARA-S Packet Ready and Transmit Complete Queue Operational States

4.12 Cell Counter Registers

This is a 32-bit counter accessed through two separate 16-bit read only registers—CELL_CTR_HI (CELL_CTR_HI_NC) and CELL_CTR_LO (CELL_CTR_LO_NC). The count indicates the total number of cells transmitted by SARA-S since it was previously reset. Each of these counters can be accessed by reading two separate locations. Reading these counters from one of the locations will auto clear the respective counter, while reading from the alternate location will preserve the count value without clearing the counters. In a system environment when the counters are accessible by different network entities, it is desirable that the counters do not auto clear on read.

For an accurate count, it is recommended that CELL_CTR_LO (CELL_CTR_LO_NC) be read before reading CELL_CTR_HI (CELL_CTR_HI_NC). If the SARA-S transmits cells back-to-back on a continuous basis at a 150 Mbps rate, it takes over 3 hours for this counter to overflow. When the counter overflows, an interrupt status bit is set, generating a maskable interrupt.

4.13 Constant Bit Rate Address Registers

These are two registers, CBR_ADDR_HI and CBR_ADDR_LO. The 16 bits of CBR_ADDR_HI and the 16 bits of CBR_ADDR_LO together form the 32-bit DMA start address (byte address) for constant bit rate (CBR) traffic. This address can be mapped either to a buffer loca-

tion in the same packet memory used to store frames queued for segmentation or to an external FIFO like device. The address programmed into the CBR address registers must be in bytes. A total of 52 bytes will be DMA-transferred starting from this location every time a CBR cell transmit request is received. The cell transmit request can originate either from an external hardware interface (CBRXMIT pin) or through software control (SEND_CBR mode bit). The data DMA-transferred into SARA-S must contain the VC table index of the CBR data followed by 48 bytes of payload. The data format is presented in Chapter 6.

4.14 Rate Queue Starting Descriptor Number Diagnostic Registers

There are eight rate queue starting descriptor number registers as follows:

- RQ_A_3_SDN
- RQ_A_2_SDN
- RQ_A_1_SDN
- RQ_A_0_SDN
- RQ_B_3_SDN
- RQ_B_2_SDN
- RQ_B_1_SDN
- RQ_B_0_SDN

Each register is associated with a corresponding rate queue. Together they contain the starting descriptor number for the linking data structure in each rate queue. This data is made accessible for diagnostic purposes and must not be programmed for normal operation.

4.15 Current Descriptor Number Diagnostic Register

When the SARA-S chip processes a descriptor either for linking in its rate queue, segmenting, or for delinking purposes, it stores the descriptor number being processed in this register. The register can therefore be read to find the descriptor most recently processed or being currently processed by the SARA-S chip. This is made accessible for diagnostic purposes and must not be programmed for normal operation.

4.16 CC_Label Diagnostic Register

When SARA-R sends any congestion control information to SARA-S, the VC label associated with that cell is stored by SARA-S in this register. This is made accessible for diagnostic purposes and must not be programmed for normal operation.

Chapter 5. SARA-R Register Descriptions

This chapter serves as a software reference for the SARA-R chipset. Detailed descriptions of the processor/host registers and the queues are included.

All SARA-R registers are accessed as 16 bits. Bit 0 is the least significant bit. All reserved and unused bits in these registers must be set to “0” for proper operation unless otherwise mentioned. Table 5-1 shows the SARA-R Processor Registers used for chip setup. The “Reset Value” column in the Tables specify default reset/power-up value of the Registers.

Register Name	Addr (Hex)	Description	Type[1]	Reset Value
MODE_REG_0	00	Mode Register 0	R/W	0000H
MODE_REG_1	01	Mode Register 1	R/W	1000H
MASK_REG	02	Mask Register	R/W	FFFFH
INTR_STATUS_REG	03	Interrupt Status Register	R/O	0000
DRP_PKT_CNTR	04	Dropped Packet Counter (cleared when read)	R/W	0000H
ERR_CNTR	05	Error Counter (cleared when read)	R/W	0000H
DRP_CBR_CNTR	06	Dropped CBR Cells Counter (cleared when read)	R/W	0000H
CBR_BASE_ADR	08	Base address for CBR and congestion control queues	R/CW	0000H
PM_ADDR_MTCH	09	Packet Memory Address Match Register	R/CW	0000H
CELL_CNTR_LO	0C	Cell Counter 0 (cleared when read)	R/CW	0000H
CELL_CNTR_HI	0D	Cell Counter 1 (cleared when read)	R/CW	0000H
COMMAND_REG	0F	Command Register	W/O	N/A
DESC_BASE	10	Base address for descriptor table	R/CW	0000H
VC_LKUP_BASE	11	Base address for VC lookup table	R/CW	0000H
REASS_BASE	12	Base address for reassembly table	R/CW	0000H
QUEUE_BASE	13	Base Address for Communication Queue	R/CW	0000H
PKT_TM_CNT	16	Packet Timeout and Count Register	R/CW	0000H
TMOUT_RANGE	17	Range of reassembly IDs for timeout	R/CW	0000H
INTRVL_CNTR	18	Packet Aging Interval Counter	R/CW	0000H
TMOUT_IND X	19	Index of packet being tested for aging	R/CW	0000H
VP_LKUP_BASE	1C	Base address for VP lookup table	R/CW	0000H

Table 5-1. SARA-R Internal Registers

Register Name	Addr (Hex)	Description	Type[1]	Reset Value
VP_FILTER	1D	VP Filter Register	R/CW	0000H
SML_Q_ST_ADR	20	Small Free Descriptor Queue Start Address	R/CW	0000H
SML_Q_ED_ADR	21	Small Free Descriptor Queue End Address	R/CW	0000H
SML_Q_RD_PTR	22	Small Free Descriptor Queue Read Pointer	R/CW	0000H
SML_Q_WR_PTR	23	Small Free Descriptor Queue Write Pointer	R/W	0000H
LRG_Q_ST_ADR	24	Large Free Descriptor Queue Start Address	R/CW	0000H
LRG_Q_ED_ADR	25	Large Free Descriptor Queue End Address	R/CW	0000H
LRG_Q_RD_PTR	26	Large Free Descriptor Queue Read Pointer	R/CW	0000H
LRG_Q_WR_PTR	27	Large Free Descriptor Queue Write Pointer	R/W	0000H
PCQ_ST_ADR	28	Packet complete Queue Start Address	R/CW	0000H
PCQ_ED_ADR	29	Packet complete Queue End Address	R/CW	0000H
PCQ_RD_PTR	2A	Packet complete Queue Read Pointer	R/W	0000H
PCQ_WR_PTR	2B	Packet complete Queue Write Pointer	R/CW	0000H
EXCP_Q_ST_ADR	2C	Exception Queue Start Address	R/CW	0000H
EXCP_Q_ED_ADR	2D	Exception Queue End Address	R/CW	0000H
EXCP_Q_RD_PTR	2E	Exception Queue Read Pointer	R/W	0000H
EXCP_Q_WR_PTR	2F	Exception Queue Write Pointer	R/CW	0000H
CBR_FIF_ST_ADR	30	CBR Queue Start Address	R/CW	0000H
CBR_FIF_ED_ADR	31	CBR Queue End Address	R/CW	0000H
CBR_FIF_RD_PTR	32	CBR Queue Read Pointer	R/W	0000H
CBR_FIF_WR_PTR	33	CBR Queue Write Pointer	R/CW	0000H
CC_FIFO_ST_ADR	34	Raw cell Queue Start Address	R/CW	0000H
CC_FIFO_ED_ADR	35	Raw cell Queue End Address	R/CW	0000H
CC_FIFO_RD_PTR	36	Raw cell Queue Read Pointer	R/W	0000H
CC_FIFO_WR_PTR	37	Raw cell Queue Write Pointer	R/CW	0000H
STATE_REG	38	State Register	R/O	8FFFH
SML_BUF_CHK	40	BA-size compare Register for small buffers	R/CW	0000H
LRG_BUF_CHK	41	BA-size compare Register for large buffers	R/CW	0000H

Table 5-1. SARA-R Internal Registers

Register Name	Addr (Hex)	Description	Type[1]	Reset Value
LRG_BUF_SIZE	42	Large buffer size	R/CW	0000H
OAMCC_CHK	44	OAM F5 code type for congestion control cells	R/CW	0000H
DRP_PKT_CNTR_NC	84	Dropped Packet Counter— don't auto-clear on Read	R/W	0000H
ERR_CNTR_NC	85	Error Counter— don't auto-clear on read	R/W	0000H
DRP_CBR_CNTR_NC	86	Dropped CBR cells counter— don't auto-clear on read	R/W	0000H
CELL_CNTR_LO_NC	8C	Cell Counter 0— don't auto-clear on read	R/CW	0000H
CELL_CNTR_HI_NC	8D	Cell Counter 1— don't auto-clear on read	R/CW	0000H

Table 5-1. SARA-R Internal Registers

1. R/W Read and Write
   R/O Read Only
   W/O Write Only
   R/CW Read and Controlled Write. Writable only when the chip is in the control write enable mode.

5.1 Mode Registers

Figure 5-1 and Figure 5-2 show the SARA-R Mode Register 0 and 1 data bits respectively. These two registers are used to setup the operating modes of the chip. Reserved and unused bits must be programmed as "0" for proper device operation, unless otherwise mentioned.

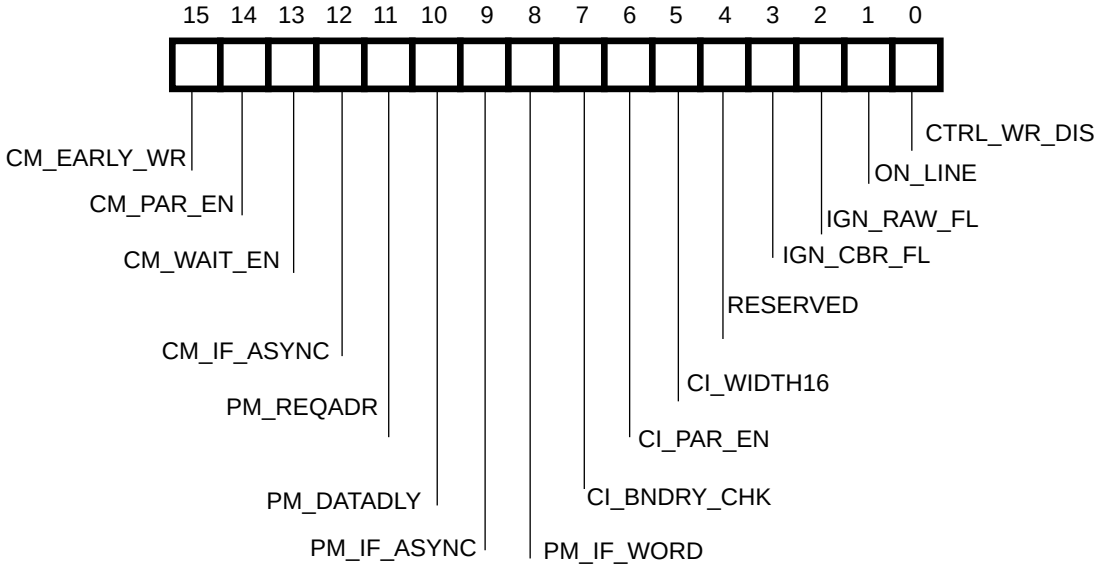


Figure 5-1. SARA-R Mode Register 0 Bit Positions

5.1.1 Mode Register 0

The following describes the individual bits in SARA-R Mode Register 0:

Bit 15: CM_EARLY_WR:

0 - Disable control memory early write signal

1 - Enable control memory early write signal

If this bit is set, the CWRT pin will become active one cycle early so that the external logic can generate synchronous write signals to the control RAMs.

Bit 14: CM_PAR_EN:

0 - Disable control memory parity checking

1 - Enable control memory parity checking

This bit is used to enable parity checking on control memory data. When set to “1”, either odd or even parity is checked based on the EVEN_PARITY bit.

Bit 13: CM_WAIT_EN:

0 - Ignore CRDY* input.

1 - Data transfer subject to CRDY* protocol.

This bit determines if the external CRDY* pin is sensed by the SARA-R chip. This bit must be set to '1' when the control memory is in the synchronous interface mode, and is ignored when the control memory is programmed to be in asynchronous mode.

Bit 12: CM_IF_ASYNC:

0 - Control memory interface is synchronous

1 - Control memory interface is asynchronous

In synchronous mode, data is transferred during control memory cycles on every clock (when CRDY* input pin is TRUE).

If the control memory interface is in asynchronous mode, the bus is relinquished after every transfer and will be re-requested. Data transfer is effectively through a complete CREQ/CGRT/CCYCST*/CRDY* handshake.

Bit 11: PM_REQADR

0 - Do not present packet memory address on the datalines PD(31:0) during packet memory bus requests

1 - Present packet memory address on the datalines PD(31:0) during packet memory bus requests

If this bit is set to “1”, the packet memory address bits 31:0 are presented on PD(31:0) data lines during memory bus request cycles for burst memory accesses.

These interactions are described inTable 5-2.

PREQ	PGRT	PLWADR	PD(31:0)
1	0	0	3-state
1	0	1	Address
1	1	X	Data

Table 5-2. Interactions of PREQ, PGRT, and PLWADR with Packet Data Bus PD(31:0)—SARA-R

Bit 10: PM_DATADLY:

0 - Do not delay packet memory data by half clock period

1 - Delay packet memory data by half clock period

This bit determines if the packet memory data is delayed by half a clock period or not. When programmed to a “1”, the data from the SARA-R is delayed by an additional half a cycle. This delay will effectively result in more hold time on the data, easing memory design requirements.

Bit 9: PM_IF_ASYNC:

0 - Packet memory interface is synchronous

1 - Packet memory interface is asynchronous

In synchronous mode, data can be transferred during packet memory cycles on every clock (when PRDY* input pin is TRUE). Either a burst of data or a single cycle data can be transferred under external PRDY input control.

If the packet memory interface is asynchronous mode, the bus is relinquished after every transfer and will be re-requested for the next transfer. Data transfer is effectively through a complete PREQ/PGRT/PCYCST*/PRDY* handshake.

Bit 8: PM_IF_WORD:

0 - Packet memory interface is 32 bits wide

1 - Packet memory interface is 16 bits wide

This bit programs the size of the packet memory interface, selecting between a 16- or a 32-bit wide data bus.

Bit 7: CI_BNDRY_CHK:

0 - Disable cell interface cell boundary check

1 - Enable cell interface cell boundary check

This bit is used to enable the boundary check feature on the cell interface receive data. When set to “1” the SARA-R checks that the first word of the cell has a Beginning-of-Cell marker. This marker is indicated by either setting the parity bit to “1” when parity has not been enabled (CI_PAR_EN = 0) on the cell interface or by parity inversion when parity has been enabled (CI_PAR_EN = 1).

Bit 6: CI_PAR_EN:

0 - Disable cell interface parity checking

1 - Enable cell interface parity checking

This bit is used to enable parity checking on the cell interface receive data. When set to “1”, either odd or even parity is checked based on the EVEN_PARITY bit. An incorrect parity bit signifies a Beginning-of-Cell marker if the CI_BNDRY_CHK bit is set. When this bit is set to “0”, a parity bit of “1” signifies a Beginning-of-Cell marker if the CI_BNDRY_CHK is set.

Bit 5: CI_WIDTH16:

0 - Cell interface is eight bits wide

1 -Cell interface is 16 bits wide

This bit selects between an eight-bit or a 16-bit wide cell interface.

Bit 4: RESERVED:

This bit must be programmed to “0” for proper device operation

Bit 3: IGN_CBR_FL:

0 - Do not ignore CBR queue full flag

1 - Ignore CBR queue full flag

If this bit is set to “0”, the new CBR cell is dropped if CBR queue is full. If this bit is set to “1”, the CBR queue is overwritten with new data even if the queue full condition is reached, ignoring the FULL flag.

Bit 2: IGN_RAW_FL:

0 - Do not ignore raw cell/congestion control queue full flag

1 - Ignore raw cell/congestion control queue full flag

If this bit is set to “0”, the new raw cell is dropped if raw queue is full. If this bit is set to “1”, the raw cell/congestion control queue is overwritten with new data even if the queue full condition is reached, ignoring the FULL flag.

Bit 1: ON_LINE:

0 - Schedule chip to enter off-line gracefully

1 - Chip is on-line

This bit is used to control the reassembly process. Upon either hard or soft reset, the chip defaults to the off-line mode so that the host can set up the chip properly.

Programming this bit to “0” during the middle of a reassembly cycle will force SARA-R to gracefully enter the off-line state after completing the current cycle(s) and will set the OFF_LINE bit in the state register. Further reassembly and control memory accesses are inhibited during the off-line state and will resume only after being reprogrammed to be on-line.

Bit 0: CTRL_WR_DIS:

0 - Controlled registers writable

1 - Controlled registers write-protected

Programming this bit to a “0” permits the processor to modify the contents of all controlled registers. This is the reset state of the chip. During normal operation, this bit may be set to “1” as an added protection to prevent inadvertent modification of the controlled registers.

5.1.2 Mode Register 1

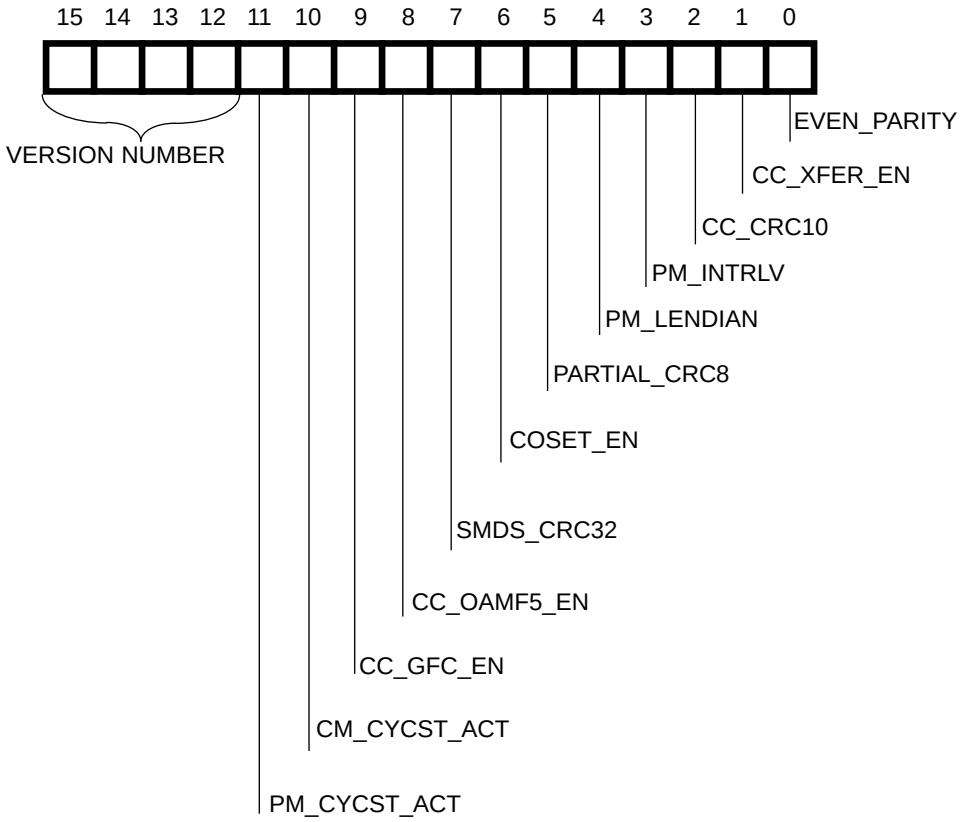


Figure 5-2. SARA-R Mode Register 1 Bit Positions

Figure 5-2 shows the individual bits in SARA-R Mode Register 1:

Bits (15:12): VERSION:

The version number of SARA-R is hard-coded in these bit locations.

Bit 11: PM_CYCST_ACT:

0 - Packet Memory PCYCST* signal is an open drain signal

1 - Packet Memory PCYCST* signal is an active pull up signal

Note: The active levels of this bit are opposite to the equivalent SARA-S bit.

Bit 10: CM_CYCST_ACT:

0 - Control Memory CCYCST* signal is an open drain signal

1 - Control Memory CCYCST* signal is an active pull up signal

Note: The active levels of this bit are opposite to the equivalent SARA-S bit.

Bit 9: CC_GFC_EN:

0 - Don’t detect congestion control from GFC bit

1 - Detect congestion control from GFC bit

This mode recognizes the congestion control (CC) cells via the generic flow control (GFC) MSB in the header. Cells with the most significant bit of GFC set to one are treated as a congestion control cell by the SARA-R when this mode bit is set to a “1”.

Bit 8: CC_OAMF5_EN:

This mode recognizes the congestion control (CC) cells from OAM F5 end-to-end payload whose type code matches the contents of the OAMCC_CHK register (Section 5.28).

0 - Don't detect CC from OAMF5

1 - Detect CC from OAMF5

The mode bits CC_GFC_EN and CC_OAMF5_EN work independently of each other to recognize congestion control cells, i.e. either or both of the modes can be used to recognize congestion control cells.

Bit 7: SMDS_CRC32:

This bit is used to indicate the range of data on which the 32-bit CRC is computed.

0 - Compute CRC over all data

1 - Compute CRC to conform to SMDS 802.6 standards

CRC32 is calculated and checked on the frames reassembled by the SARA-R. The polynomial used is ($x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$). The range of the checking is described in Section 1.2.1.

Bit 6: COSET_EN:

This bit controls the COSET header error check (HEC) enable.

0 - Disable Header Checksum exclusive ORed with COSET.

1 - Enable Header Checksum exclusive ORed with COSET.

When this bit is programmed to a "1", the eight-bit header error checksum generated by polynomial ($x^8 + x^2 + x + 1$) is XORed with the COSET polynomial ($x^6 + x^4 + x^2 + 1$). This mode bit must be set to a "1" for T1S1.5 compatibility.

Bit 5: PARTIAL_CRC8:

This bit controls HEC checking range.

0 - Compute HEC on all four bytes

1 - Ignore first byte for HEC

SARA-R can calculate the header checksum on either the full four bytes of the ATM header or can skip the first header byte. In SMDS/802.6 applications, the header byte-0 is used as the access control field and is not used for computing the header check sum. In such applications, this bit must be set to a "1".

Bit 4: PM_LENDIAN:

This bit controls the byte order.

0 - Big Endian byte order

1 - Little Endian byte order

This bit indicates how the SARA-R should generate the data in the packet memory. Both the little-endian and the big-endian formats are supported for increased system design flexibility.

Bit 3: PM_INTRLV

This is the packet memory address select for interleaved memory.

0 - Packet memory is not interleaved

1 - Packet memory is interleaved

This bit is used to select the packet memory address bits asserted on PA(15:0) pins. It can be programmed to ease problems in the memory system design. The input pin, PLWADR, determines whether the high address bits of the packet memory address bus or the low address bits of the packet memory address bits are presented on the bus.

When the low address bits are being requested (PLWADR = 1), address (15:0) is always presented on the address bus. When the high address bits are requested (PLWADR = 0), the address bits presented on the address bus depends on this mode bit:

- If the PM_INTRLV bit is 0, the high order address bits (23:8) are presented on the address bus.
- If the PM_INTRLV bit is 1 and the packet memory interface is 32 bits wide, the address bits (23:9,2) are presented on the address bus.
- If the PM_INTRLV bit is 1 and the packet memory interface is 16 bits wide, the address bits (23:9,1) are presented on the address bus.

Bit 2: CC_CRC10:

This bit controls the congestion control payload CRC check. It determines whether the congestion control cells are checked for correct 10-bit payload CRC. The payload CRC for congestion control cells will be checked only if this bit is set.

0 - Do not check congestion control cells for payload CRC

1 - Check congestion control cells for payload CRC

Bit 1: CC_XFER_EN:

This enables the transfer of congestion notification cells to the packet memory raw cell queue.

0 - Don’t transfer congestion control cells to raw cell queue

1 - Transfer congestion control cells to raw cell queue

Bit 0: EVEN_PARITY:

This bit determines the parity attribute across the control memory interface, packet memory interface and the cell interface.

0 - Odd parity

1 - Even parity

When set to “0”, odd parity is generated across control and packet memory interfaces; odd parity is checked across the cell and control memory interfaces if respective parity checking is enabled.

When set to “1”, even parity is generated across control memory and packet memory interfaces; even parity is checked across the cell and control memory interfaces if respective parity checking is enabled.

5.2 Interrupt Status Register

Figure 5-3 shows the SARA-R interrupt status register bits. The interrupt status register contains sticky bits which are set by certain events. Every bit in the status register has an associated bit in the mask register. If a particular mask bit is set to a “0”, then the corresponding status bit when set will result in the generation of an interrupt. If a particular mask bit is set to a “1”, then the corresponding status bit will not result in the generation of an interrupt, but will be set when the event occurs. The status bits are cleared when read by the processor.

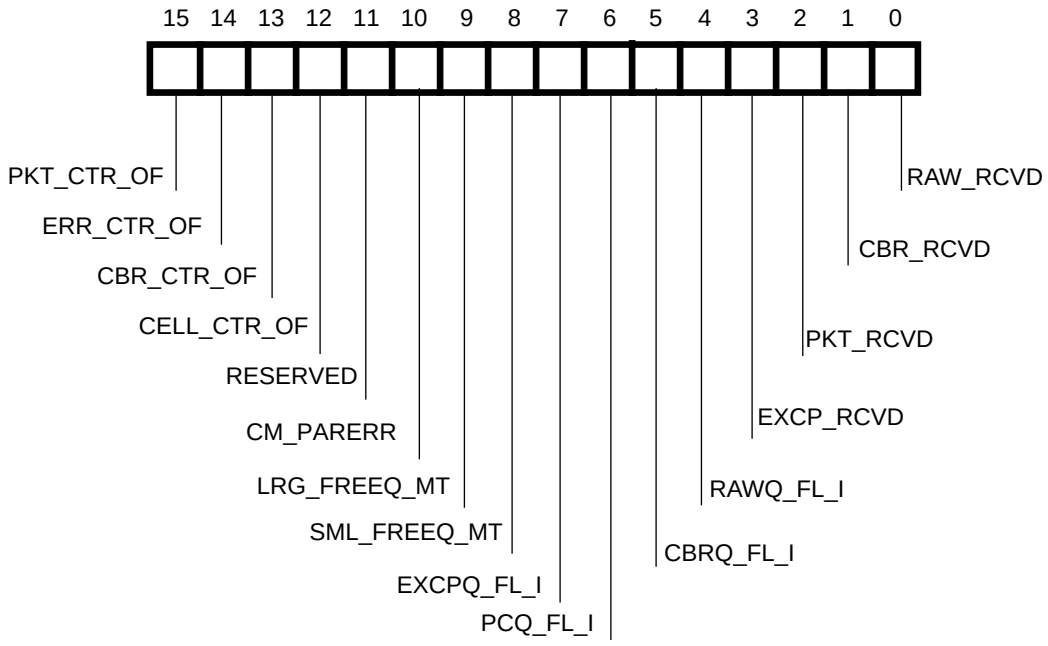


Figure 5-3. Interrupt Status Register Bit Positions

Bit 15: PKT_CTR_OF:

This bit is set when the “dropped packet counter” (DRP_PKT_CNTR) overflows.

Bit 14: ERR_CTR_OF:

This bit is set when the “error counter” (ERR_CNTR) overflows.

Bit 13: CBR_CTR_OF:

This bit is set when the “dropped CBR cells counter” (DRP_CBR_CNTR) overflows.

Bit 12: CELL_CTR_OF:

This bit is set when the 32-bit cell counter overflows.

Bit 11: RESERVED:

This bit is unused in this version of the chip and is read as zero.

Bit 10: CM_PARERR:

This bit, when set, indicates that a parity error has been detected during control memory read operations. Processing of the current cell is aborted, the cell is dropped and the SARA-R starts processing the next cell.

Bit 9: LRG_FREEQ_MT:

This bit, when set, indicates that the large free descriptor queue was empty and that one or more packets have been dropped due to this condition. This condition will occur when an AAL5 packet or a large AAL3/4 packet is received and the large queue is empty or when a small AAL3/4 packet is received and both of the queues are empty.

Bit 8: SML_FREEQ_MT:

This bit, when set, indicates that the small free descriptor queue was empty and that one or more packets have been dropped due to this condition. This condition will occur when a small AAL 3/4 packet is received, and both the small and large queues are empty.

Bit 7: EXCPQ_FL_I:

This bit, when set, indicates that the exception queue was full and that one or more errors have not been reported as a result of this condition.

Bit 6: PCQ_FL_I:

This bit, when set, indicates that the packet complete queue was full and as a result, SARA-R has not been able to write the descriptor associated with a completely reassembled packet to that queue. The reassembled packet will still be held by SARA-R.

Bit 5: CBRQ_FL_I:

This bit, when set, indicates that the constant bit rate traffic queue was full and that one or more CBR cells have been dropped as a result of this condition. This bit should be masked out and ignored when the IGN_CBR_FL mode bit is set.

Bit 4: RAWQ_FL_I:

This bit, when set, indicates that the raw cell queue was full and that one or more raw cells have been dropped as a result of this condition. This bit should be masked out and ignored when the IGN_RAW_FL mode bit is set.

Bit 3: EXCP_RCVD:

This bit, when set, indicates that an exception error has occurred causing the exception queue to become not-empty. The exception queue must become empty before this bit can be set again. Additional details regarding the conditions that result in this bit being set are described in Chapter 7.

Bit 2: PKT_RCVD:

This bit, when set, indicates that a descriptor is written into the packet complete queue, causing the queue to become non-empty. The packet complete queue must become empty before this bit can be set again.

Bit 1: CBR_RCVD:

This bit, when set, indicates that a CBR cell has been received by SARA-R and has been placed in the CBR queue in packet memory causing the queue to become non-empty. The CBR queue must become empty before this bit can be set again.

Bit 0: RAW_RCVD:

This bit, when set, indicates that a raw cell has been received by SARA-R and has been placed in the raw cell queue in packet memory causing the queue to become non-empty. The raw cell queue must become empty before this bit can be set again.

5.3 State Register

Figure 5-4 shows the SARA-R state register bits. These bits do not generate an interrupt. These are read-only status bits indicating the dynamic states of all the queues in control memory and the cell queues in the packet memory. All bits in this register get cleared automatically when the condition causing them to be set goes away.

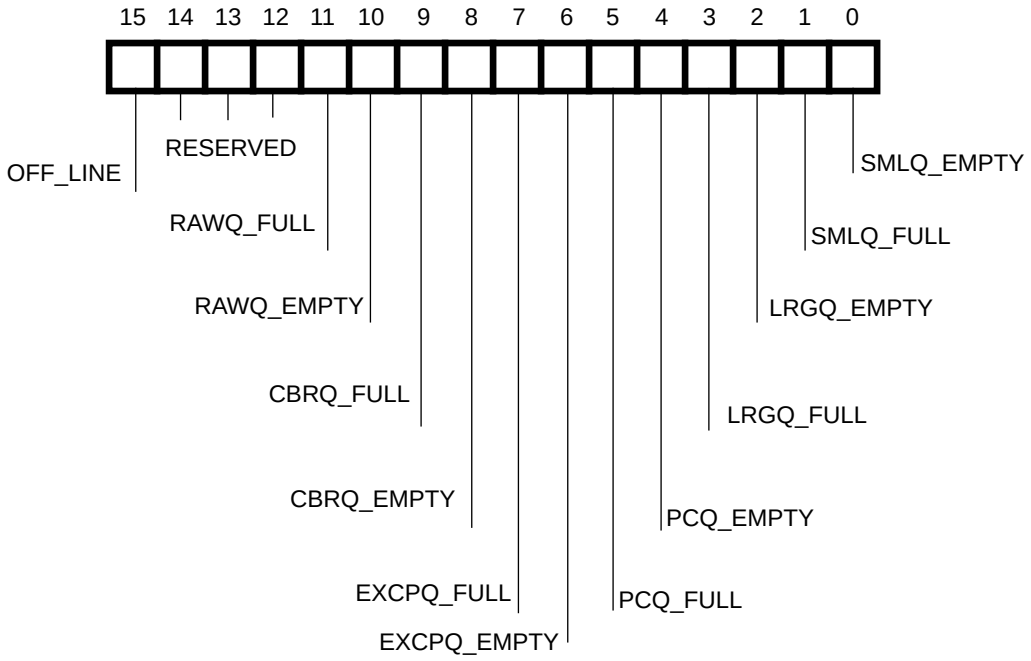


Figure 5-4. SARA-R State Register Bit Positions

Bit 15: OFF_LINE:

This bit indicates the state of the chip and is set to “1” immediately after a hard/soft reset (chip is in off-line state). Packet reassembly and control memory accesses will not be performed until the chip is put back on-line (ON_LINE of Mode Register 0).

When the processor forces the chip to enter the off-line state, this bit must be polled to see if the chip has gracefully entered off-line. If this bit still indicates on-line (“0”), it implies that the chip is still in the middle of a reassembly cycle and has not yet gracefully entered the off-line state (“1”).

During reassembly, this bit will be in the “0” state.

Bit (14:12): RESERVED

These bits are unused and read as “0”.

Bit 11: RAWQ_FULL

This bit indicates that the raw cell queue full condition exists.

Bit 10: RAWQ_EMPTY

This bit indicates that the raw cell queue empty condition exists.

Bit 9: CBRQ_FULL

This bit, when set, indicates that the constant bit rate (CBR) cell queue full condition exists.

Bit 8: CBRQ_EMPTY

This bit, when set, indicates that the constant bit rate (CBR) cell queue empty condition exists.

Bit 7: EXCPQ_FULL

This bit, when set, indicates that the exception queue full condition exists.

Bit 6: EXCPQ_EMPTY

This bit, when set, indicates that the exception queue empty condition exists.

Bit 5: PCQ_FULL

This bit, when set, indicates that the packet complete queue full condition exists.

Bit 4: PCQ_EMPTY

This bit, when set, indicates that the packet complete queue empty condition exists.

Bit3: LRGQ_FULL

This bit, when set, indicates that the large descriptor queue full condition exists.

Bit 2: LRGQ_EMPTY

This bit, when set, indicates that the large descriptor queue empty condition exists.

Bit 1:SMLQ_FULL

This bit, when set, indicates that the small descriptor queue full condition exists.

Bit 0: SMLQ_EMPTY

This bit, when set, indicates that the small descriptor queue empty condition exists.

5.4 Interrupt Mask Register

The SARA-R interrupt mask register is written by the host/processor in order to prevent the generation of interrupts caused by events recorded in the status register. Each mask register bit, when set, masks out the interrupt generated by the corresponding bit in the status register. Figure 5-5 shows the mask register bit positions.

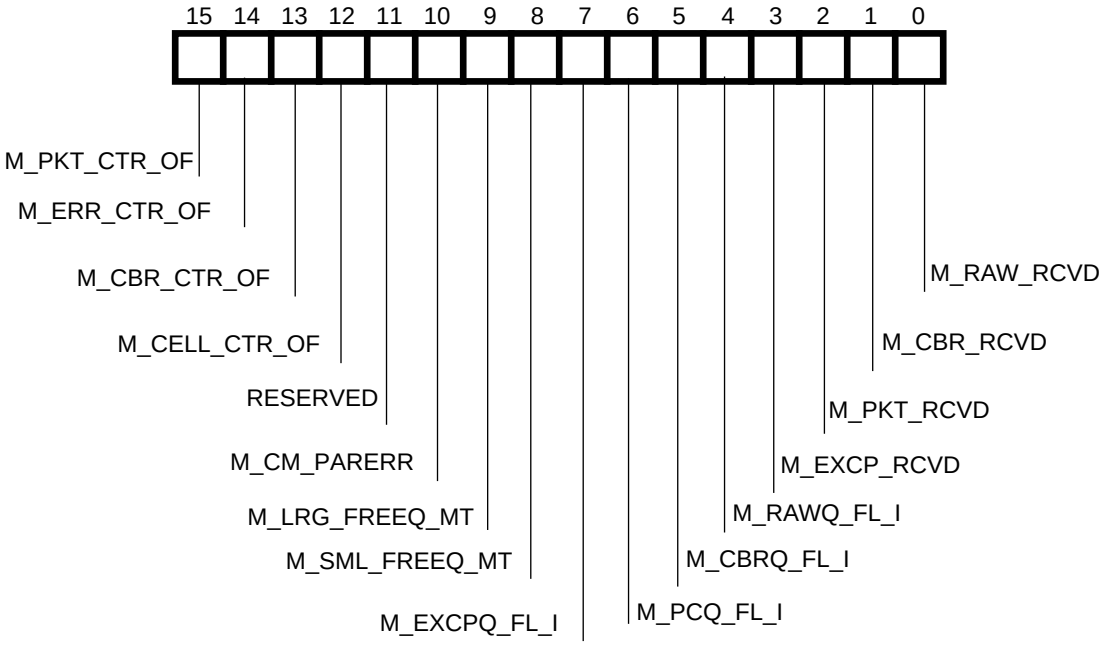


Figure 5-5. Mask Register Bit Positions

Bit 15: M_PKT_CTR_OF

0 - Unmask interrupt generation because of PKT_CTR_OF

1 - Mask interrupt generation because of PKT_CTR_OF

Bit 14: M_ERR_CTR_OF

0 - Unmask interrupt generation because of ERR_CTR_OF

1 - Mask interrupt generation because of ERR_CTR_OF

Bit 13: M_CBR_CTR_OF

0 - Unmask interrupt generation because of CBR_CTR_OF

1 - Mask interrupt generation because of CBR_CTR_OF

Bit 12: M_CELL_CTR_OF

0 - Unmask interrupt generation for cell counter overflow

1- Mask interrupt generation for cell counter overflow

Bit 11: RESERVED

This bit is unused and must be programmed to “1” for future compatibility

Bit 10: M_CM_PARERR

0 - Unmask interrupt generation because of CM_PARERR

1 - Mask interrupt generation because of CM_PARERR

Bit 9: M_LRG_FREEQ_MT

0 - Unmask interrupt generation because of LRG_FREEQ_MT

1 - Mask interrupt generation because of LRG_FREEQ_MT

Bit 8: M_SML_FREEQ_MT

0 - Unmask interrupt generation because of SML_FREEQ_MT

1 - Mask interrupt generation because of SML_FREEQ_MT

Bit 7: M_EXCPQ_FL

0 - Unmask interrupt generation because of EXCPQ_FL_I

1 - Mask interrupt generation because of EXCPQ_FL_I

Bit 6: M_PCQ_FL_I

0 - Unmask interrupt generation because of PCQ_FL_I

1 - Mask interrupt generation because of PCQ_FL_I

Bit 5: M_CBRQ_FL_I

0 - Unmask interrupt generation because of CBRQ_FL_I

1 - Mask interrupt generation because of CBRQ_FL_I

Bit 4: M_RAWQ_FL_I

0 - Unmask interrupt generation because of RAWQ_FL_I

1 - Mask interrupt generation because of RAWQ_FL_I

Bit 3: M_EXCP_RCVD

0 - Unmask interrupt generation because of EXCP_RCVD

1 - Mask interrupt generation because of EXCP_RCVD

Bit 2: M_PKT_RCVD

0 - Unmask interrupt generation because of PKT_RCVD

1 - Mask interrupt generation because of PKT_RCVD

Bit 1: M_CBR_RCVD

0 - Unmask interrupt generation because of CBR_RCVD

1 - Mask interrupt generation because of CBR_RCVD

Bit 0: M_RAW_RCVD

0 - Unmask interrupt generation because of RAW_RCVD

1 - Mask interrupt generation because of RAW_RCVD

5.5 Queue Base Address Register

This register contains the base address of the four communication queues in control memory. This base address, when concatenated with the queue read or write pointer, generates the control memory address to the appropriate queue entry. Figure 5-6 shows the address generation.

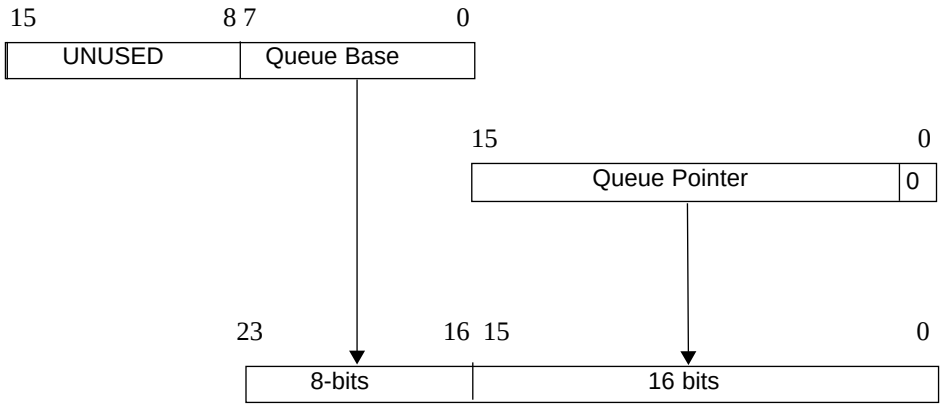


Figure 5-6. SARA-R Queue Address Generation

5.6 Descriptor Table Base Address Register

This register contains the base address of the buffer descriptor table. The value in the descriptor base address, when concatenated with the descriptor number and the table offset, points to the descriptor entry in control memory. Each buffer descriptor table entry is 32 bytes long. The offset determines the specific 16-bit word within the table entry being accessed. Figure 5-7 shows how the descriptor table address is formed.

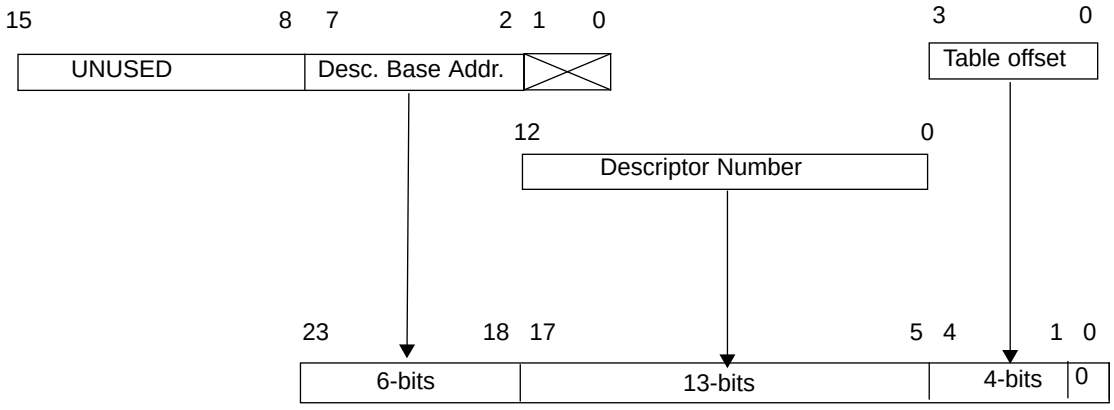


Figure 5-7. SARA-R Descriptor Table Address generation

5.7 VC Table Base Address Register

This register contains the base address of the VC table. The value in this VC_LKUP_BASE register is concatenated with the lower bits of the VCI to generate the VC table address.

The least significant three bits of this base register determines the number of bits of the base register used in generating the address. This is useful in configuring the control memory for various table sizes and is illustrated in Figure 5-8. Table 5-3 shows the maximum number of VC Table entries possible for various values in the least significant 3-bits of the VC_LKUP_-BASE register. The lower order bits of the VCI are used to determine the memory address when the least significant bits of VC_LKUP_BASE is programmed to use less than 16 bits.

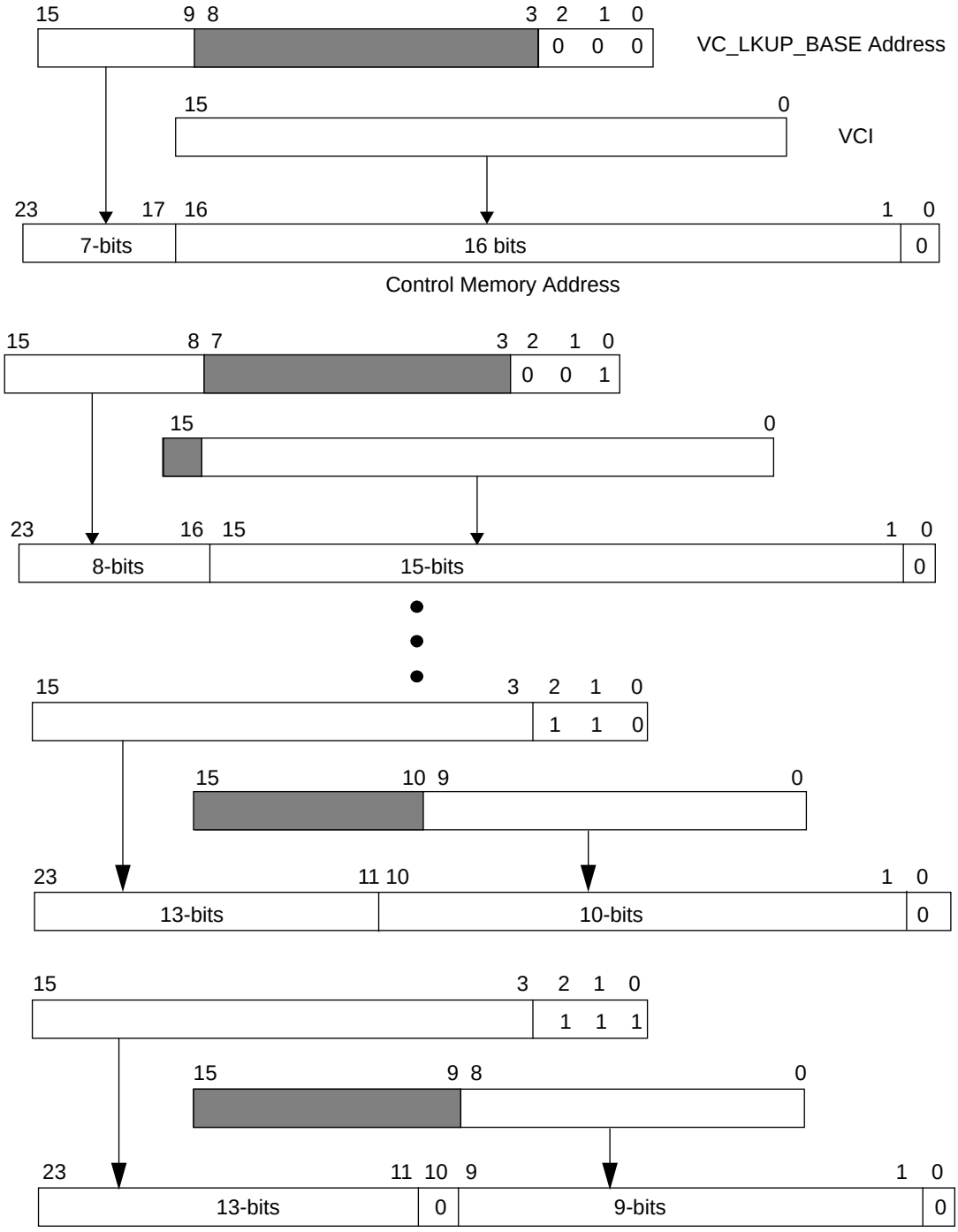


Figure 5-8. SARA-R VC Table Address Generation

Least Significant Bits of VC Base Register	Memory address		Maximum Number of VCs Possible
	Number of bits used from VC_LKUP_BASE	Number of bits used from VCI	
000	7	16	65,536
001	8	15	32,768
010	9	14	16,384
011	10	13	8,192
100	11	12	4,096
101	12	11	2,048
110	13	10	1,024
111	13	9	512

Table 5-3. Number of VC Table Entries for Various Values of the VC_LKUP_BASE Register

5.8 VP Base Address Register

This register contains the base address of the VP table. This base address is concatenated with the eight-bit VPI field to index the VP table. This address generation is illustrated in Figure 5-9.

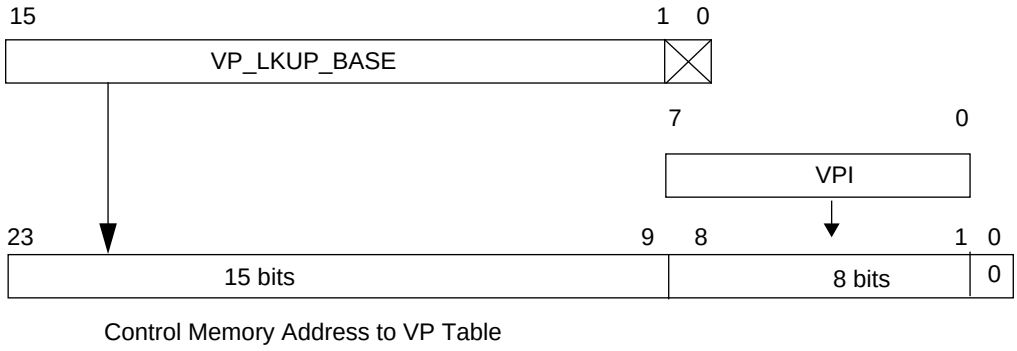


Figure 5-9. SARA-R VP Base Address Register

5.9 VP Filter Register

Figure 5-10 shows a diagram of the VP Filter Register. This register is used to determine whether reassembly will be done on VPI/VCI or VCI/MID. Depending on the result, either the VP or VC table will be used. The VP Filter Register uses the most significant eight bits (15:8) as mask bits, and the least significant eight bits (7:0) as compare bits. The compare bits are Exclusive NORed with the VPI; this result in turn is ORed with the mask bits. This result is then compared with eight ones; if the result compares, the assembly is on VCI and MID (optionally), while if the results do not compare the assembly is on VPI and VCI.

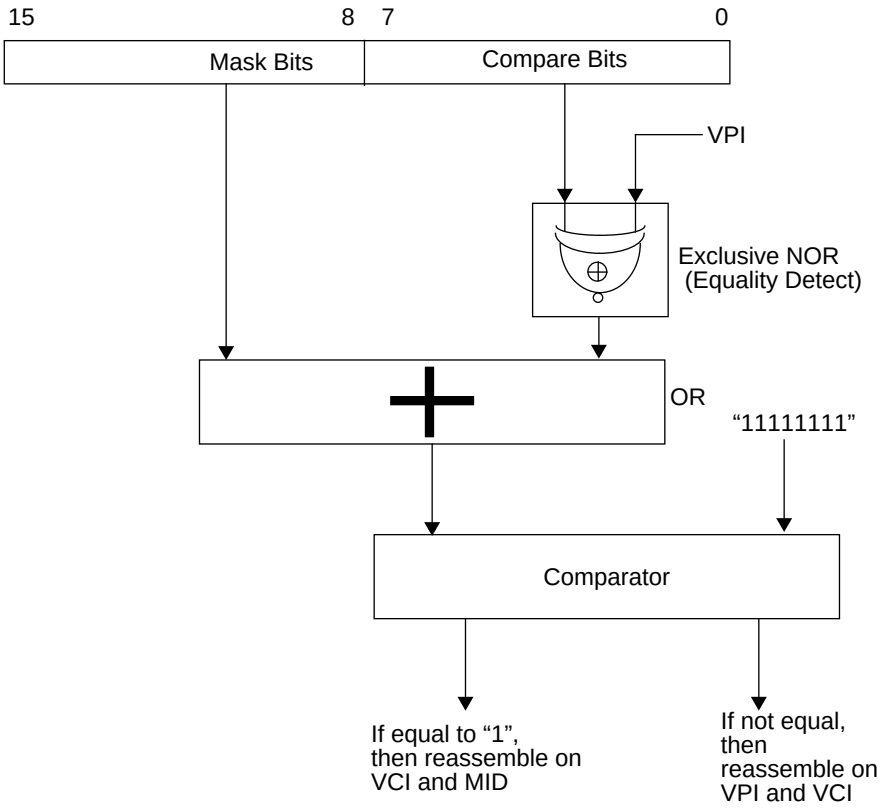


Figure 5-10. VP Filter Register

5.10 Reassembly Table Base Register

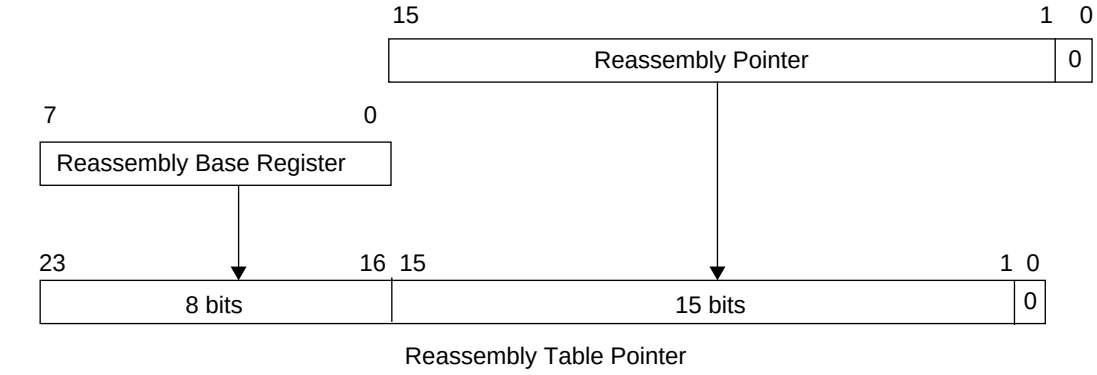
If bit 0 of the reassembly pointer from the VC or VP table equals “0”, then concatenating the 16 bits of the reassembly pointer with eight bits of the reassembly base registers gives the 24-bit reassembly table pointer. This case is shown in the top of Figure 5-11 and is used on circuits requiring VPI-only (i.e. VP table entry LSB=0) reassembly. 256 VPI-only and up to 32,768 VCI-only circuits are supported.

If, on the other hand, bit 0 of the reassembly pointer from the VC or VP table equals “1”, then bits 1, 2, and 3 of the reassembly pointer determine the number of bits which will be used for VPI/VCI reassembly (VP table) or VCI/MID reassembly (VC table). As is shown in the lower two diagrams of Figure 5-11, the reassembly table pointer is formed by concatenating the upper seven bits of the reassembly base register with the number of MSB of the reassembly pointer and the number of LSB of the MID (or VCI) as specified by Table 5-4.

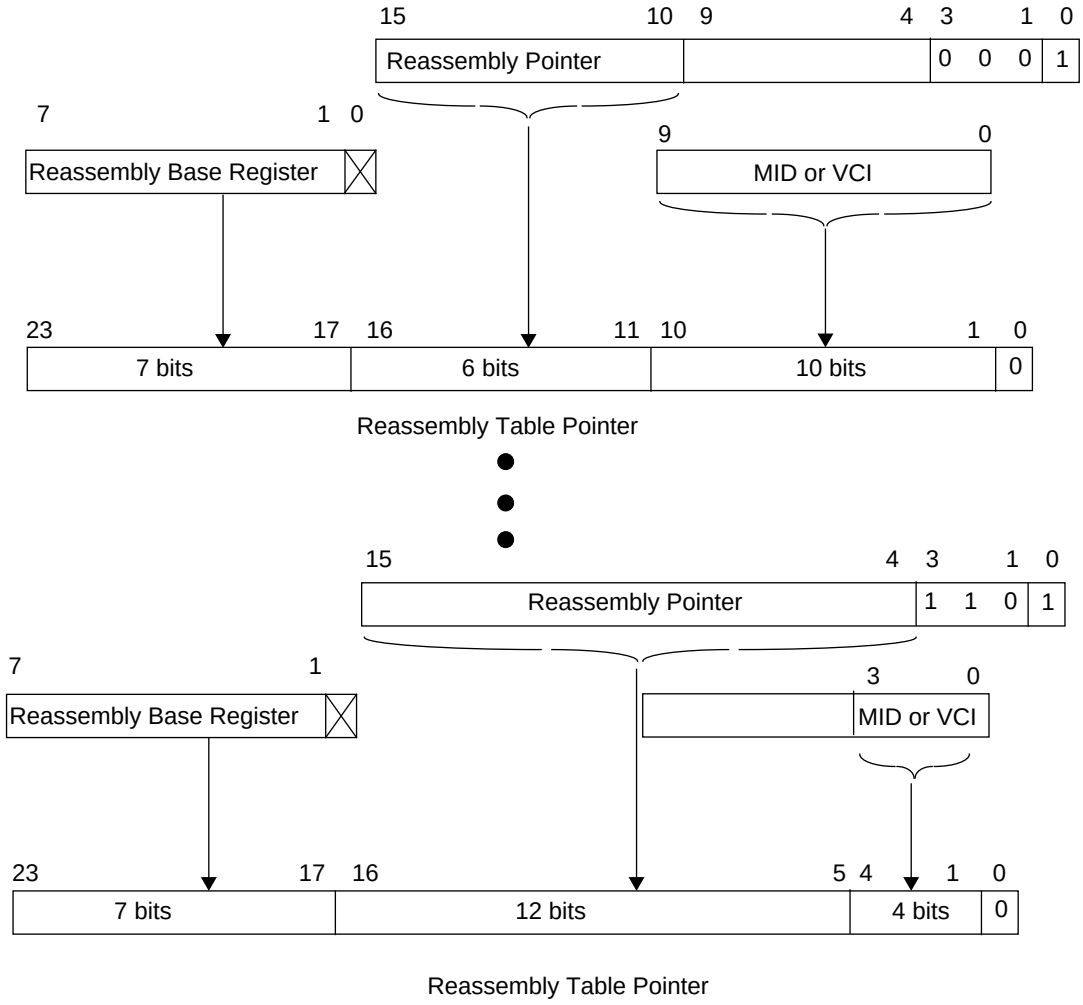
Bits 3, 2 and 1 of Reassembly Pointer	Number of Reassembly Pointer MSB Bits to be Concatenated	Number of Least Significant Bits from MID/VCI[1]
000	6	10
001	7	9
010	8	8
011	9	7
100	10	6
101	11	5
110	12	4
111	Invalid VCI/VPI	

Table 5-4. Reassembly Table Pointer Construction

1. This represents the bits of the MID field for VCI/MID reassembly and VCI field for VPI/VCI reassembly.



a: Bit 0 of reassembly pointer =0



b: Bit 0 of reassembly pointer =1

Figure 5-11. Reassembly Table Pointer Construction

5.11 Packet Memory Address Match Register

During packet memory cycles the packet memory long word address is compared to the value programmed in the packet memory address register. Upon match, the signal PAMTCH is forced active. The external memory system can use this signal to terminate the memory cycles. This is helpful when memory cycle termination is required when crossing a page boundary (Example: DRAM page boundary).

Figure 5-12 describes the operation of this register. Bits (7:0) of this register are always compared with bits (9:2) of the packet memory address. The contents of bits (15:12) are used as mask bits to determine whether bits (11:8) are also used in the comparison. Packet memory address bits (1:0) are always ignored for generating the PAMTCH output.

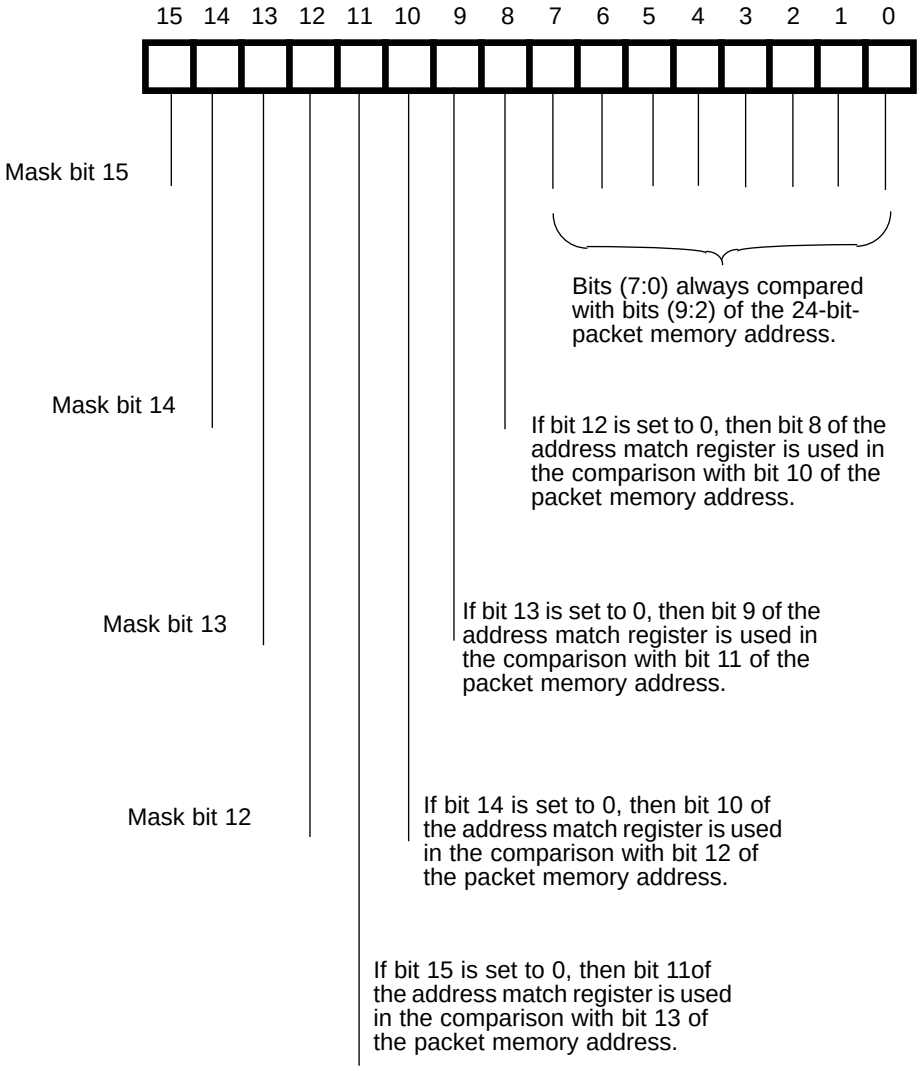


Figure 5-12. SARA-R Packet Memory Address Match Register

5.12 Command Register

The command register is a write-only register used to issue software commands to SARA-R as follows:

- reset SARA-R when xx55 is written to this register
- reset SARA-R state machine when xxAA is written to this register
- reset SARA-R DRP_PKT_CNTR when xxF1 is written to this register
- reset SARA-R ERR_CNTR when xxF2 is written to this register
- reset SARA-R DRP_CBR_CNTR when xxF4 is written to this register
- reset SARA-R CELL_CNTR_LO/HI when xxF8 is written to this register
- reset all the above counters when xxFF is written to this register

Values written to this register other than those listed above will be ignored.

5.13 Dropped Packet Count Register

This 16-bit counter indicates the number of packets dropped by SARA-R because of a shortage of available free buffers in packet memory. The counter is cleared when read by the host/processor at address DRP_PKT_CNTR and not cleared when read at address DRP_PKT_CNTR_NC. It can be written by the host/processor in controlled write mode for diagnostic purposes. A bit is set in the status register when this counter overflows.

5.14 Error Count Register

This 16-bit counter indicates the number of cells received with errors (such as HEC errors, CRC errors). This counter is cleared when read by the processor at address ERR_CNTR and not cleared when read at address ERR_CNTR_NC. It can be written by the processor in controlled write mode for diagnostic purposes. A bit is set in the status register when this counter overflows.

5.15 Dropped CBR Cells Count Register Queue

This 16-bit counter indicates the number of constant bit rate cells dropped by SARA-R because of lack of space in the CBR queue in packet memory. When IGN_CBR_FL is set, this counter counts the number of cells overwritten when the CBR queue is full. It can be written to by the host/processor in controlled write mode for diagnostic purposes. A bit is set in the status register when this counter overflows.

5.16 Received Cell Count Register

This 32-bit counter counts the total number of non-error and non-idle cells received. This is accessible by reading two 16-bit register locations. The counters are auto-cleared when they are read using addresses CELL_CNTR_LO and CELL_CNTR_HI, and not cleared when read from addresses CELL_CNTR_LO_NC and CELL_CNTR_HI_NC. For an accurate count, it is recommended that CELL_CNTR_LO be read before reading CELL_CNTR_HI. The sum of the error count register and the cell count register is equal to the total number of non-idle cells received.

5.17 Small Buffer Check Register

The SML_BUF_CHK register contains a value depending upon the size (in bytes) of the small buffers in packet memory (AAL 3/4). As long as the received packet can fit into this buffer (as determined by the BA size field in the Beginning of Message (BOM) or Single Segment Message (SSM) cell received) SARA-R will use small buffers for the packets. This register should be programmed to 52 bytes less than the small buffer size. This is necessary because the BA size doesn't include the eight bytes of CS_PDU header and trailer, and the SARA-R always writes the entire last cell into memory. On circuits programmed for AAL 5, large buffers are always used.

5.18 Large Buffer Check Register

The LRG_BUF_CHK register contains a value depending upon the size (in bytes) of the large buffers in packet memory for AAL 3/4 circuits. When the received packet cannot fit into the small buffer or when there are no more small buffers available, SARA-R will use the large buffers for the packets. This register should be programmed to 52 bytes less than the large buffer size. This is necessary because the BA size doesn't include the eight bytes of CS_PDU header and trailers, and the SARA-R always writes the entire last cell into memory.

5.19 Large Buffer Size Register

The LRG_BUF_SIZ register contains the size (in bytes) of the large buffers in packet memory for AAL 5 circuits. It is programmed to the actual large buffer size. For applications using both AAL 3/4 and AAL 5, the large buffer check register is programmed to 52 bytes less than this register.

5.20 Packet Timeout Count Register

The PKT_TM_CNT register is used for setting-up the packet timeout interval. The upper and lower eight bits of this register serve different functions.

The upper eight bits are used to initialize the packet timeout counter field in the descriptor table when a buffer is first assigned to a packet.

The lower eight bits are used to define the terminal count of the packet aging interval counter, which is used to determine how often a packet is aged.

5.21 Packet Aging Interval Count Register

The interval counter (INTRVL_CNTR) is a free-running 12-bit counter used to age packets. The upper eight bits are refreshed from the lower eight bits of the packet timeout count register. The lower four bits are initialized to "0" when refreshed.

Whenever this interval counter overflows, the timeout index register is incremented. At that time, the active descriptor associated with the VC pointed to by the timeout index is stamped with a new time stamp. Aging, if any, is performed by incrementing the packet timeout counter field in the descriptor table entry. An overflow of that counter will terminate the packet reassembly process of that descriptor and the descriptor will be returned to the processor by writing it into the packet complete queue with an error status bits set in the descriptor table entry.

This counter is made accessible for diagnostic purposes.

5.22 Timeout Index Register

The TMOUT_INDX register contains the reassembly pointer whose packet is being aged. It is incremented each time the packet aging interval counter overflows. The aging process will occur only if a packet is active in the reassembly table.

This register is made accessible for diagnostic purposes.

5.23 Timeout Index Range Register

The reassembly table can be set up anywhere in memory by appropriately initializing the VP and VC tables. The timeout index range register (TMOUT_RANGE) is used to define the range of the reassembly table that is scanned to age-out packets in the middle of reassembly. A diagram of the Timeout Index Range Register is shown in Figure 5-13.

The range is defined in increments of 256 locations of the reassembly table.

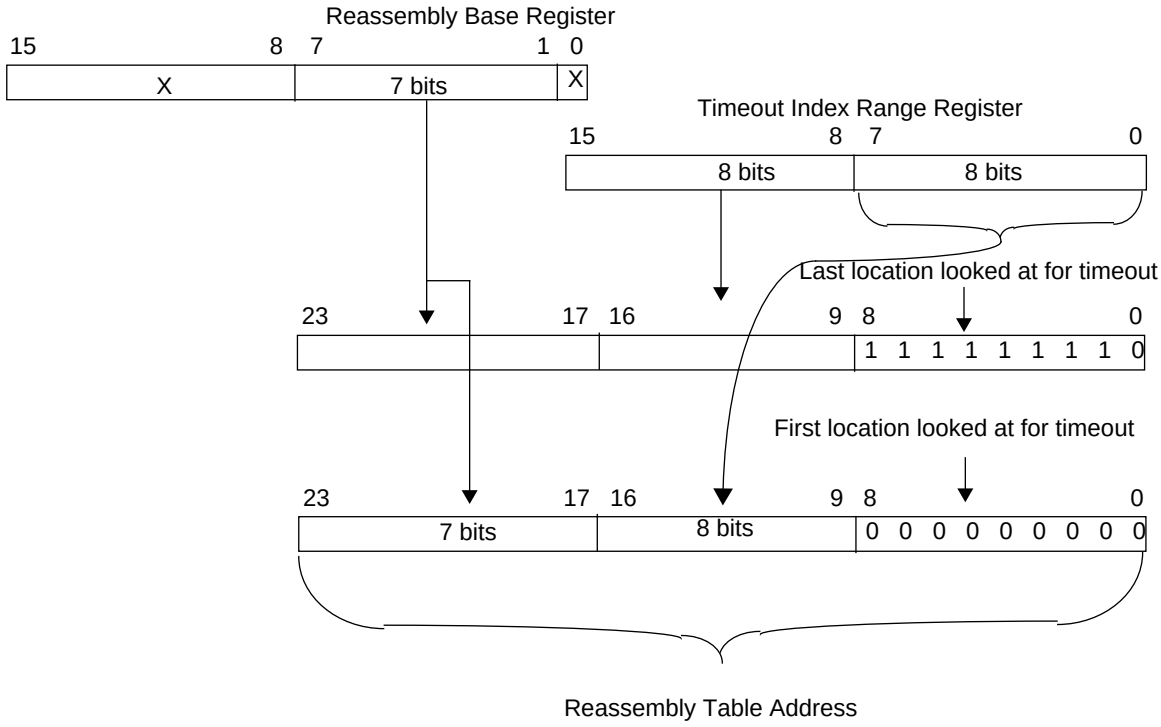


Figure 5-13. Timeout Index Range Register

5.24 Communication Queues

SARA-R uses four queues to communicate with the software/processor for passing descriptors - small free descriptor queue, large free descriptor queue, packet complete queue and the exception queue. Each queue consists of:

- starting address register
- ending address register
- read pointer
- write pointer

The queue length (difference between the end address and the start address) should be programmed greater (at least by one entry) than the maximum number of descriptors used to prevent any possible queue overflow.

Figure 5-14 shows the operational states of the queues during various conditions. The queues operate as follows: At initialization, the pointers should be programmed to be read pointer (RP) = write pointer (WP) = starting address (SA). Any time RP=WP, the queue is empty. Descriptors are added to the queue by incrementing the WP by one entry length. The WP wraps around when it exceeds the end address (EA). The SARA-R automatically wraps the pointer (WP or RP) as it increments this pointer. Software is responsible for detecting the EA boundary and wrapping the pointer to SA. When the WP is one entry short of the RP, the queue full condition is reached.

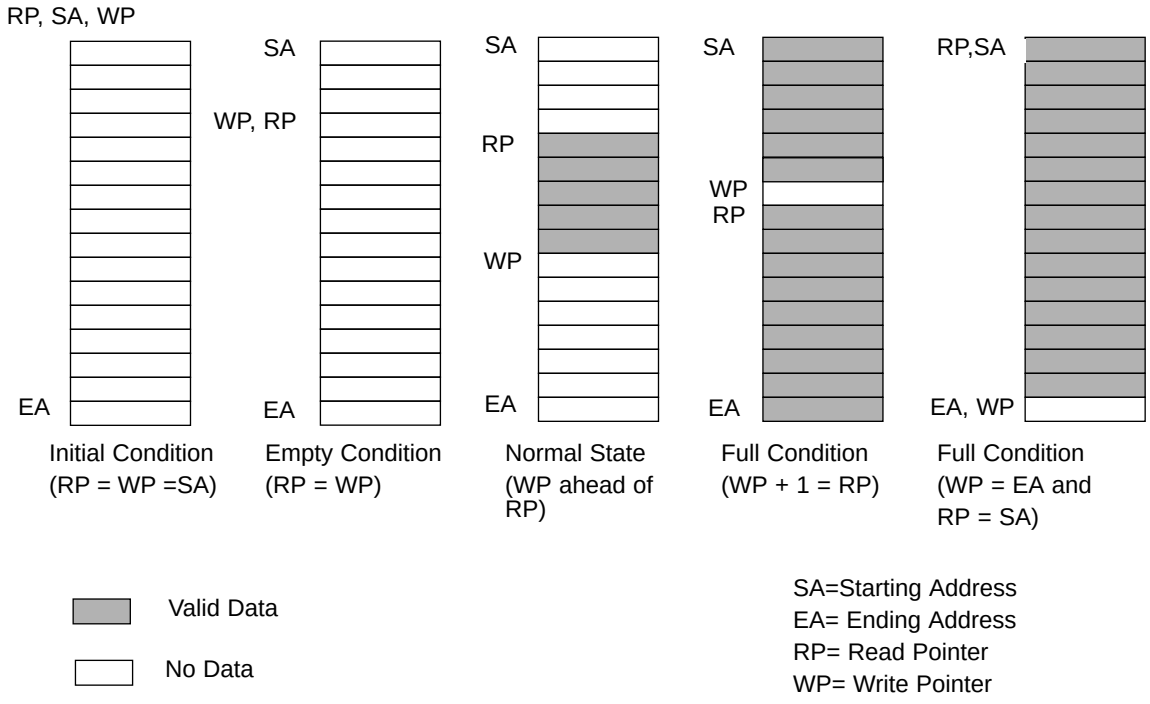


Figure 5-14. SARA-R Communication queues - Operational States

5.24.1 Small Free Descriptor Queue

This queue is filled by the software with the descriptor numbers of the small buffers that are available for packet reassembly.

SARA-R reads a descriptor from this queue whenever a packet requiring a small buffer is being re-assembled.

5.24.2 Large Free Descriptor Queue

This queue is loaded by the software with the descriptor numbers of the large buffers that are available for packet reassembly.

SARA-R reads a descriptor from this queue whenever a packet requiring a large buffer is being re-assembled. It also fetches a descriptor from this queue if a packet requiring a small buffer is being received and the small free descriptor queue is empty. AAL 5 packets always use large buffers.

5.24.3 Packet Complete Queue

This queue is loaded by SARA-R with the descriptor numbers of the buffers that contain completely re-assembled packets.

5.24.4 Exception Queue

This queue is loaded by the SARA-R with the VCI of the cell that caused the exception.

Figure 5-15 shows an entry in the exception queue.

The following error codes are defined:

- 000 - No error

- 001 - Out of sequence COM message received
- 010 - Out of sequence EOM message received
- 100 - No free descriptor available and packet has been dropped
- 101 - No large descriptor available and packet has been dropped.
- 110 - Cell received on invalid VC.
- 111 - Cell received on invalid VP.

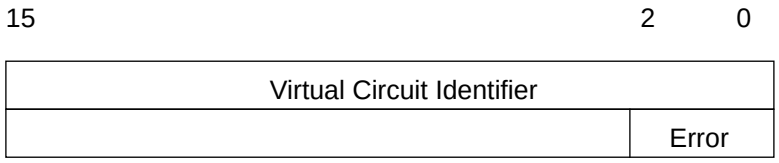


Figure 5-15. Entries in Exception Queue

5.25 CBR and Raw Cell Queue Base Address Registers

This register contains the upper eight bits of the CBR and the raw cell queue location in the packet memory. These bits, when concatenated with the 16-bit write pointer of the corresponding queue, point to the beginning of the cell entry. This is illustrated in Figure 5-16.

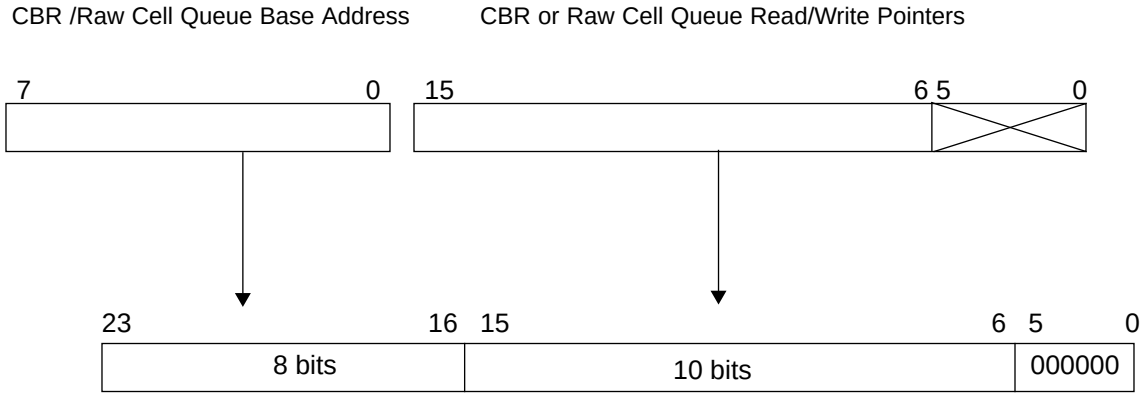


Figure 5-16. CBR and Raw Cell Queue Address Generation

5.26 CBR Queue Registers

The CBR queue in packet memory contains the cells received on a VC index that has been programmed as carrying CBR traffic (as indicated by an entry in the VC lookup table). This queue is similar in definition to the communication queues described in Section 4.11. It is defined by the start and end address registers along with the read and write pointers.

Each entry contains 64 bytes of data of which only the first 52 bytes are used to store the first four bytes of the header (the HEC is discarded after checking) and the 48 bytes of payload associated with the CBR cell.

5.27 Raw Cell Queue Registers

The raw cell queue in packet memory contains the received raw cells, OAM F5, or congestion notification cells. OAMF5 cells are indicated by the payload type field in the ATM header. This queue is similar in definition to the communication queues described in Section 4.11. It contains the start and end address registers along with the read and write pointers.

Each entry is 64 bytes long, of which only the first 52 bytes are used to store the first four bytes of the header and the 48 bytes of payload associated with the raw cell.

5.28 OAMCC_CHK Register

This register is used to check the first byte of OAM F5 end-to-end cells to determine if the cell is a congestion notification cell. This is illustrated in Figure 5-17. The OAMCC_CHK register uses the most significant eight bits (15:8) as mask bits, and the least significant eight bits (7:0) as compare bits. The compare bits are Exclusive NORed with the first byte of the OAM F5 cell payload; this result in turn is ORed with the mask bits. This result is then compared with eight ones; if the result compares, the cell is a congestion notification cell; if the results do not compare the cell is not.

This check is enabled by the bit CC_OAMF5_EN in Mode Register 1.

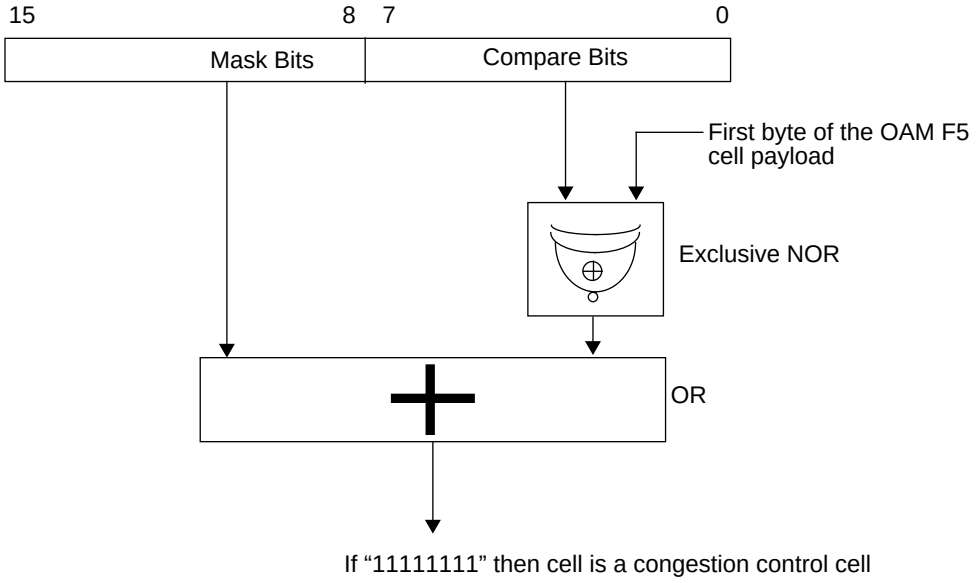


Figure 5-17. OAMCC_CHK Register

Chapter 6. SARA-S Software Interfaces

This chapter describes the various memory data structures, communication queues and software procedures for using the SARA-S chipset.

6.1 SARA-S Control Memory

The control memory contains data structures which are used by both software and the SARA chipset to maintain dynamic information. It is also used to communicate between the SARA's and the software entity. Each of the SARA devices can have a separate control memory or can share common memory.

The SARA-S control memory holds the buffer descriptor tables, the VC table, the packet ready queue and the transmit complete queue. Each of these is described in the following sections.

6.1.1 Buffer Descriptor Table

The buffer descriptor table is a set of 32-byte entries. Each entry contains parameters for segmenting packets and the location of the packet in the packet memory. When a packet is set up for segmentation, an available buffer descriptor is initialized with the packet parameters, as explained later in this chapter.

The upper six bits of the base location of the buffer descriptor table in memory is stored in the descriptor base address register (DESC_BASE). Each buffer descriptor table entry is referenced by its descriptor number which can range from 1 to 8,191. Descriptor number 0 is used internally by the SARA-S and must not be used. The location of the entry is determined by concatenating the descriptor number with the descriptor base address (as shown in section 4.7).

Software can assign the actual descriptor numbers anywhere within the range of 0 < Descriptor Number < 8,192. Therefore, the buffer descriptor table can be mapped into a desired region(s) of the control memory by allocating the buffer descriptor numbers as shown in Figure 6-1. The open memory regions may be used for other tables or queues.

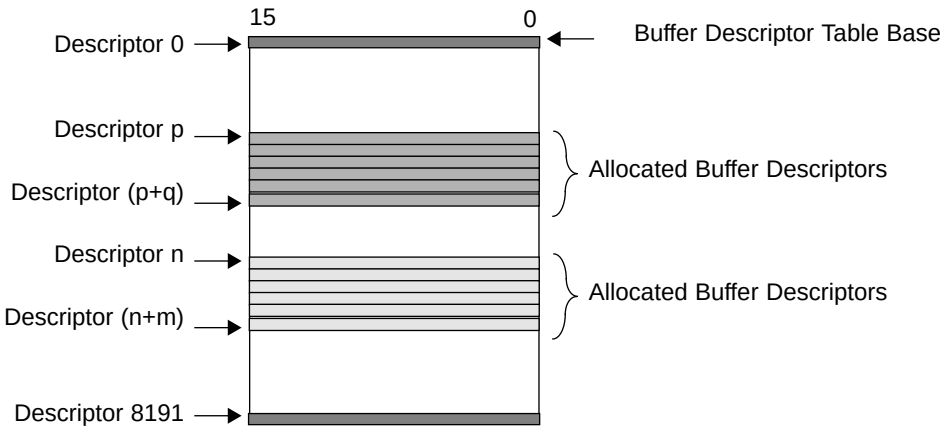


Figure 6-1. Buffer Descriptors Mapped into Two Regions

An example of the determination of usable descriptor numbers, when a specific region of memory is available, is given below:

Example (refer to Figure 4-8)

The region of available memory is:

Start address of region is 120800H

End address of region 121000H

The size of the available memory is 0800H in which 40H (0800H/32) descriptors can be assigned.

The Descriptor base register will be programmed with the value:

((120800H & FC0000H) >> 16) = 100000H (Mask off the lower 18 bits)

The location of the first descriptor from the base address is:

120800H & 03FFFFH = 020800H (Lower 18 bits of start address)

which implies that the first descriptor number is:

(020800H/32) = 1040H

and correspondingly, the last descriptor number is

1040H + 40H -1 = 107FH

The organization of each descriptor table entry is shown in Figure 6-2, including the type of accesses by the software entity and the SARA-S and the address of the individual entries in the table. Note that the addresses CH to 16H are used by SARA-S and must not be used for another purpose by software.

Address	15 … 0	SW	SARA-S
0H	Descriptor Mode Bits	W	R/W
2H	VC Table Index	W	R
4H	0 0 0 0 0 0 | Message Identifier (MID)	W	R
6H	Packet Byte Count	W	R/W
8H	Packet Memory Start Address (High)	W	R
AH	Packet Memory Start Address (Low)	W	R
CH - 16H	Reserved for SARA-S use		R/W
18H	Reserved for future SARA-S use		
1AH - 1EH	UNUSED		

Figure 6-2. Descriptor Table Entry Organization

15 … 13	12	11	10	9 8	7 … 0
PTI_VALUE	XD_INTT	EOM_EN	APP_CRC32	PKT_TYPE	0 0 0 0 0 0 0 0

Figure 6-3. Descriptor Mode Bits

Figure 6-3 shows the organization of the descriptor mode bits. The individual descriptor mode bits are described below:

Bits (15:13): PTI_VALUE:

These bits are the substitute PTI_VALUE used for the ATM cell header when the PKT_TYPE field indicates that an OAM cell type is to be transmitted with the PTI values from the descriptor table (PKT_TYPE = 11).

Bit 12: XD_INTT:

This bit, when set, indicates that SARA-S must generate a maskable interrupt after the frame associated with this descriptor has been completely segmented and delinked.

Bit 11: EOM_EN:

This bit, when set, indicates that SARA-S must set the EOM bit in the ATM header by forcing the PTI field to xx1 for EOM and SSM cells for AAL3/4 and AAL5 cells (Descriptor PKT_-TYPE = 00 or 01). This bit must be set to 1 for AAL-5.

Bit 10: APP_CRC32:

This bit, when set, indicates that SARA-S must generate and include a 32-bit frame check sequence (FCS) for the frame associated with this descriptor. Note that a bit can alternately be set in the VC table for similar effect. This bit provides FCS control on a per frame basis where as the bit in the VC table provides FCS control on a per connection basis. The FCS is generated and included in the frame as discussed in the sequence in section 1.2.1**.**

Bit (9:8): PKT_TYPE:

These two bits indicate the segmentation algorithm desired. They are used to setup the descriptor entry for AAL3/4, AAL5, or OAM (PTI value from either the descriptor Table or the VC table):

00 - AAL3/4 segmentation performed

01 - AAL5 segmentation performed

10 - An OAM cell is transmitted using PTI values from the VC table

11 - An OAM cell is transmitted using PTI values from the descriptor table.

The formats for both the normal data and OAM cells are explained later when discussing the packet memory.

Bit (7:0): RESERVED:

These bits MUST be initialized to “0” by software before passing descriptors to SARA-S. SARA-S uses this area to store temporary variables; unpredictable results will occur if these are not initialized to “0”.

VC Table Index:

This field determines the virtual circuit to which the packet associated with the descriptor belongs. This 16-bit field is used as a pointer to the VC table entry corresponding to the virtual circuit.

Message Identifier:

This 10-bit field (bit 9 through bit 0) is used by the SARA-S as the MID field of the AAL3/4 adaptation layer in the ATM cell.

Packet Byte Count:

This field is programmed with the length in bytes of the packet queued for segmentation. The packet byte count is required to be a multiple of four bytes (i.e. the lower two bits must be zero). The SARA-S decrements this field down to zero during the segmentation process.

Packet Memory Start Address:

The packet starting byte address in packet memory address space, relative to the SARA-S, is generated by concatenating the high and low fields of the packet memory start address. The contents of this field must be initialized to the byte-address of the beginning of the packet buffer in the packet memory. This address must also be 32-bit aligned.

6.1.2 Virtual Circuit (VC) Table:

The virtual-circuit table is comprised of 16-byte entries in the control memory that contain information relating to each virtual connection. When a virtual connection is established, the contents of the entry corresponding to the virtual circuit are initialized by software. The reserved fields should not be modified. The SARA-S uses the information in the VC table in generating the cell headers and for performing the average metering and congestion control functions on the virtual circuit.

The upper 11 bits of the VC table base location in the memory and the size of the table are stored in the virtual circuit base address register (VC_BASE). Each VC table entry is referenced by the corresponding VC Table Index (VC Index). The location of the entry is determined by concatenating the VC index with the VC base address (as shown in Section 4.8).

The organization of each VC table entry is shown in Figure 6-4 including the type of accesses by the software entity, and SARA-S along with the address of the individual entries in the table.

Address	15 … 8	7 … 5	0	SW	SARA-S
0H	Reserved				
2H	Reserved				
4H	ATM Header 0 (hdr0)	ATM Header 1 (hdr1)		W	R
6H	ATM Header 2 (hdr2)	ATM Header 3 (hdr3)		W	R
8H	Virtual Circuit Mode Bits			W	R
AH	Reserved				R/W
CH	Reserved	Cell Quota		W	R/W
EH	Reserved				R/W

Figure 6-4. Virtual Circuit Table Entry

ATM Header 0/1/2/3

These bytes compose the ATM cell header of cells generated on the corresponding virtual circuit. The SARA-S uses these as the ATM cell header without modification (with the optional exception of the payload type field). Software should set these fields to conform to ATM standards. The SARA-S transmits hdr0 followed by hdr1, hdr2, hdr3, and computes the header error checksum (HEC) on these bytes when the cell is transmitted.

Virtual Circuit Mode Bits

The organization of the virtual-circuit mode bits is shown in Figure 6-5.

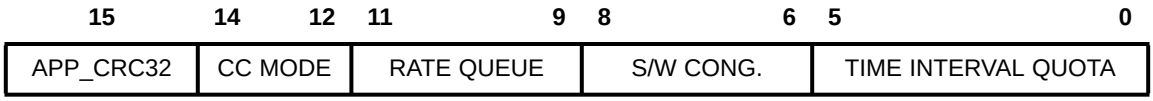


Figure 6-5. Virtual Circuit Mode Bits Word

The individual VC mode bits are described below:

APP_CRC32:

If this bit is set, a frame-level FCS (32-bit CRC) is generated and included for all AAL3/4 and AAL5 frames as described in section 1.2. Note that a bit can alternately be set in the descriptor table for similar effect. This bit provides FCS control on a per connection basis where as the bit in the descriptor table provides FCS control on a per frame basis.

Congestion Control (CC) Mode

This 3-bit field defines the rate of recovery (of the cell rate) after the virtual circuit has been subjected to congestion control. Recovery takes place when the congestion level maintained by the SARA-S (on a per VC basis) is changed to the next lower level as a result of congestion. These mode bits define the number of cell transfers that must take place without encountering a congestion notification cell to cause recovery of the cell rate.

When the CC MODE field is set to “110,” the segmentation rate is determined by the contents of the S/W CONG field.

If the mode is set to flush packets queued for transmission on this VC, the SARA-S will remove the packets from the rate queue and release the descriptors to the host by writing it into the transmit complete queue with the appropriate completion code. This mode can be used by software to purge packets on a VC. These modes are defined in Table 6-1.

CC Mode	Recovery Action
000	Recover after every one cell transfer
001	Recover after every two cell transfers
010	Recover after every four cell transfers
011	Recover after every eight cell transfers
100	Recover after every sixteen cell transfers
101	Reserved
110	Segmentation rate controlled by software
111	Flush packets on this VC

Table 6-1. Congestion Control Mode

Rate Queue

This 3-bit field links the virtual circuit to one of the eight rate queues, as defined in Table 6-2.

Code	Rate Queue
000	High priority bank (A) queue “0”
001	High priority bank (A) queue “1”
010	High priority bank (A) queue “2”
011	High priority bank (A) queue “3”
100	Low priority bank (B) queue “0”
101	Low priority bank (B) queue “1”
110	Low priority bank (B) queue “2”
111	Low priority bank (B) queue “3”

Table 6-2. Rate Queue Modes

Software Congestion Level

Bits 6 through 8 set the software congestion level from 000 to 111 in eight steps, which is used when CC MODE is set to “110.” This is shown in Table 6-3.

Software Congestion Level	Segmentation Rate
000	Normal
001	50%
010	25%
011	12.5%
100	6.25%
101	Segmentation Suspended
110	Reserved
111	Reserved

Table 6-3. Software Congestion Level Modes

Time Interval Quota

This 6-bit field is the time interval quota (TIQ) used for average metering, and determines the number of rate queue accesses required for issuing a credit for one cell transfer on that particular VC. Note that a cell transfer occurs only if there is a positive credit for transfer on that VC. Its usage is explained in the average metering function section below.

Cell Quota

This 6-bit field is used to determine the cell quota (CQ) or the maximum number of credits that a VC can accumulate and therefore sets the limit on the maximum number of cells that can be transferred as a burst at peak rate. It is defined as multiples of 32 cells. Its usage is explained in the average metering function section below.

6.1.3 Average Metering Function

The average rate of data transmission on a virtual circuit is a fraction of the peak rate of the rate queue with which that VC is associated. This fraction is determined by two parameters—TIQ and CQ described earlier.

The average metering function is based on the “leaky bucket algorithm” in which a cell transfer credit is issued to the VC periodically. One cell transfer credit is issued to a VC after every (TIQ + 1) accesses to the rate queue to which that VC belongs. Every cell transfer consumes one cell credit. SARA-S segments and transmits a cell on a VC if there is an available credit for cell transfer during every rate queue access. In case there are no cells for segmentation and transmission, the credits will accumulate up to a maximum of 32 times CQ.

The desired peak data rate of a rate queue is determined by the contents of the corresponding rate queue register (RQ_REG_A(0:3), RQ_REG_B (0:3)), as explained in Section section 4.5. The rate queue is accessed once every time the rate queue counter overflows, i.e., at the peak rate. The available credits to segment and transfer a cell are stored in the VC table cell counter and a cell is transmitted only when this internal counter is non-zero. Every cell transmission decrements the cell counter.

As long as the number of cells transferred on a particular VC over any averaging interval is less than maximum credit allowed, cells will be segmented and transferred at the peak rate since credits will be accumulating faster than they are used up. For these purposes, averaging interval is defined as (TIQ+1) * (CQ * 32) * 424 /Peak rate, which is the time required to accumulate the maximum number of credits allowed.

The following ranges are possible with SARA-S:

CQ: Minimum = 0. Maximum = 2048 cells (programmed as multiples of 32 cells)

TIQ: Minimum = 0. Maximum = 63

The effect of various values of TIQ and CQ are described in Table 6-4.

TIQ	CQ	Description
X	0	No cell Quota—Disable VC. Segmentation and cell transfers will not occur for this VC. This can be used by software to temporarily suspend a connection.
0	> 0	One credit issued for every rate queue access—Average rate = Peak rate. This can be used by software to disable the average rate metering effect.
> 0	> 0	Average Metering in force. A positive cell transfer credit will be issued every (TIQ +1) rate queue accesses. Cell transfer rate will be: a) When all credit is used up: Cell transfer rate = Avg rate = Peak rate/ (TIQ+1) b) When positive credit exists: Cell transfer rate=Peak rate.

Table 6-4. Average Metering Rate

The cell transfer algorithm is described below in “C” syntax. TI_CTR and CELL_CTR are internal counters used to store temporary count values:

```
for each_rate_queue_access
{
        if (TI_CTR <= 0)
        {
                TI_CTR = TIQ;
                CELL_CTR = CELL_CTR + 1;
                if (CELL_CTR >= 32*CQ)
                        CELL_CTR = 32 * CQ;
        }
        else TI_CTR = TI_CTR-1
        if (CELL_CTR > 0)
        {
                /*Transfer one cell*/
                CELL_CTR = CELL_CTR - 1;
        }
}
```

Example:

Given a peak_rate = 10 Mbps; Average_Rate = 0.5 Mbps and Max _Burst_Length = 96 cells;

The following values are programmed:

CQ = (Max_Burst_Length/32) = 96/32 = 3.

TIQ = (Peak_Rate / Average_Rate) -1 = 10Mbps / 0.5Mbps - 1 = 19.

Averaging Interval = (TIQ+1) * (CQ*32) *424/Peak_Rate

= 20 * 96 * 424/10Mbps

= 81ms.

6.1.4 Packet Ready Queue

The packet ready queue is used to queue the descriptors of packets for segmentation. The packet ready queue is defined by the four registers in the SARA-S: PRQ_ST_ADR, PRQ_ED_ADR, PRQ_RD_PTR, PRQ_WR_PTR, which hold the starting address, ending address, read pointer and write pointer respectively. The operational states of the packet ready queue are shown in Figure 6-7 and the contents of each valid entry are shown in Figure 6-6. While this queue is non-empty, the SARA-S reads the descriptor number from this queue and links the descriptor into the appropriate rate queue for packet segmentation.

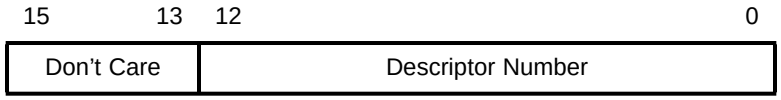


Figure 6-6. Packet Ready Queue Entry

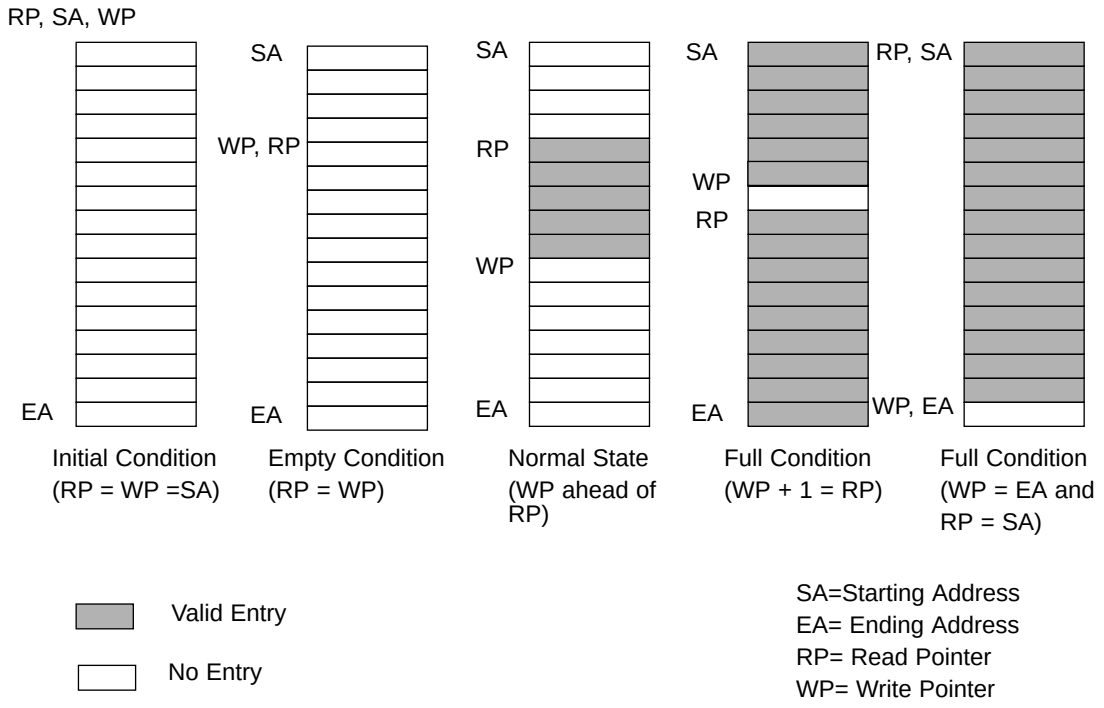


Figure 6-7. Communication Queue Operational States

The starting and ending addresses, when concatenated with the contents of the QUEUE_BASE register, define the location of the queue in the control memory. The address generation is shown in Figure 6-8.

The sequence of operations to load the packet ready queue are:

1. If the packet ready queue is not defined to be large enough to hold all the descriptors, then check that the packet ready queue is not full before writing into it.
2. Read the packet ready queue write pointer from the SARA-S (PRQ_WR_PTR).
3. Store the descriptor number of the buffer descriptor for segmentation into the queue location in the control memory pointed to by the write pointer.
4. Increment the write pointer to point to the next location in the queue, while taking care of the wrap-around condition. Multiple descriptors can be added to the packet ready queue by repeating Steps (3) and (4). After all the descriptors have been written into the packet ready queue, go to Step (5).
5. Store the incremented write pointer to the PRQ_WR_PTR to update the SARA-S and trigger the segmentation process on these descriptors.

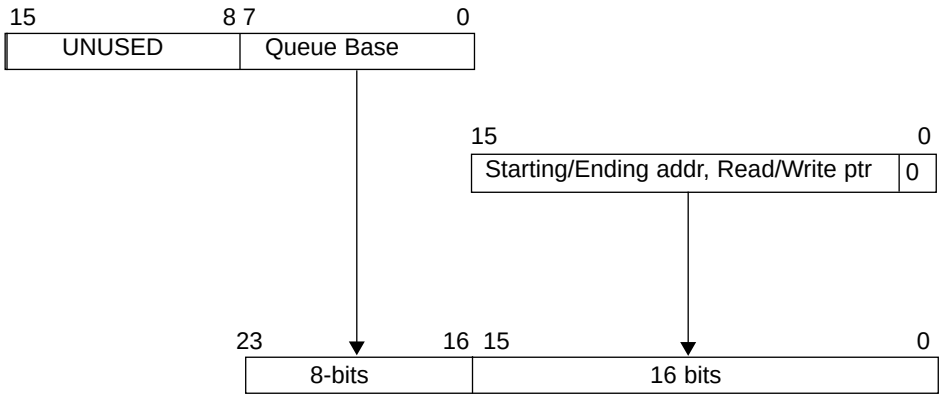


Figure 6-8. Address Generation of Queue Entry

6.1.5 Transmit Complete Queue

The transmit complete queue is used by the SARA-S to return the descriptor numbers of packets that have been segmented. When this queue transitions from an empty to a non-empty state, the SARA-S generates a maskable interrupt to the host. The descriptors are read from this queue to use for subsequent packet segmentation. Like the packet ready queue, this queue is defined by the four registers in the SARA-S: TCQ_ST_ADR, TCQ_ED_ADR,TCQ_RD_PTR, TCQ_WR_PTR which hold the starting address, ending address, read pointer and write pointer respectively. The operational states of the transmit complete queue are shown in Figure 6-7. The queue entry address generation is shown in Figure 6-8. Figure 6-9 shows the format of entries in this queue. Table 6-5 shows the completion codes for the transmit complete queue.

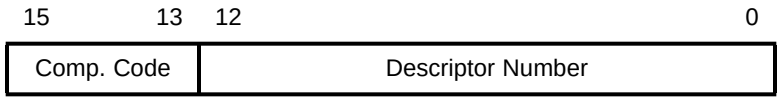


Figure 6-9. Transmit Complete Queue Entry

Completion Code	Description
000	Normal Completion
001	Descriptor flushed as a result of the flush code in the VC table
111	Descriptor de-linked as a result of Packet Memory parity error
Others	Reserved

Table 6-5. Completion Codes for Transmit Complete Queue

The sequence of operations to unload the transmit complete queue are:

1. Check that the transmit complete queue is not empty (TCQ_NOT_EMPTY bit in INTR_STATUS_REG of SARA-S). If the bit is “0,” go to Step (5).
2. Read the transmit complete queue read and write pointers from the SARA-S (TCQ_RD_PTR, TCQ_WR_PTR respectively).
3. If the read pointer is not equal to the write pointer, read and process the descriptor number of the segmented packet from the queue location in the control memory pointed to by the read pointer; otherwise go to Step (5).

4. Increment the read pointer to point to the location of the next entry in the queue (take care of the wrap-around condition). Store the incremented read pointer to the TCQ_RD_PTR register in the SARA-S. Go to Step (2).
5. Exit.

6.2 SARA-S Packet Memory

The packet memory is used to store the packet data for packet segmentation and the packet reassembly. The packet memory can be 32 bits or 16 bits wide. Each of the SARA's can have either separate or can share common packet memory. The packet memory and the control memory can be mapped as regions of the same memory. The SARA-S reads data from, and the SARA-R writes to, the packet memory.

The location of the sources and sinks for the constant bit rate (CBR) traffic is also mapped in the packet memory region.

Three packet data structures are stored in the SARA-S packet memory: packet data for segmentation, OAM (test) cell data, and CBR data. These are described below.

6.2.1 Packet Data Buffers

The location of the packet data for segmentation is stored in the buffer descriptor associated with the packet. The data must be 32-bit word-aligned. The SARA-S reads the data from the packet memory and segments it into cells for transmission. The address of the packet data in the packet memory is generated as shown in Figure 6-10.

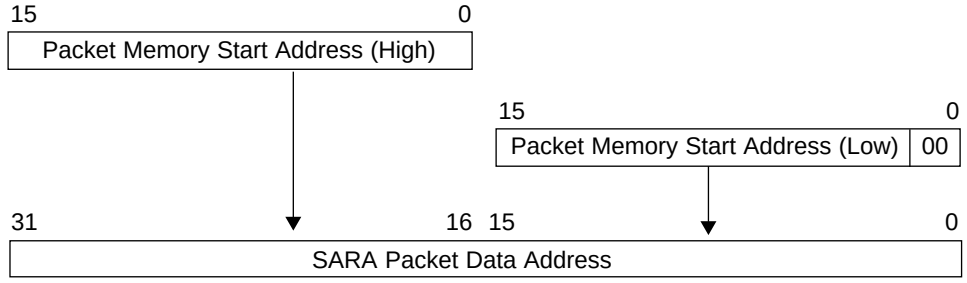


Figure 6-10. Address of Data in Packet Memory

The format expected of the data stored in the packet memory is shown in Table 6-6 and Table 6-7 for big-endian and little endian modes respectively. The SARA-S transmits TX0 first followed by TX1, TX2, TX3, TX4,..., TXk, TXl, TXm, TXn.

Addr	31 16	15 0
00	TX 0	TX 1
04	TX 2	TX 3
08	TX 4	TX 5
.	TX 6	TX 7
.	.	.
	.	.
	TX k	TX l
	TX m	TX n

Addr	15 0	
00	TX 0	TX 1
02	TX 2	TX 3
04	TX 4	TX 5
.	TX 6	TX 7
.	.	.
	.	.
	TX k	TX l
	TX m	TX n

Table 6-6. Packet Memory Data in Big-Endian 32-bit and 16-bit Modes

Addr	31 16	15 0
00	TX 3	TX 2
04	TX 1	TX 0
08	TX 7	TX 6
.	TX 5	TX 4
.	.	.
	.	.
	TX n	TX m
	TX l	TX k

Addr	15 0	
00	TX 1	TX 0
02	TX 3	TX 2
04	TX 5	TX 4
.	TX 7	TX 6
.	.	.
	.	.
	TX n	TX m
	TX l	TX k

Table 6-7. Packet Memory Data in Little-Endian 32-bit and 16-bit Modes

6.2.2 OAM (Test) Data

The OAM (test) cell is a special case of the normal AAL 5 data type. As in the case of AAL5, the SARA-S gets the entire 48-byte payload from the packet memory. The cell header is appended from the VC table in control memory. The PTI bits can optionally be generated from bits in the descriptor, being controlled by the PKT_TYPE in the buffer descriptor. SARA-S always calculates CRC-10 error detection code for all OAM F5 cells.

The format of the data is shown in Table 6-6 and Table 6-7.

6.2.3 CBR Data

The location of the CBR data in the packet memory address space is determined by concatenating the contents of the registers CBR_ADDR_HI and CBR_ADDR_LOW of the SARA-S as shown in Figure 6-11.

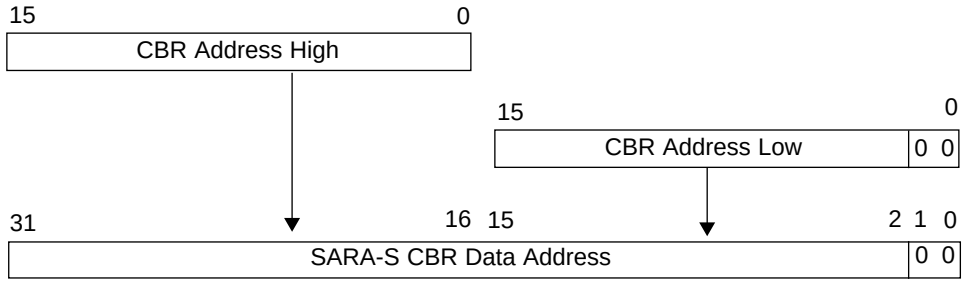


Figure 6-11. Address of SARA-S CBR Data in Packet Memory

The format of the data in the packet memory for the CBR cells is shown in Table 6-8 and Table 6-9. The first 2-bytes are used as a pointer to the VC table from which the ATM header bytes are fetched.

Addr	31 — 16	15 — 0
00	CBR VC Index	Ignore / Ignore
04	Pyld 0	Pyld 1 / Pyld 2 / Pyld 3
08	Pyld 4	Pyld 5 / Pyld 6 / Pyld 7
	.	. / . / .
	.	. / . / .
	.	. / . / .
	.	. / . / .
30H	Pyld 44	Pyld 45 / Pyld 46 / Pyld 47

Addr	15 — 0
00	CBR VC Index
02	Ignore / Ignore
04	Pyld 0 / Pyld 1
32H	Pyld 46 / Pyld 47

Table 6-8. Packet Memory Data in Big-Endian 32-bit and 16-bit Modes for CBR Cells

Addr	31 — 16	15 — 0
00	Ignore / Ignore	CBR VC Index
04	Pyld 3 / Pyld 2	Pyld 1 / Pyld 0
08	Pyld 7 / Pyld 6	Pyld 5 / Pyld 4
	.	. / . / .
	.	. / . / .
	.	. / . / .
30H	Pyld 47 / Pyld 46	Pyld 45 / Pyld 44

Addr	15 — 0
00	CBR VC Index
02	Ignore / Ignore
04	Pyld 1 / Pyld 0
32H	Pyld 47 / Pyld 46

Table 6-9. Packet Memory Data in Little-Endian 32-bit and 16-bit Modes for CBR Cells

6.3 SARA-S Basic Software Functions

Three software processes will be described for SARA-S. These are start-up initialization, virtual-circuit setup, and transmitting a packet.

6.3.1 Startup Initialization

The sequence of steps that should be followed during SARA-S start-up initialization are:

- Allocating buffers in the packet memory
- Allocating and initializing buffer descriptors
- Allocating the VC table
- Setting-up the communication queues
- Setting peak-rate metering parameters
- Initializing the mask bits
- Setting the mode registers

6.3.1.1 Allocating Buffers in the Packet Memory

Buffers in the packet memory are required for storing packets for segmentation. Since a packet must be contained within one buffer, the buffers must be large enough to hold the entire packet. The maximum size of any buffer is 65535 bytes. These buffers can be set up at start-up initialization or created when a packet is setup for segmentation. The important requirement is to estimate the number of buffer descriptors that need to be allocated in the control memory.

6.3.1.2 Allocating and Initializing the Buffer Descriptors

Each buffer in packet memory needs an associated buffer descriptor in the control memory during packet segmentation. Each buffer descriptor has the packet memory start address which is a pointer to the beginning of the buffer in the packet memory as shown in Figure 6-12.

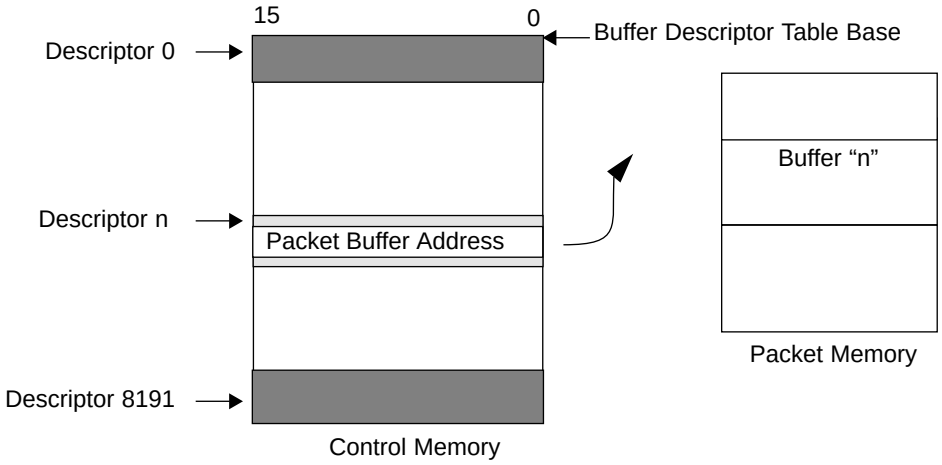


Figure 6-12. Buffers in Packet Memory

The packet buffer start address (both high and low fields in the buffer descriptor) is determined at initialization. If the buffers in the packet memory are allocated at packet setup, loading these fields with the buffer address can be a part of the packet setup. All other bits should be set to zero in each buffer descriptor entry at startup initialization.

The DESC_BASE register in the SARA-S is programmed based upon the location of the buffer descriptor table.

6.3.1.3 Allocating the VC Table

The space requirements for the VC table is determined by the number of virtual circuits that are to be supported (Table 6-10). Each entry in the VC table is accessed by its index. Since the entries in this table are programmed when a virtual circuit is established, nothing needs to be written into this table at startup initialization.

The VC_BASE register in the SARA-S must be programmed based upon the location of the VC table and the number of VCs that will be supported. The significant bits when accessing the VC table are the lower bits of the VC index.

The maximum size of any buffer is 65535 bytes.

Number of VCs	Bits 2-0 of VC_LKUP_BASE Register
65,536	000
32,768	001
16,384	010
8,192	011
4,096	100
2,048	101
1,024	110
512	111

Table 6-10. Setting the Number of VCs for SARA-S

6.3.1.4 Setting-up the Communication Queues

The packet ready queue is used to queue the descriptor numbers for the packets to be segmented. The transmit complete queue is used to return the descriptor numbers after the packets have been segmented. Hence each queue must be large enough to hold all the descriptor numbers active at any time (which may be all of the descriptors).

These queues can be setup in any part of the control memory. The base location of these queues is programmed in the QUEUE_BASE register in the SARA-S. Since these queues are empty upon initialization, the read and write pointer registers of the queues (PRQ_RD_PTR, PRQ_WR_PTR, TCQ_RD_PTR, TCQ_WR_PTR) in the SARA-S are programmed to point to the beginning of the queue (the same as the starting address).

The transmit complete queue can be initialized with all the descriptors and hence used to hold all the free descriptors. A descriptor number can be taken from the transmit complete queue when a packet is setup for segmentation.

6.3.1.5 Setting the Peak-Rate Metering Parameters

SARA-S contains eight rate queues organized into two banks of four. Bank A is of higher priority than bank B. Within each bank the four rate queues 0, 1, 2, and 3 are serviced in a round-robin manner. Each rate queue has a rate counter and an associated rate register, as explained in section 4.5. The rate counter is a free-running counter that is initialized from the corresponding rate registers whenever it overflows. When the rate counter overflows, the corresponding rate queue is serviced, i.e., one cell is transmitted from each of the packets queued on the rate queue.

Each rate queue register must be initialized to realize the peak rate of that queue. The rate counters are clocked with individually pre-scaled versions of the SARA-S clock. The pre-scaling is determined by setting the PRESCALER bits in the rate queue register. Bits 7 through 0 of the rate queue register are used to reload the rate queue counter. The rate queue must be enabled by setting the RQ_ENABLE bit of the rate queue register in order to permit segmentation of virtual circuits associated with that rate queue.

6.3.1.6 Initializing the Mask Bits

After reset, the mask register (MASK_REG) is set to all “1”s, which will disable all interrupts from the SARA-S. The mask register is arranged so that each bit corresponds to the status bit in the same location in the status register (INTR_STATUS_REG). To enable an interrupt from the status register, the corresponding mask register bit must be set to “0”.

6.3.1.7 Setting the Mode Registers

The configuration parameters of the SARA-S mode registers (MODE_REG_0, MODE_REG_1) must be programmed at initialization in accordance with the system design. The SARA-S can then be placed on-line.

6.3.2 Virtual Circuit Setup

When a virtual circuit is established, the corresponding entry in the VC table needs to be set up. The fields that need to be programmed are:

- The ATM header fields.
- Virtual-circuit mode bits and cell quotas that establish the rate queue to be used, the average metering parameters, and congestion control characteristics desired on the virtual circuit.
- All reserved fields must be initialized to zero.

6.3.3 Transmitting a Packet

When a packet needs to be segmented, a free buffer descriptor is needed. This can be obtained from a separate software-maintained list or from the transmit complete queue. The transmit complete queue can be used to hold the available descriptors during normal operation.

Once the descriptor is obtained, the following fields in the buffer descriptor table must be initialized:

- Descriptor mode bits with bits 7 through 0 initialized to ZERO
- VC index
- MID (0 if not used)
- Packet byte count (must be an integral multiple of four bytes)

After the buffer descriptor is initialized, the descriptor number must be written to the packet ready queue using the procedure described earlier in Section 6.1.4.

The SARA-S will subsequently segment the packet and return the descriptor number of the buffer descriptor through the transmit complete queue and generate a maskable interrupt.

The frames to be segmented must be setup using the following guidelines:

1. AAL 3/4 frame without CRC-32 (Figure 6-13)

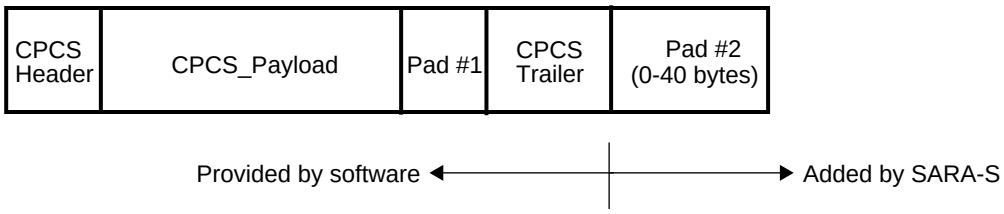


Figure 6-13. AAL 3/4 Frame Without CRC-32

The SARA-S will segment the entire frame, and pad (Pad #2) the unused space following the CPCS trailer with zeros.

2. AAL 3/4, 802.6 frame with CRC-32 (Figure 6-14)

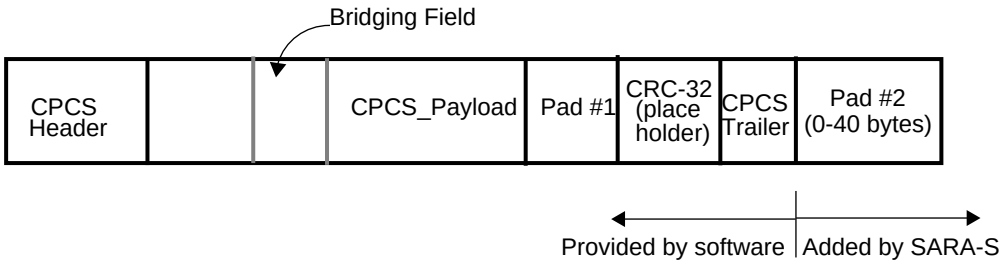


Figure 6-14. AAL 3/4 802.6 Frame With CRC-32

The SARA-S will compute the CRC-32 over the CPCS_Payload and Pad #1. The bridging field (the last two bytes of the 20-byte header in the CPCS_Payload) will be treated as zero for the purpose of the CRC-32 computation. The SARA-S will overwrite the CRC-32 field allocated by the software with the computed value.

3. AAL 3/4, Non-802.6 Frame with CRC-32 (Figure 6-15)

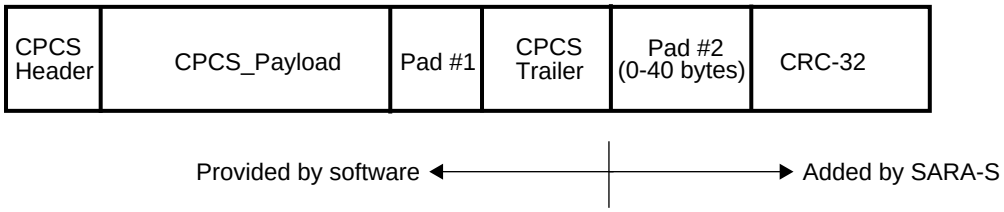


Figure 6-15. AAL 3/4, Non-802.6 Frame with CRC-32

The SARA-S computes the CRC-32 over the CPCS_Header, CPCS_Payload, Pad #1, CPCS_Trailer, and the Pad #2 field (0 to 40 bytes). Pad #2 is added by the SARA-S (all zeros) to fill the last cell and the CRC-32 is placed in the last four bytes.

4. AAL 5 (Figure 6-16)

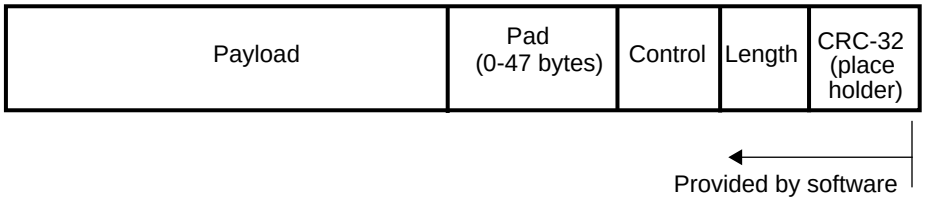


Figure 6-16. AAL 5 Frame with CRC-32

The entire frame is provided by the software and must be an integral multiple of 48 bytes. The SARA-S computes the CRC-32 over the payload, pad, control and length fields. The last four bytes of the last cell are overwritten by SARA-S with the computed CRC-32 value.

6.3.4 Transmitting OAM Cells

When an OAM cell needs to be segmented, a free buffer descriptor is needed. This can be obtained from a separate software-maintained list or from the transmit complete queue. The transmit complete queue can be used to hold the available descriptors during normal operation.

Once the descriptor is obtained, the following fields in the buffer descriptor table must be initialized:

- Set descriptor mode bit APP_CRC 32 = 0

- Set descriptor mode bit PKT_TYPE = 10/11
- Set the PTI value if PKT_TYPE = 11
- VC Index
- Packet byte count=0X30

After the buffer descriptor is initialized, the descriptor number must be written to the packet ready queue using the procedure described earlier in Section 6.1.4.

The SARA-S will subsequently segment the packet and return the descriptor number of the buffer descriptor through the transmit complete queue and generate a maskable interrupt.

6.3.5 Transmitting a CBR Cell

To transmit a CBR cell, the cell should be first loaded into the packet memory location indicated by the CBR_ADDR_HI and CBR_ADDR_LO registers, using the data format described in section 6.2.3.

The CBRXMIT signal is activated to transmit one CBR cell. When the SARA-S has performed this task it will activate the CBRDONE signal whereby the CBRXMIT signal can be deactivated. The cell can also be transmitted by setting the SEND_CBR bit in Mode Register 1 of the SARA-S.

Chapter 7. SARA-R Software Interfaces

This chapter describes the various memory data structures, communication queues and software procedures for using the SARA-R chipset.

7.1 SARA-R Control Memory

The control memory contains data structures which are used by both software and the SARA chipset to maintain dynamic information. It is also used to communicate between the SARA's and the software entity. Each of the SARA devices can have a separate control memory or can share common memory.

The SARA-R control memory is used to hold the buffer descriptor tables, the VC table, the VP table, the reassembly table, the small free descriptor queue, the large free descriptor queue, the packet complete queue and the exception queue. Each of these is described further in the following sections.

7.1.1 Buffer Descriptor Table

The buffer descriptor table is a set of 32-byte entries that contain information about a reassembled packet and the location of the packet buffer in the packet memory. When the SARA-R reassembles a packet, it stores the packet parameters in the descriptor and stores the packet in the packet memory location specified in the descriptor.

The upper six bits of the buffer descriptor table base location in memory is stored in the descriptor base address register (DESC_BASE). Each buffer descriptor table entry is referenced by its descriptor number which can range from 1 to 8191. Descriptor number 0 should not be used, in order to assure compatibility with the SARA-S. The location of the entry is determined by concatenating the descriptor number with the descriptor base address (as shown in Section 5.6). The location of the entire descriptor table in the control memory can be determined by choosing the appropriate range of descriptor numbers that will be used for reassembling packets as shown in the example in Section 6.1.1.

The organization of each descriptor table entry is shown in Figure 7-1, including the type of accesses by the software entity and SARA-R, and the address of the individual entries in the table.

Address	15 — 0	SW	SARA-R
0H	Descriptor Mode/Status Bits	R/W	R/W
2H	Virtual Circuit Index	R	W
4H	Message ID (MID) / Virtual-Path Index (VPI)	R	W
6H	Packet Byte Count	R	R/W
8H	Packet Memory Start Address (High)	W	R
AH	Packet Mem Start Addr. (Low)	W	R
CH	DMA Address - Upper	R	R/W
EH	DMA Address - Lower	R	R/W
10H	Residual CRC - Upper	R	R/W
12H	Residual CRC - Lower	R	R/W
14H	Reserved | Packet Time-out Count	R	R/W
16H	Reserved for SARA use		R/W
18H - 1EH	Unused		

Figure 7-1. Receive Descriptor Table Entry Organization

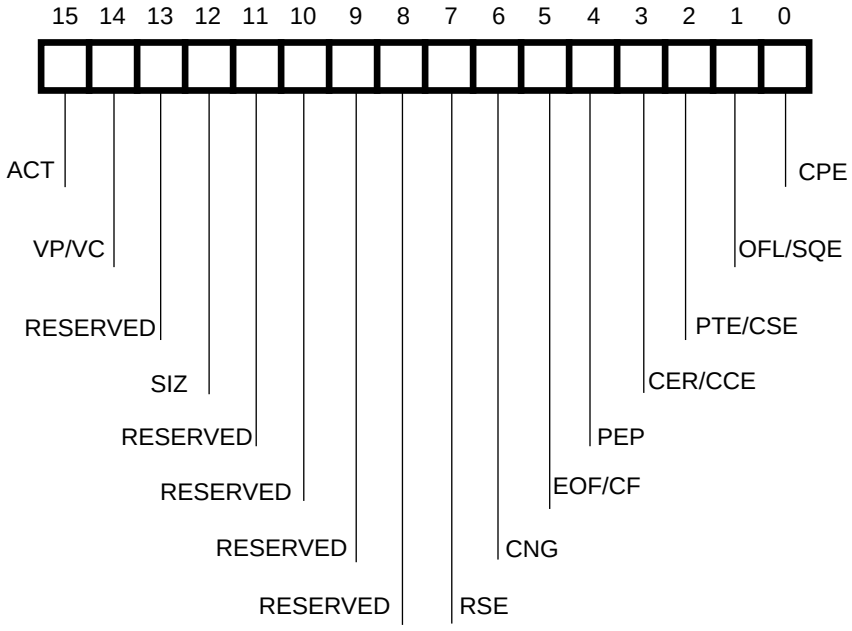


Figure 7-2. Descriptor Mode/Status Bits

The descriptor mode bits, whose positions are shown in Figure 7-2, are used to setup the descriptor entry. The status bits are used by the SARA-R to report the error status of the reassembled packet.

Bit 15: ACT

This bit is set high when the SARA-R is in the process of reassembling the packet associated with this descriptor. When the reassembly process is completed, this bit is set to a “0” by the SARA-R before the descriptor is passed to the software entity.

This bit must not be modified.

Bit 14: VP/VC:

0 - VCI/MID reassembly and MID index is in entry address 4H

1 - VPI/VCI reassembly and VP index is in entry address 4H

This bit indicates whether the packet reassembly was performed on the VPI or VCI.

Bit 13: Reserved

Bit 12: SIZ:

0 - Small buffer size

1 - Large buffer size

This bit indicates the buffer size, set by the host at initialization. It determines the size of buffer in the packet memory i.e. a small buffer if it is set to a “0” and a large buffer it is set to a “1”. The sizes of the buffers are programmed in the SARA-R registers SML_BUF_CHK, LRG_BUF_CHK and LRG_BUF_SIZE.

Bits 11 through 8: Reserved.

Bit 7: RSE

0 - No roll over sequence error

1 - Roll over sequence error detected

This indicates roll over sequence error at packet boundaries; i.e. the first cell of this packet did not have the successive sequence number from the last cell received on this circuit.

Bit 6: CNG

0 - No congestion experienced by this packet in the network.

1 - Congestion was experienced by one or more cells of this packet during transit across the network.

CNG is set when the SARA-R receives a cell with the congestion bit set in the cell header, as an indicator of congestion on the circuit.

Bit 5: EOF/CF

0 - No error

1 - End of packet error/cell flushed

If the CPE bit is “0” then bit-5 is EOF otherwise it is CF.

EOF is set to a “1” when an end-of-packet error occurs. This happens when the last cell of a packet is not received and the packet is terminated.

CF is set to a “1” when a cell that may belong to the active packet was flushed due to errors on the link. The packet is terminated upon this error condition.

Bit 4: PEP

0 - No parity error

1 - Parity error in cell payload

This bit is set to a “1” if a parity error is encountered on the link interface in the payload of the packet.

Bit 3: CER/CCE

0 - No error

1 - Packet CRC error/cell payload CRC error (for AAL 3/4)

If CPE is “0” then bit-3 is CER, otherwise it is CCE.

CER is set to a “1” when a packet CRC error occurs. CER should be ignored if packet CRC is not being used on this VC, or CRC 32 is calculated and appended as per IEEE 802.6 requirement.

CCE is set to a “1” when a cell is received with an erroneous payload CRC (10-bit CRC) belonging to the active packet (for AAL 3/4). The packet is terminated.

Bit 2: PTE/CSE

0 - No error

1 - Packet timeout/cell size error (for AAL 3/4)

If CPE is “0” then bit 2 is PTE, otherwise it is CSE.

PTE is set to a “1” when a packet has not completed within the pre-programmed amount of time (aged out) and hence is terminated.

CSE is set to a “1” when a cell is received with a cell-size that violates the size definition of AAL 3/4. The active packet on the VC on which this cell was received is terminated.

Bit 1: OFL/SQE

0 - No error

1 - Buffer overflow error/cell sequence error (for AAL 3/4)

If CPE is “0” then bit 1 is OFL otherwise it is SQE.

OFL is set to a “1” when a packet overflows the packet buffer in the packet memory and hence is terminated.

SQE is set to a “1” when a cell is received with an out of order AAL 3/4 sequence number (except for sequence errors between EOM and COM/SSM, which are only flagged via RSE). The active packet on the VC that this cell was received on is terminated.

Bit 0: CPE

0 - Bits 5, 3:1 are packet error bits

1 - Bits 5, 3:1 are cell errors

This bit is used to qualify the error conditions on bit 5 and bits 3 through 1. When a cell error is encountered, this bit is set to a “1”, otherwise it is a “0”.

Virtual Circuit Index

This field determines the virtual circuit that packet was received on.

Message ID (MID)/VP Identifier (VPI)

This field indicates the VPI (in bits 7:0) on which the packet was received if the VP/VC bit is “1”, otherwise it contains the MID field in the lower 10 bits for AAL 3/4 packets.

Packet Byte Count

This field is loaded by the SARA-R from the BA_Size field of the BOM/SSM cells for AAL 3/4 packets or from the LRG_BUF_SIZE register for AAL 5 packets. The contents are counted down by the SARA-R during the packet reassembly process. The purpose is to detect the end of packets and buffer overflows.

Packet Memory Start Address

The packet starting location in the packet memory address space, relative to SARA-R, is generated by concatenating the high and low fields of the packet memory start address. The contents of this field must be initialized to the 32-bit word address of the beginning of the packet buffer in the packet memory.

DMA Address

The contents of this field are initialized from the packet memory start address field and contain a pointer to the location immediately following the end of the packet.

Residual CRC

This field is used to maintain the residual 32-bit CRC value of the packet in reassembly (between cells).

Packet Time-out Count

This is an 8-bit field that is initialized by the SARA-R at the start of the packet reassembly and is used to maintain the age of the packet. This field is incremented at periodic intervals while the packet is in the active reassembly process. If this field overflows, the packet is terminated and the PTE bit is set in the descriptor status field.

7.1.2 Reassembly Table Pointer/Descriptor Table Entry Generation

Figure 7.3 shows the process of generating the descriptor table address. Several steps are involved in this process:

1. Determine whether VPI/VCI fields or VCI/MID fields are used for address generation.

2. Generate control memory address for VPI/VCI table entry.

3. Extract reassembly pointer field from the VPI/VCI table entry. Generate reassembly table address using the reassembly pointer field.

4. Extract descriptor number from reassembly table entry (Figure 7-6). Generate descriptor table address using the descriptor number.

Step 1: The reassembly can be done either on VPI/VCI or VCI/MID fields. The VP filter compares the VPI bits with the unmasked bits of the VPI filter register. If they do not match, the reassembly is based on VPI and VCI, and the VP Table entry (Figure 7-5) is used to generate the reassembly pointer. If the results match, the reassembly is based on VCI and MID fields. In this case, the VC Table entry (Figure 7-4) is used to generate the reassembly pointer.

Step 2:

a) If the reassembly is based on VPI/VCI, the 24-bit control memory address used to access VP Table entry is generated by concatenating the 15 MSB of VP Base Address register, 8-bit VPI and address bit 0 set to “0”. Note that the SARA control memory address bus only includes bits 1 - 23; bit 0 is not produced by SARA.

b) If the reassembly is based on VCI/MID, the 24-bit control memory address used to access VC Table ent2ry is generated by concatenating (see Figure 5-8):

   i) *x* most significant bits of VC_LKUP_BASE register, where $x \in [7,13]$; the least significant 3 bits determine the number of bits used.

   ii) *y* LSB of VCI, where $y \in [16,9]$; the least significant 3 bits of the VC_LKUP_BASE register determine the number of bits used. However, when *x*=13 and *y*=9, bit 10 of control memory address is set to “0.”

   iii) Bit 0 of control memory address set to “0.”

   For example, when the 3 least significant bits of VC_LKUP_BASE address are 001, the 24-bit control memory address consists of:

   i) Bits 15:8 of VC_LKUP_BASE register

   ii) Bits 14:0 of VCI

   iii) “0”

Step 3: In this step, the reassembly table address (See Figure 5-11) is generated using the VP/VC table entry.

a) If bit 0 of the VP/VC table entry is “0,” the 24-bit reassembly table address is the concatenation of bits 7:0 of reassembly base register, bits 15:1 of VP/VC table entry, and “0.”

b) If bit 0 of the VP/VC table entry is “1,” bits 1:3 determine the number of bits used from VP/VC table entry. The 24-bit reassembly table address is the concatenation of bits 7:1 of reassembly base register, *x* most significant bits from VP/VC table entry, (16 - *x*) least significant bits from VCI/MID field, and “0.”

Step 4: In this step, the descriptor table address is generated using the reassembly table entry. The descriptor entry address is the concatenation of bits 7:2 of descriptor base address register, bits 12:0 of reassembly table entry (Descriptor Number), 4-bit table offset, and “0” for bit 0.

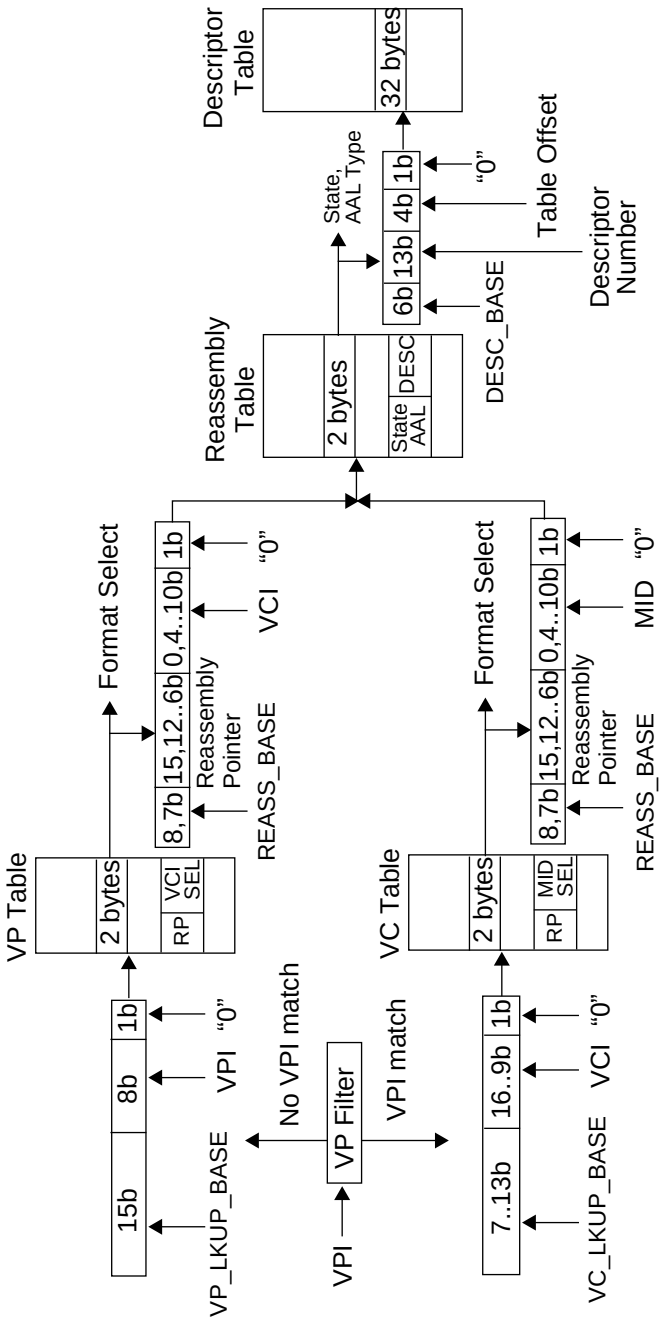


Figure 7-3. Reassembly Table Pointer Process

7.1.3 VC Table

The organization of each VC table entry is shown in Figure 7-4. If bit 0 is "0", the reassembly pointer comprises the 16 bits of the address to the reassembly table location (supporting VCI-only reassembly of up to 32,768 VCs). Refer to Figure 7-4.

If on the other hand, bit 0 is “1”, then the least significant bits of the reassembly pointer are determined by Table 7-1, where the value of bits 3, 2, and 1 (the MID Select Mode in Figure 7-4) determine the number of MID bits that make up the reassembly table pointer, as shown in Figure 7-3. VCI/MID reassembly allows between four and ten MID bits to be used. When the least significant four bits of a VC table entry are set to “1111,” the cells arriving on those virtual circuits are discarded.

The VC table is initialized by the software and read by the SARA-R during the reassembly process. The SARA-R never writes into the VC table

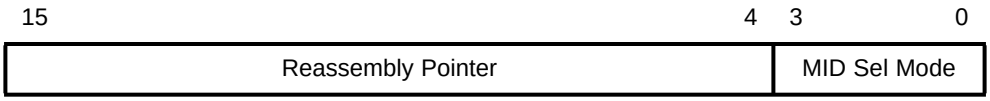


Figure 7-4. VC Table Entry

Bits 3, 2, and 1 of MID Select Mode	Number of MID Bits in Reassembly Pointer
000	10
001	9
010	8
011	7
100	6
101	5
110	4
111	Invalid VC

Table 7-1. Number of MID Bits in Reassembly Pointer

7.1.4 VP Table

The organization of each VP table entry is shown in Figure 7-5. If bit 0 is “0”, the reassembly pointer contains the 16 bit pointer to the reassembly table location (VP only reassembly)

If on the other hand, bit 0 is “1”, then the least significant bits of the reassembly pointer are determined by Table 7-2, where the value of bits 3, 2, and 1 (the VC Select Mode in Figure 7-5) determine the number of VCI bits that make up the reassembly pointer, as shown in Figure 7-3. VPI/VCI reassembly allows between four and ten bits of VCI to be used. When the least significant four bits of a VP table entry are set to “1111,” the cells arriving on those virtual paths are discarded.

The VP table is initialized by the software and read by the SARA-R during the reassembly process. The SARA-R never writes into the VP table.

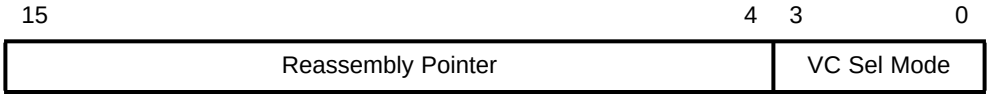


Figure 7-5. VP Table Entry

Bits 3, 2, and 1 of VC Select Mode	Number of VCI bits in reassembly pointer
000	10
001	9
010	8
011	7
100	6
101	5
110	4
111	Invalid VP

Table 7-2. Number of VCI Bits in the Reassembly Pointer

7.1.5 Reassembly Table

The reassembly table is comprised of two-byte entries that contain information relating to each virtual connection. At start-up, the contents of the reassembly table are initialized by the software. The SARA-R maintains the state and the descriptor number of an active packet in the reassembly table.

The organization of each reassembly table entry is shown in Figure 7-6.

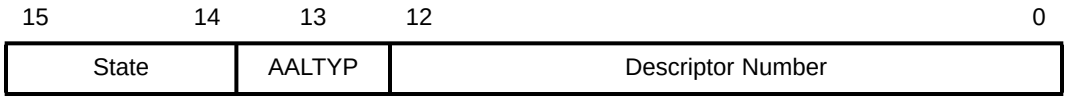


Figure 7-6. Reassembly Table Entry

The state bits indicate the state of the packet reassembly on the virtual circuit, and are described in Table 7-3.

State Bits[1]	AALTYP	Description
00	0	No AAL3/4 packet in reassembly
01	0	AAL3/4 Packet is being reassembled
10	0	AAL3/4 Packet was terminated due to errors
00	1	No AAL 5 packet in reassembly
01	1	AAL 5 Packet is being reassembled
10	1	AAL 5 Packet was terminated due to errors
11	1	Virtual circuit belongs to Raw Cell traffic. All cells are loaded into the Raw Cell queue in packet memory.
11	0	Virtual circuit belongs to CBR traffic. All cells are loaded into the CBR queue in packet memory.

Table 7-3. Reassembly Table Entry State Bits Description

1. When the State Bits are not set to ‘11’, the SARA-R maintains the state of the packet reassembly.

The descriptor number is the index of the buffer descriptor associated with the packet in the reassembly process. The SARA-R does not alter the AALTYP bit (or the state bits when set to “11”).

7.1.6 Small/Large Free Descriptor Queues

The small and large free descriptor queues are used to queue the free descriptors for packet reassembly. When the SARA-R receives a packet, it fetches a descriptor from one of these queues (based on the size of the packet for AAL 3/4; AAL 5 packets always use large buffers). The small free descriptor queue is defined by the four registers in the SARA-R:

- SML_Q_ST_ADR
- SML_Q_ED_ADR
- SML_Q_RD_PTR
- SML_Q_WR_PTR.

The large free descriptor queue is defined by the four registers in the SARA-R:

- LRG_Q_ST_ADR
- LRG_Q_ED_ADR
- LRG_Q_RD_PTR
- LRG_Q_WR_PTR

These registers hold the starting address, ending address, read pointer and write pointer respectively. The operational states of the small and large free descriptor queues are shown in Figure 6-7. The addressing of the queues was shown in Figure 6-8. The format of each entry is shown in Figure 7-7.

The sequence of operations to load the free descriptor queues are similar to the steps in loading the packet ready queue.

1. Check that the small and large free descriptor queues are not full. This step can be avoided if the small and large free descriptor queues are defined to be large enough to hold all the small and large descriptors respectively.
2. Read the free descriptor queue write pointer from the SARA-R (SML_Q_WR_PTR/ LRG_Q_WR_PTR).
3. Store the descriptor number of the free buffer descriptor for packet reassembly into the queue location in the control memory pointed to by the write pointer.
4. Increment the write pointer to point to the to the next location in the queue (take care of the wrap-around condition). Multiple descriptors can be added to the free descriptor queue by repeating Steps (3) and (4). After all the descriptors have been written into the free descriptor queue go to Step (5).
5. Store the incremented write pointer to the SML_Q_WR_PTR/LRG_Q_WR_PTR to update the SARA-R.

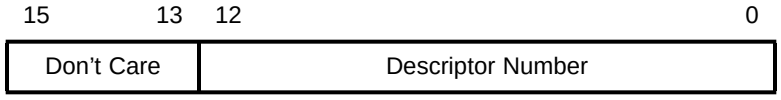


Figure 7-7. Small/Large Free Descriptor and Packet Complete Queue Entry

7.1.7 Packet Complete Queue

The packet complete queue is used by the SARA-R to queue the descriptors of reassembled packets. When this queue becomes non-empty, the SARA-R generates a maskable interrupt to the host. The descriptors are read from this queue, and the received packet is processed by software. The starting address, ending address, read pointer and write pointer are stored in the SARA-R registers PCQ_ST_ADR, PCQ_ED_ADR, PCQ_RD_PTR, PCQ_WR_PTR respectively. The operational states of the packet complete queue are shown in Figure 6-7. The queue locations are addressed as was shown in Figure 6-8. The contents of entries of the packet complete queue are shown in Figure 7-7.

The sequence of operations to unload the packet complete queue are similar to the steps in unloading the transmit complete queue.

1. Check that the packet complete queue is not empty (PKT_RCVD bit in INTR_STATUS_REG of SARA-R. If the bit is "0," go to Step (5).
2. Read the packet complete queue read and write pointers from the SARA-R (PCQ_RD_PTR, PCQ_WR_PTR respectively).
3. If read pointer is not equal to the write pointer, read and process the descriptor number of the reassembled packet from the queue location in the control memory pointed by the read pointer; otherwise go to Step (5).
4. Increment the read pointer to point to the location of the next entry in the queue (take care of the wrap-around condition). Store the incremented read pointer to the PCQ_RD_PTR register in the SARA-R. Go to Step (2).
5. Exit.

Read the packet complete queue write pointer from the SARA-R (PCQ_WR_PTR). If the write pointer is not equal to the read pointer go back to Step (3).

7.1.8 Exception Queue

The exception queue is used by the SARA-R to transfer the exception errors to the software entity. The exception queue is used to transfer error conditions that cannot be associated with a packet; error conditions that can be associated with a descriptor of a packet are transferred through the descriptor status bits.

When this queue becomes non-empty, the SARA-R generates a maskable interrupt to the host. The software reads the descriptors from this queue, and processes the received exception. The contents of the SARA-R registers EXCP_Q_ST_ADR, EXCP_Q_ED_ADR, EXCP_Q_RD_PTR, EXCP_Q_WR_PTR are the starting address, ending address, read pointer and write pointer of the exception queue respectively. The operational states of the exception queue are shown in Figure 6-7. The address generation was shown in Figure 6-8.

The sequence of operations to unload the exception queue are similar to those of the packet complete queue, except that each entry in the exception queue is four bytes long.

1. Check that the exception queue is not empty (EXCP_RCVD bit in INTR_STATUS_REG of SARA-R). If the bit is “0,” go to Step (5).
2. Read the exception queue read and write pointers from the SARA-R (EXCP_Q_RD_PTR, EXCP_Q_WR_PTR respectively).
3. If read pointer is not equal to the write pointer, read and process the received exception from the queue location in the control memory pointed by the read pointer; otherwise go to Step (5).
4. Increment the read pointer to point to the location of the next entry in the queue (take care of the wrap-around condition). Store the incremented read pointer to the EXCP_Q_RD_PTR register in the SARA-R. Go to Step (2).
5. Exit.

The contents of the entries in the exception queue are shown in Figure 7-8. Table 7-4 shows the error code explanations.

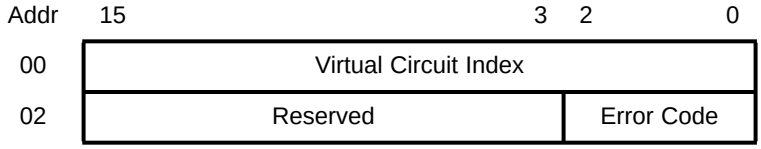


Figure 7-8. Exception Queue Entry

Error Code	Description
000	No Error (should never occur)
001	Out of Sequence COM cell received
010	Out of Sequence EOM cell received
011	Reserved
100	No buffers available (small packet dropped)
101	No large buffers available (large packet dropped)
110	Invalid VCI (cell received on invalid VC)
111	Invalid VPI (cell received on invalid VP)

Table 7-4. Error Code Explanations for Exception Queue

7.2 SARA-R Packet Memory

The packet memory is used to store the packet data for packet segmentation and the packet reassembly. The packet memory can be 32- or 16-bits wide. Each of the SARA's can have either separate or can share common packet memory. The packet memory and the control memory can be mapped as regions of the same memory. The SARA-S reads data from, and the SARA-R writes to, the packet memory.

The location of the sources and sinks for the constant bit rate (CBR) traffic is also mapped in the packet memory region.

Three packet data structures are stored in the SARA-R packet memory: packet data for reassembly, raw cells and CBR data. These are described below.

7.2.1 Packet Data for Reassembly

The location of the reassembled packet data is stored in the buffer descriptor associated with the packet. The SARA-R gets the data from the cell payload and writes it into the packet memory, 32-bit word-aligned. The address of the packet data in the packet memory is located in the buffer descriptor.

The format of the data stored in the packet memory is shown in Table 7-5 for Big Endian and Table 7-6 for Little Endian-mode. The order of the received data bytes is RX0 followed by RX1, RX2, RX3, RX4,..., RXk, RXl, RXm, RXn.

Addr	31	16	15	0
00	RX 0	RX 1	RX 2	RX 3
04	RX 4	RX 5	RX 6	RX 7
08	.	.	.	.
.	.	.	.	.
.	RX k	RX l	RX m	RX n

Addr	15	0
00	RX 0	RX 1
02	RX 2	RX 3
04	.	.
.	RX k	RX l
.	RX m	RX n

Table 7-5. Packet Memory Data in Big Endian 32-bit and 16-bit Modes

Addr	31	16	15	0
00	RX 3	RX 2	RX 1	RX 0
04	RX 7	RX 6	RX 5	RX 4
08	.	.	.	.
.	.	.	.	.
.	RX n	RX m	RX l	RX k

Addr	15	0
00	RX 1	RX 0
02	RX 3	RX 2
04	.	.
.	RX l	RX k
.	RX n	RX m

Table 7-6. Packet Memory Data in Little Endian 32-bit and 16-bit Modes

7.2.2 CBR Data

Any virtual circuit can be set up for CBR traffic. Cells received on these VCs are treated as CBR cells. The VC setup is described in Section 7.3.1.8.

CBR cells are stored in a circular cell queue in the packet memory address space. The cell queue is defined by a starting address, ending address, read pointer and write pointer. These are stored in the SARA-R registers CBR_FIF_ST_ADR, CBR_FIF_ED_ADR, CBR_FIF_RD_PTR, CBR_FIF_WR_PTR respectively. When a CBR cell is received, the cell is written into the entry pointed to by the write pointer and the pointer is incremented to point to the next entry location. Each entry is 64 bytes; the CBR cell occupies the first 52 bytes of the entry.

The location of the CBR data in the packet memory address space is determined by concatenating the contents of the appropriate read or write pointers with the CBR_BASE_ADR register as shown in Figure 7-9. The raw cell/congestion control cell queue shares the same base address as the CBR queue.

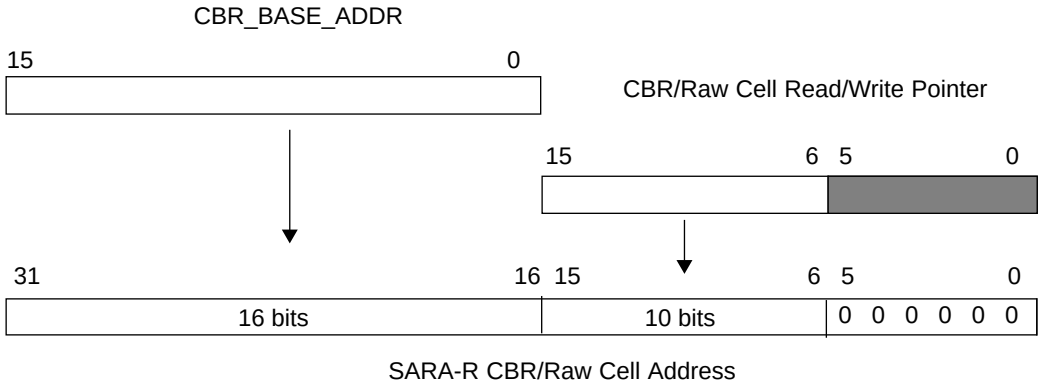


Figure 7-9. Address of SARA-R CBR/Raw Cell Data in Packet Memory

The format of the data in the packet memory for the cells is shown in Table 7-7 for Big Endian Mode and in Table 7-8 for Little Endian Mode. The first four bytes are the first four bytes of the cell header (the header CRC byte is not present) and the remaining 48 bytes are the cell payload.

Addr	31 16	15 0
00	Header 0 Header 1	Header 2 Header 3
04	Pyld 0 Pyld 1	Pyld 2 Pyld 3
08	Pyld 4 Pyld 5	Pyld 6 Pyld 7
	. .	. .
	. .	. .
30H	Pyld 44 Pyld 45	Pyld 46 Pyld 47

Addr	15 0	
00	Header 0	Header 1
02	Header 2	Header 3
04	Pyld 0	Pyld 1
	.	.
	.	.
32H	Pyld 46	Pyld 47

Table 7-7. Packet Memory 32-bit and 16-bit Data Format of CBR /Raw Cells—Big Endian Mode

Addr	31 16	15 0
00	Header 3 Header 2	Header 1 Header 0
04	Pyld 3 Pyld 2	Pyld 1 Pyld 0
08	Pyld 7 Pyld 6	Pyld 5 Pyld 4
	. .	. .
	. .	. .
30H	Pyld 47 Pyld 46	Pyld 45 Pyld 44

Addr	15 0	
00	Header 1	Header 0
02	Header 3	Header 2
04	Pyld 1	Pyld 0
	.	.
	.	.
32H	Pyld 47	Pyld 46

Table 7-8. Packet Memory 32-bit and 16-bit Data Format of CBR /Raw Cells—Little Endian Mode

7.2.3 Raw Cell Queue

This queue is used to store OAM F5, congestion notification, and raw cells (including OAM F4).

OAM F5 cells have the payload type field in the cell header of 100 or 101. They may be used for operation and more general functions.

Congestion notification (also called “congestion control”) cells are those identified either by the most significant bit of the first byte in the cell header (MSB of the GFC field) being set to “1” or by special OAM F5 cells, as described in Section 5.28.

Raw cells are those received on circuits whose reassembly table entry is set to direct the cells to the raw cell queue. This includes OAM F4 cells. All raw cells will be checked for CRC-10. If there is CRC-10 error, the LSB of GFC will be set to “1”.

The raw cell queue operates identically to the CBR queue. The raw cells, OAM F5 and congestion notification cells are stored in a circular cell queue in the packet memory address space. The queue is defined by a starting address, ending address, read pointer and write pointer. These parameters are stored in the SARA-R registers CC_FIFO_ST_ADR, CC_FIFO_ED_ADR, CC_FIFO_RD_PTR, CC_FIFO_WR_PTR respectively. When a cell is received, the cell is written into the entry pointed to by the write pointer and the pointer is incremented to point to the next entry location. Each entry is 64 bytes. The raw cell occupies the first 52 bytes of the entry.

The location of the cell data in the packet memory address space is determined by concatenating the contents of the appropriate read or write pointers with the CBR_BASE_ADR register as shown in Figure 7-9. The raw cell queue shares the same base address as the CBR queue.

The format of the data in the packet memory for the raw cells is shown in Table 7-7 and Table 7-8. The first four bytes are the first four bytes of the cell header (header CRC is not present) and the remaining 48 bytes are the cell payload.

7.3 SARA-R Basic Software Functions

This section describes start-up initialization, virtual-circuit setup and packet reception for SARA-R.

7.3.1 Startup Initialization

The sequence of steps that should be followed during SARA-R initialization are:

- Allocating buffers in the packet memory
- Allocating and initializing buffer descriptors
- Allocating the VC, VP and reassembly tables
- Setting-up the communication queues
- Initializing the mask bits
- Setting up the VC, VP and reassembly tables
- Setting the mode registers

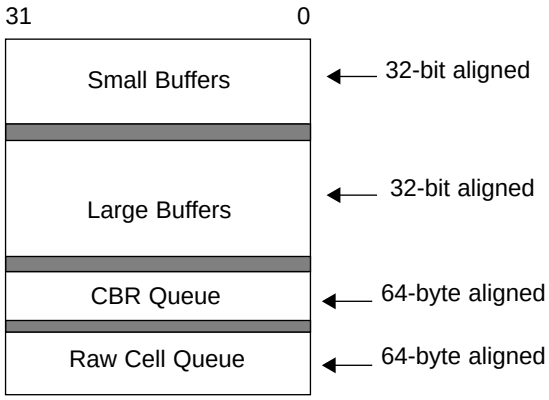


Figure 7-10. Packet Memory Data Structure Alignments

7.3.1.1 Allocating Buffers in the Packet Memory

Buffers in the packet memory are required for reassembling packets. These buffers have to be setup at start-up initialization. The SARA-R supports two programmable sizes of buffers—small and large. The split between small and large buffers is dependent upon the traffic pattern that is expected and the available packet memory. There is no required ratio of the small to large buffers, but the maximum number of descriptors (small and large) is 8,191. The maximum size of any buffer is 65,535 bytes.

The LRG_BUF_SIZE register should be programmed with a value equal to the actual size of the buffer; the LRG_BUF_CHK and SML_BUF_CHK registers should be programmed with values that are 52 less than the actual size of the large and small buffers. This is required because the SARA-R stores the entire last cell of the packet (even if its size is zero) and the BA_SIZE field in the CS_PDU header does not include the length of the CS_PDU header and trailer.

The CBR queue is also mapped into the packet memory region. The size of the CBR queue is dependent upon the system requirements. If a real FIFO is used for the CBR traffic, all the CBR queue pointers in the SARA-R should be programmed to point to the packet memory

address of the real FIFO (queue size of 0). In addition the IGN_CBR_FL mode bit in the Mode Register 0 of the SARA-R (MODE_REG_0) should be set to “1”. This will ignore the FIFO full condition check during CBR reception.

The raw cell queue is located in the packet memory. This is used when the software entity wants to monitor the congestion notification cells, OAM F5 cells, and raw cells being received on the link. Some memory should be allocated for this queue (optional) and the queue pointers must be initialized. Here too, the IGN_RAW_FL mode bit in the Mode Register 0 of the SARA-R (MODE_REG_0) could be set to “1”. This will allow the software entity to look at the most recent cells received. Figure 7-10 shows the packet memory data structure alignments.

7.3.1.2 Allocating and Initializing the Buffer Descriptors

Once the number of small and large buffers in packet memory has been determined, the buffer descriptor table can be allocated. Each buffer in packet memory needs an associated buffer descriptor in the control memory.

The buffer descriptor fields that are programmed at initialization are the descriptor mode bits and the packet memory start address.

- The ‘SIZ’ bit in the descriptor mode bits is set to a “0” for descriptors of small buffers and to a “1” for large buffers. All other descriptor mode bits must be set to zero.
- The packet memory start address (high and low) fields are programmed to point to the buffer in packet memory as shown earlier.

The DESC_BASE register in the SARA-R is programmed based upon the location of the buffer descriptor table. The buffer descriptor table must start on a 32-byte boundary.

7.3.1.3 Allocating the VP Table

The VP table entries must be allocated. There are always 256 entries, and the table must be aligned on a 512 byte boundary. The VP_LKUP_BASE register must be programmed based upon the location of the VP table. The entries in the table must point to the appropriate location in the reassembly table.

7.3.1.4 Allocating the VC Table

The number of virtual circuits that are to be supported determines the allocation of the VC table. Each entry in the VC table is indexed by the VC index. The VC index is the cell VCI header field when reassembling on VCI-only. All entries in the VC table must be initialized to the appropriate location in the reassembly table.

The VC_LKUP_BASE register in the SARA-R must be programmed based upon the location of the VC table and the number of VCs that will be supported. Supporting less than 65,536 VCs causes the appropriate most significant bits of the VC index to be masked to zero before accessing the VC table. The number of VCs supported is shown in Table 7-9.

Number of VCs	Bits 2 Through 0 of VC_LKUP_BASE Register
65,536	000
32,768	001
16,384	010
8,192	011
4,096	100
2,048	101
1,024	110
512	111

Table 7-9. Setting the Number of VCs for SARA-R

The VC table must be aligned on the boundary corresponding to the table size. For example, if the VC table is configured for 1024 VCs (2048-byte table size), it must be aligned on a 2048-byte boundary.

7.3.1.5 Allocating the Reassembly Table

The reassembly table size must always be a multiple of 256 entries and must be aligned on a 512 byte boundary. The dimensions depend on the total number of circuits to be supported.

7.3.1.6 Setting-up the VC Table

To setup the VC table, pointers must be set up either to indicate an invalid circuit, VCI-only reassembly, or VCI/MID reassembly. For VCI-only, the VC table entry forms the lower 16 bits of the reassembly table pointer. For reassembly using MID, the reassembly base register, reassembly pointer, and MID value are concatenated to generate the reassembly table pointer. For more details on the number of MID and reassembly pointer bits used to generate the reassembly table pointer, see Section 5.10.

7.3.1.7 Setting-up the VP Table

To setup the VP table, pointers must be setup either to indicate an invalid circuit, VPI-only reassembly or VPI/VCI reassembly. For VPI-only, the VP table entry forms the lower 16 bits of the reassembly table pointer. For reassembly using VCI, the reassembly base register, reassembly pointer, and VCI value are concatenated to generate the reassembly table pointer. For more details on the number of VCI and reassembly pointer bits used to generate the reassembly table pointer, see Section 5.10.

7.3.1.8 Setting-up the Reassembly Table

The reassembly table must be initialized such that all AAL3/4 entries are initialized to “0”, all AAL5 entries are initialized to 2000H, all CBR circuits are initialized to C000H, and all raw cell circuits are initialized to E000H.

7.3.1.9 Setting-up the Communication Queues

The small and large free descriptor queues are used by the SARA-R to obtain descriptors to reassemble packets. The packet complete queue is used to return the descriptors after the packets have been reassembled. The small and large free descriptor queues should be large enough to hold all the small and large descriptor numbers respectively. The packet complete queue must be large enough to hold all the descriptor numbers. These queues must be 16-bit aligned.

The size of the exception queue is determined based on statistical analysis of the rate of exception conditions and should be set accordingly. This queue must be 32-bit aligned.

These queues can be setup in any part of the control memory. The base location of these queues is programmed in the QUEUE_BASE register in the SARA-R. The starting and ending addresses of the queues are programmed in the corresponding registers in the SARA-R. The small free descriptor queue is initialized with the descriptor numbers of small buffers and the large free descriptor queue is initialized with the descriptor numbers of large buffers using the procedure described in Section 7.1.6. After initialization, the read pointers will point to the start of the queue and the write pointers will point to the first empty location of the corresponding queue.

Since the packet complete and exception queues are empty upon initialization, the read and write pointer registers of these queues in the SARA-R are programmed to point to the beginning of the corresponding queue. A sample control memory map is shown in Figure 7-11.

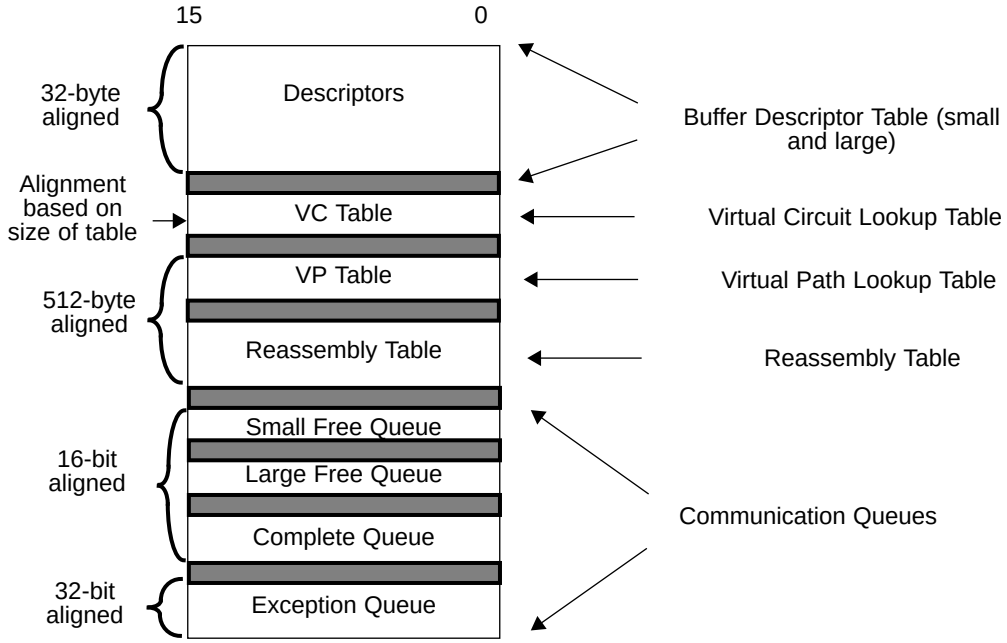


Figure 7-11. Control Memory Data Structure Alignments

7.3.1.10 Initializing the Mask Bits

After reset, the mask register (MASK_REG) gets set to all “1”s, which disables all interrupts from the SARA-R. The mask register is arranged so that each bit corresponds to the status bit in the same location in the status register (INTR_STATUS_REG). To enable an interrupt from the status register, the corresponding mask register bit must be set to “0”.

7.3.1.11 Setting the Mode Registers

The configuration parameters of the SARA-R mode registers (MODE_REG_0, MODE_REG_1) must be programmed at initialization in accordance with the system design. The SARA-R can then be placed on-line.

7.3.2 Receiving a Cell

When a cell is received by the SARA-R, it is processed in the following steps:

1. If the cell is an idle cell, it is dropped. Dropped idle cells are not counted.
2. If the cell is a congestion notification cell, the congestion control information is

transferred to the SARA-S, and the cell is optionally loaded into the raw cell queue, based on the setting of the CC_XFER_EN bit in Mode Register 1 of the SARA-R.

3. If the cell is an OAM F5 cell, it is loaded into the raw cell queue.
4. If the cell is none of the above, the VPI field is passed through a filter, and the appropriate entry in the VP lookup table or VC lookup table is read.
    - If the entry indicates an invalid VP or VC, the cell is discarded and an exception error is reported.
    - Otherwise the appropriate entry in the reassembly table is read.
        - If the entry indicates a CBR connection, the cell is fed to the CBR queue.
        - If the entry indicates a raw cell connection, the cell is located in the raw cell queue.
        - Otherwise the cell is processed as either an AAL 3/4 or AAL 5 packet, as indicated by the entry.

7.3.3 Receiving a Packet

When the SARA-R receives the beginning of a packet, it fetches a descriptor from the large descriptor queue for an AAL 5 packet or from the small or large free descriptor queue (based upon the comparison of the SML_BUF_CHK register and the BA_Size field of the received packet) for an AAL 3/4 packet and reassembles the packet in the packet memory buffer associated with the buffer descriptor. When the packet is completely reassembled or the packet is terminated due to errors, the buffer descriptor number is loaded into the packet complete queue. When the packet complete queue goes from an empty state to a non-empty state, the SARA-R generates the packet-received maskable interrupt.

Upon receipt of a packet-received interrupt, the descriptor number of the reassembled packet is read from the packet complete queue as described earlier. The descriptor number points to the buffer descriptor. The buffer descriptor will contain the status, the 16-bit VC index, and either the 8-bit VPI or 10-bit MID of the reassembled packet. The packet memory start address points to the packet location in the packet memory. The DMA address points to the location immediately following the end of the packet.

If the lower six bits of the descriptor mode/status bits are non-zero, then an error was encountered during packet reassembly. This packet may be dropped if erroneous packets are not acceptable. Error codes may be used to detect the cause of the error. If no error is encountered, the packet can be processed and made available to higher layer protocol processing. Once the packet is processed, the descriptor status bits should be set to “0”, and the descriptor written into the small or large free descriptor queue based upon the ‘SIZ’ bit in the descriptor mode field.

Table 7-10 explains the receive descriptor error status bits, bits (5:0) of the first buffer descriptor word. The specified error conditions are independent of the bits marked “X”. Mutually independent error conditions can be valid at the same time for the same descriptor.

Error Bits	Error Condition
000000	No Error
X1XXXX	Parity error in the cell payload at the Cell Interface
0X0011	Out of Sequence Cell received
0X0101	Invalid length of Cell received

Table 7-10. Receive Descriptor Error Status Bits

Error Bits	Error Condition
0X1XX1	Erroneous payload CRC of Cell Received
1X0001	Packet terminated due to short cell
0X10X0	Error detected in the 32-bit Packet CRC (Ignore if packet CRC is not used)
1X0000	Packet terminated due to the reception of a new packet
0XX010	Packet terminated due to buffer overflow when the last cell was received
1X0010	Packet terminated due to buffer overflow before the last cell was received
XX0100	Packet terminated due to the packet aging process in the SARA-R

Table 7-10. Receive Descriptor Error Status Bits

7.3.4 Receiving Constant Bit Rate Traffic

When the SARA-R receives a cell on VCs that are marked as carrying CBR traffic (7.3.2), the SARA-R stores the 4-byte cell-header (the HEC byte is dropped) and the 48-byte payload in the CBR queue. The write pointer of the CBR queue is incremented. If the queue changes state from empty to non-empty, the CBR_RCVD bit is set in the SARA-R status register.

When the CBR cell is then read from this CBR queue, the CBR queue read pointer is incremented by software.

If the CBR queue is full, the CBR cell is dropped and the CBR cell dropped counter is incremented.

The destination of the CBR data can be a real FIFO. The FIFO is mapped in the packet memory address space and the CBR queue pointers in the SARA-R should point to the address of the FIFO. The IGN_CBR_FL bit must be set in the SARA-R mode register 0. The IGN_CBR_FL bit will cause the CBR queue full-condition (generated by the CBR queue pointers) to be ignored by the SARA-R.

7.3.5 Receiving Congestion Notification Cells

When the SARA-R receives a congestion notification cell, it stores the 4-byte cell-header (the HEC byte is dropped) and the 48-byte payload of the congestion control cell in the raw queue. The write pointer of the raw queue is incremented. If the queue changes state from empty to non-empty, the RAW_RCVD bit is set in the SARA-R status register.

If the raw queue is full or the SARA-S is not ready to receive the congestion control information over the congestion control interface, the congestion notification cell is dropped.

The transfer of the congestion control information to the SARA-S will not be affected when the storing of the cells in the raw cell queue is disabled by setting the CC_XFER_EN bit to ‘0’ in the SARA-R mode register 1.

When the congestion notification cell is read from the raw cell queue, the raw cell queue read pointer is incremented by software.

If the IGN_RAW_FL bit is set in the SARA-R mode register 0, the SARA-R will ignore the raw cell queue full-condition (caused by the raw cell queue pointers in the SARA-R) and overwrite the previous data.

7.3.6 Exception Queue Handling

The exception queue is needed to report errors that cannot be reported through the descriptor status bits. When the SARA-R receives a COM (Continuation-of Message) or a EOM (End-of-Message) cell and there is no packet reassembly in progress on the corresponding VC, it reports the reception of an Out-of-Sequence cell error message through the exception queue. Also, if it receives a BOM (Beginning-of-Message) or a SSM (Single-Segment-Message) and there are no buffers available for reassembly, it will report this error condition through the exception queue. When an exception error is reported, the SARA-R will not report the same error if it receives further COM or EOM cells on the same VC.

When an exception error occurs, the contents of the exception queue are read by software and the errors logged if needed.

Appendix A. Control Memory Size Examples

Table A-1 shows the SARA-R control data structures. The corresponding SARA-S control memory data structures are shown in Table A-2.

Data Structure	Size per Entry	Number of Entries	
		Min.	Max.
Buffer Descriptor	32 bytes	1	8,191
Virtual Circuit Table	2 bytes	512	65,536
VP Table	2 bytes	256	256
Reassembly Table	2 bytes	256	65,536
Packet Complete Queue	2 bytes	1	8,192
Large Free Descriptor Queue	2 bytes	1	8,192
Small Free Descriptor Queues	2 bytes	1	8,192
Exception Queue	4 bytes	1	8,192

Table A-1. SARA-R Control Memory Data Structures

Data Structure	Size per Entry	Number of Entries	
		Min.	Max.
Buffer Descriptor	32 bytes	1	8,191
Virtual Circuit Table	16 bytes	512	65,536
Transmit Complete Queue	2 bytes	1	8,192
Packet Ready Queue	2 bytes	1	8,192

Table A-2. SARA-S Control Memory Data Structures

Table A-3 shows SARA-R control memory size examples for both a small and a large system. The corresponding SARA-S control memory size examples are shown in Table A-4.

Data Structure	Small System		Large System	
	Number	Size	Number	Size
Buffer Descriptor	256	8 kB	1,024	32 kB
Virtual Circuit Table	512	1 kB	2,048	4 kB
VP Table	256	512 B	256	512 B
Reassembly Table	512	1 kB	2,048	4 kB
Packet Complete Queue	256	512 B	1,024	2 kB
Large Free Descriptor Queue	128	256 B	512	1 kB
Small Free Descriptor Queue	128	256 B	512	1 kB
Exception Queue	128	512 B	512	2 kB
TOTAL		12 kB		46.5 kB

Table A-3. SARA-R Control Memory Size Example

Data Structure	Small System		Large System	
	Number	Size	Number	Size
Buffer Descriptor	224	7 kB	896	28 kB
Virtual Circuit Table	512	8 kB	2,048	32 kB
Transmit Complete Queue	256	512 B	1,024	2 kB
Packet Ready Queue	256	512 B	1,024	2 kB
TOTAL		16 kB		64 kB

Table A-4. SARA-S Control Memory Size Example

In the above tables, two different system size examples are shown. For the small system, two 8K x 8 SRAM chips could be used for each SARA; for the large system, two 32K x 8 SRAM chips can be used for each SARA.

Control Memory Maps for a Small System

The control memory maps for the SARA-R and SARA-S could each look like the following:

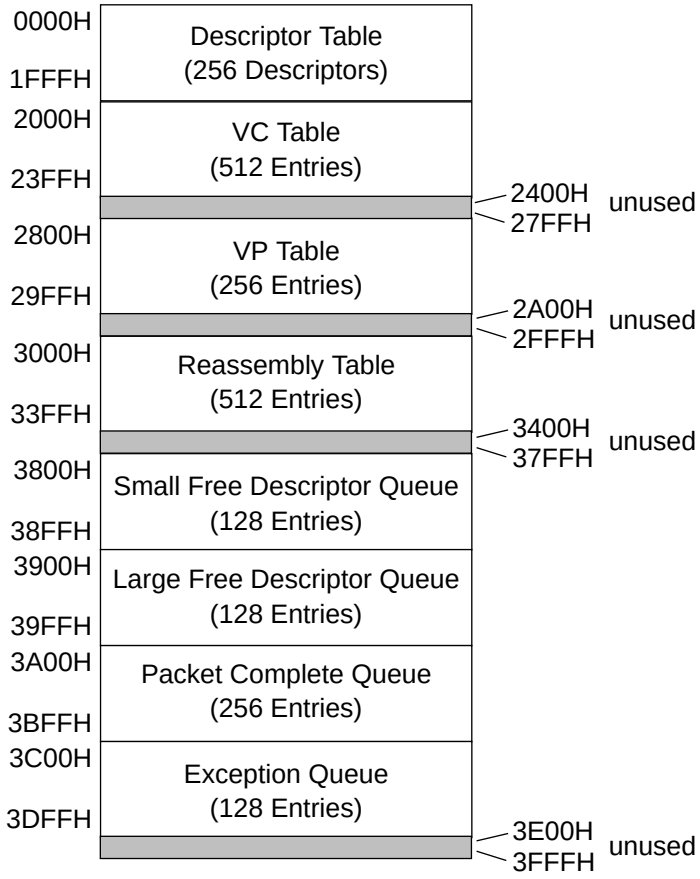


Figure A-1. SARA-R Control Memory Map for Small System

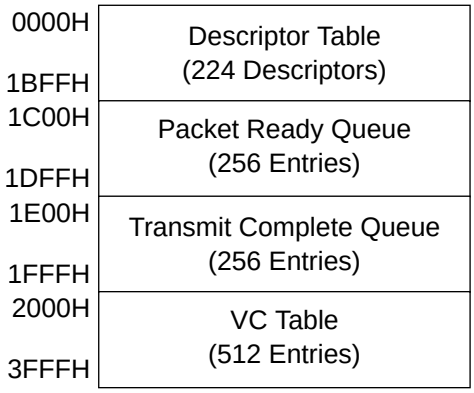


Figure A-2. SARA-S Control Memory Map for Small System

Calculation of SARA-R Register Values for a Small System

SARA-R Register Set-up:

Descriptor_Table_Base = 0x000000;

VC_Table_Base = 0x002000;

Num_Valid_VC_Bits = 9;

VP_Table_Base = 0x002800;

Reass_Table_Base = 0x003000

Reass_Table_End = 0x0033FF;

Queue_Base = 0x003800;

Legend:
0x = hexadecimal
>>N = right shift by N bits
<<N = left shift by N bits
| = logical OR
& = logical AND

R_DESC_BASE = (Descriptor_Table_Base >> 16) & 0xFFFC = 0x0000;

R_VC_LKUP_BASE = ((VC_Table_Base >> 8) & 0xFFF8) |
(16 - Num_Valid_VC_Bits) = 0x0027;

R_VP_LKUP_BASE = (VP_Table_Base >> 8) & 0xFFFE = 0x0028;

R_REASS_BASE = (Reass_Table_Base >> 16) & 0xFFFE = 0x0000;

R_TMOUT_RANGE = ((Reass_Table_Base >> 9) & 0x00FF) |
(Reass_Table_End >> 1) &0xFF00 = 0x1918;

R_QUEUE_BASE = (Queue_Base >> 16) = 0x0000;

R_SML_Q_ST_ADR = 0x3800;

R_SML_Q_ED_ADR = 0X38FE;

R_SML_Q_RD_PTR = 0x3800;

R_SML_Q_WR_PTR = 0x38FE;

R_LRG_Q_ST_ADR = 0x3900;

R_LRG_Q_ED_ADR = 0x39FE;

R_LRG_Q_RD_PTR = 0x3900;

R_LRG_Q_WR_PTR = 0x39FE;

R_PCQ_ST_ADR = 0x3A00;

R_PCQ_ED_ADR = 0x3BFE;

R_PCQ_RD_PTR = 0x3A00;

R_PCQ_WR_PTR = 0x3A00;

R_EXCP_Q_ST_ADR = 0x3C00;

R_EXCP_Q_ED_ADR = 0x3DFC;

R_EXCP_Q_RD_PTR = 0x3C00;

R_EXCP_Q_WR_PTR = 0x3C00;

Clock_Period = 50 ns;

Timeout_Period = 60 seconds;

Number of descriptor accesses for aging packets = 64; /* arbitrary */

Time interval between descriptor accesses = 60 seconds / 64 / 65536 = 14.3 µs;

Number of clock periods between descriptor accesses = 14.3 µs / Clock_Period = 286 = 0x11E;

R_PKT_TM_CNT = ((0x100 - 0x40) << 8) | (0x100 - (0x11E>>4)) = 0xC0EF;

R_SML_BUF_CHK = (Small_buf_size - 52);

R_LRG_BUF_CHK = (Large_buf_size - 52);

R_LRG_BUF_SIZE = Large_buf_size;

Calculation of SARA-S Register Values for a Small System

SARA-S Register Set-up:

Descriptor_Table_Base = 0x000000;

VC_Table_Base = 0x002000;

Num_Valid_VC_Bits = 9;

Queue_Base = 0x001C00;

S_DESC_BASE_ = (Descriptor_Table_Base >> 16) & 0xFFFC = 0x0000;

S_VC_LKUP_BASE = ((VC_Table_Base >> 8) & 0xFFF8) |
(16 - Num_Valid_VC_Bits) = 0x0027;

S_QUEUE_BASE = (Queue_Base >> 16) = 0x0000;

S_PRQ_ST_ADR = 0x1C00;

S_PRQ_ED_ADR = 0x1DFE;

S_PRQ_RD_PTR = 0x1C00;

S_PRQ_WR_PTR = 0x1C00;

S_TCQ_ST_ADR = 0x1E00;

S_TCQ_ED_ADR = 0x1FFE;

S_TCQ_RD_PTR = 0x1E00;

S_TCQ_WR_PTR = 0x1FFE;

S_RQ_REG_A_0 = (0x400 | 0xD4); /* 50 Mbps with Pre-scalar = 0 & Queue Enabled */

S_RQ_REG_A_1 = (0x400 | 0xD4); /* 50 Mbps with Pre-scalar = 0 & Queue Enabled */

S_RQ_REG_A_2 = (0x500 | 0xF3); /* 45 Mbps with Pre-scalar = 1 & Queue Enabled */

S_RQ_REG_A_3 = (0x500 | 0xF3); /* 45 Mbps with Pre-scalar = 1 & Queue Enabled */

S_RQ_REG_B_0 = (0x600 | 0xF8); /* 20 Mbps with Pre-scalar = 2 & Queue Enabled */

S_RQ_REG_B_1 = (0x600 | 0xF8); /* 20 Mbps with Pre-scalar = 2 & Queue Enabled */

S_RQ_REG_B_2 = (0x700 | 0xFB); /* 10 Mbps with Pre-scalar = 3 & Queue Enabled */

S_RQ_REG_B_3 = (0x700 | 0xFB); /* 10 Mbps with Pre-scalar = 3 & Queue Enabled */

The Mode, Mask registers in the SARA-R/SARA-S and the VP_FILTER register in the SARA-R should be set based on the implementation of the system.

Control Memory Maps for a Large System

The control memory maps for the SARA-R and SARA-S could each look like the following:

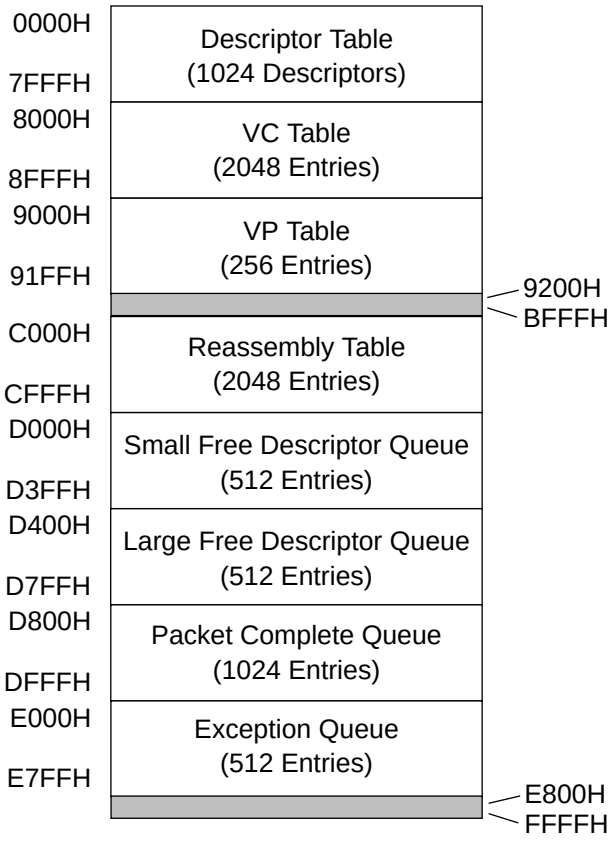


Figure A-3. SARA-R Control Memory Map for Large System

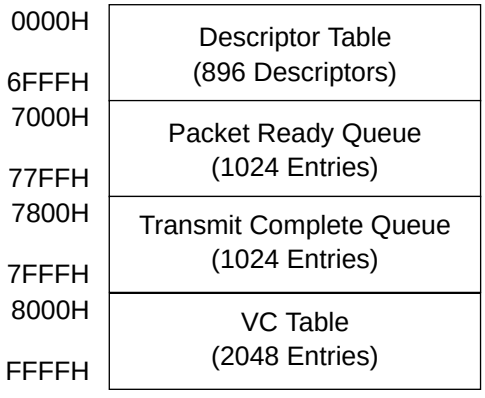


Figure A-4. SARA-S Control Memory Map for Large System

Calculation of SARA-R Register Values for a Large System

SARA-R Register Set-up:

Descriptor_Table_Base = 0x000000;
VC_Table_Base = 0x008000;
Num_Valid_VC_Bits = 11;
VP_Table_Base = 0x009000;
Reass_Table_Base = 0x00C000;
Reass_Table_End = 0x00CFFF;
Queue_Base = 0x00D000;
R_DESC_BASE = (Descriptor_Table_Base >> 16) & 0xFFFC = 0x0000;
R_VC_LKUP_BASE = ((VC_Table_Base >> 8) & 0xFFF8) |
(16 - Num_Valid_VC_Bits) = 0x0085;
R_VP_LKUP_BASE = (VP_Table_Base >> 8) & 0xFFFE = 0x0090;
R_REASS_BASE = (Reass_Table_Base >> 16) & 0xFFFE = 0x0000;
R_TMOUT_RANGE = ((Reass_Table_Base >> 9) & 0x00FF) |
(Reass_Table_End >> 1) & 0xFF00) = 0x6760;
R_QUEUE_BASE = (Queue_Base >> 16) = 0x0000;
R_SML_Q_ST_ADR = 0xD000;
R_SML_Q_ED_ADR = 0XD3FE;
R_SML_Q_RD_PTR = 0xD000;
R_SML_Q_WR_PTR = 0xD3FE;
R_LRG_Q_ST_ADR = 0xD400;
R_LRG_Q_ED_ADR = 0xD7FE;
R_LRG_Q_RD_PTR = 0xD400;
R_LRG_Q_WR_PTR = 0xD7FE;
R_PCQ_ST_ADR = 0xD800;
R_PCQ_ED_ADR = 0xDFFE;
R_PCQ_RD_PTR = 0xD800;
R_PCQ_WR_PTR = 0xD800;
R_EXCP_Q_ST_ADR = 0xE000;
R_EXCP_Q_ED_ADR = 0xE7FC;
R_EXCP_Q_RD_PTR = 0xE000;
R_EXCP_Q_WR_PTR = 0xE000;
Clock_Period = 50 ns;
Timeout_Period = 60 seconds;
Number of descriptor accesses for aging packets = 32; /* arbitrary */
Time interval between descriptor accesses = 60 seconds / 32 / 65536 = 28.6 µs;
Number of clock periods between descriptor accesses = 28.6 µs / Clock_Period = 572 = 0x23C
R_PKT_TM_CNT = ((0x100 - 0x20) << 8) | (0x100 - (0x23C>>4)) = 0xE0DD;
R_SML_BUF_CHK = (Small_buf_size - 52);
R_LRG_BUF_CHK = (Large_buf_size - 52);
R_LRG_BUF_SIZE = Large_buf_size;

Calculation of SARA-S Register Values for a Large System

SARA-S Register Set-up:
Descriptor_Table_Base = 0x000000;
VC_Table_Base = 0x008000;
Num_Valid_VC_Bits = 11;
Queue_Base = 0x007000;
S_DESC_BASE_ = (Descriptor_Table_Base >> 16) & 0xFFFC = 0x0000;
S_VC_LKUP_BASE = ((VC_Table_Base >> 8) & 0xFFF8) | (16 - Num_Valid_VC_Bits)
= 0x0085;
S_QUEUE_BASE = (Queue_Base >> 16) = 0x0000;
S_PRQ_ST_ADR = 0x7000;
S_PRQ_ED_ADR = 0x77FE;
S_PRQ_RD_PTR = 0x7000;
S_PRQ_WR_PTR = 0x7000;
S_TCQ_ST_ADR = 0x7800;
S_TCQ_ED_ADR = 0x7FFE;
S_TCQ_RD_PTR = 0x7800;
S_TCQ_WR_PTR = 0x7FFE;

S_RQ_REG_A_0 = (0x400 | 0xD4); /* 50 Mbps with Pre-scalar = 0 & Queue Enabled */
S_RQ_REG_A_1 = (0x400 | 0xD4); /* 50 Mbps with Pre-scalar = 0 & Queue Enabled */
S_RQ_REG_A_2 = (0x500 | 0xF3); /* 45 Mbps with Pre-scalar = 1 & Queue Enabled */
S_RQ_REG_A_3 = (0x500 | 0xF3); /* 45 Mbps with Pre-scalar = 1 & Queue Enabled */
S_RQ_REG_B_0 = (0x600 | 0xF8); /* 20 Mbps with Pre-scalar = 2 & Queue Enabled */
S_RQ_REG_B_1 = (0x600 | 0xF8); /* 20 Mbps with Pre-scalar = 2 & Queue Enabled */
S_RQ_REG_B_2 = (0x700 | 0xFB); /* 10 Mbps with Pre-scalar = 3 & Queue Enabled */
S_RQ_REG_B_3 = (0x700 | 0xFB); /* 10 Mbps with Pre-scalar = 3 & Queue Enabled */

The Mode, Mask registers in the SARA-R/SARA-S and the VP_FILTER register in the SARA-R should be set based on the implementation of the system.

Appendix B. Packet Memory Interface Example

The SARA packet memory interface allows for a number of different memory configurations. These different configurations allow the users to select the most appropriate mode for their applications based on system latency, bus width, functional task division and bus format (e.g. multiplexed or non-multiplexed). The example in Figure B-1 shows the SARA chips interfacing to video DRAM (VRAM) where the SARA chips read or write data to the VRAM through its serial port. The SARA chips are configured in a non-multiplexed address /data bus format.

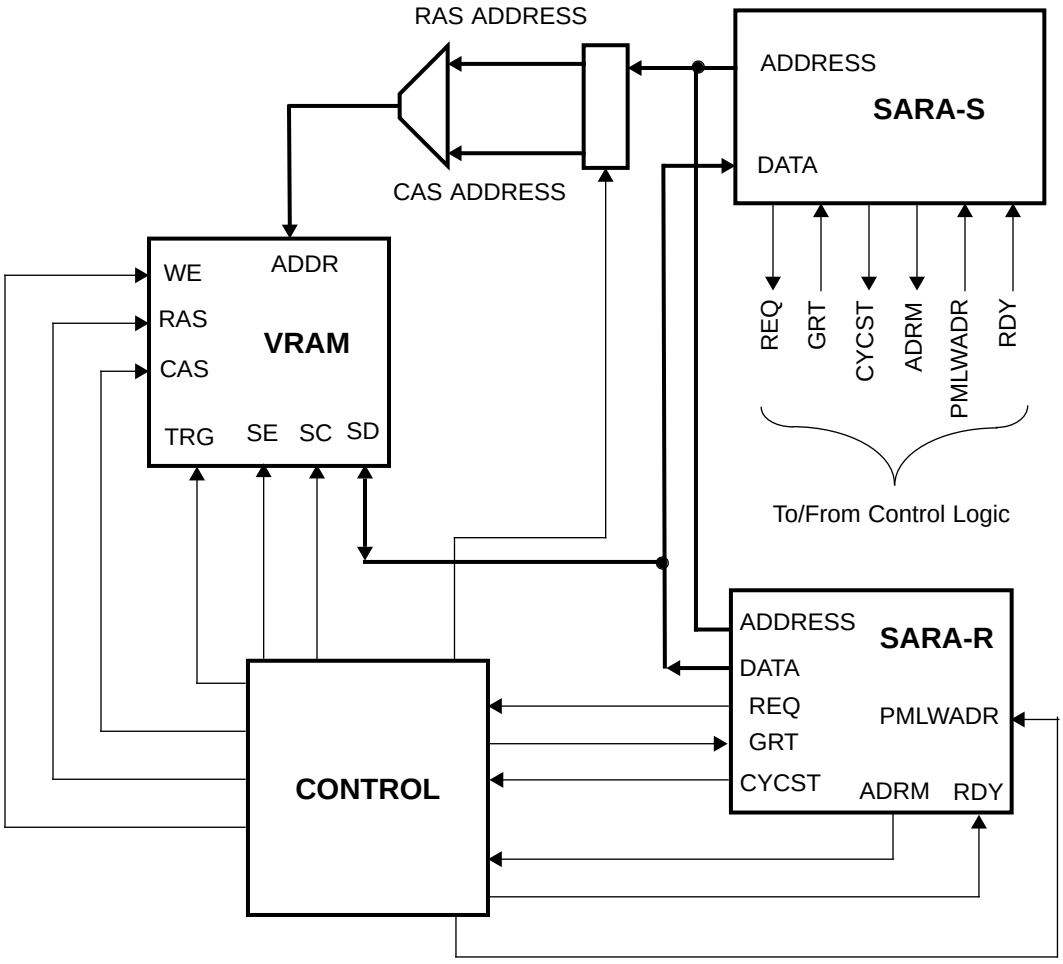


Figure B-1. Packet Memory Interface Example

The control logic block generates the appropriate signals for packet memory and the SARA chipset. The random access VRAM port and control is not shown.

Appendix C. Link Interface Example

The SARA link interface is designed for ease in connecting to external FIFOs as required by the particular application. Figure C-1 shows a specific example of a SARA link interface design. The receive link interface is designed with an external FIFO. The control block generates the appropriate signals for the SARA chipset and the link controller chips.

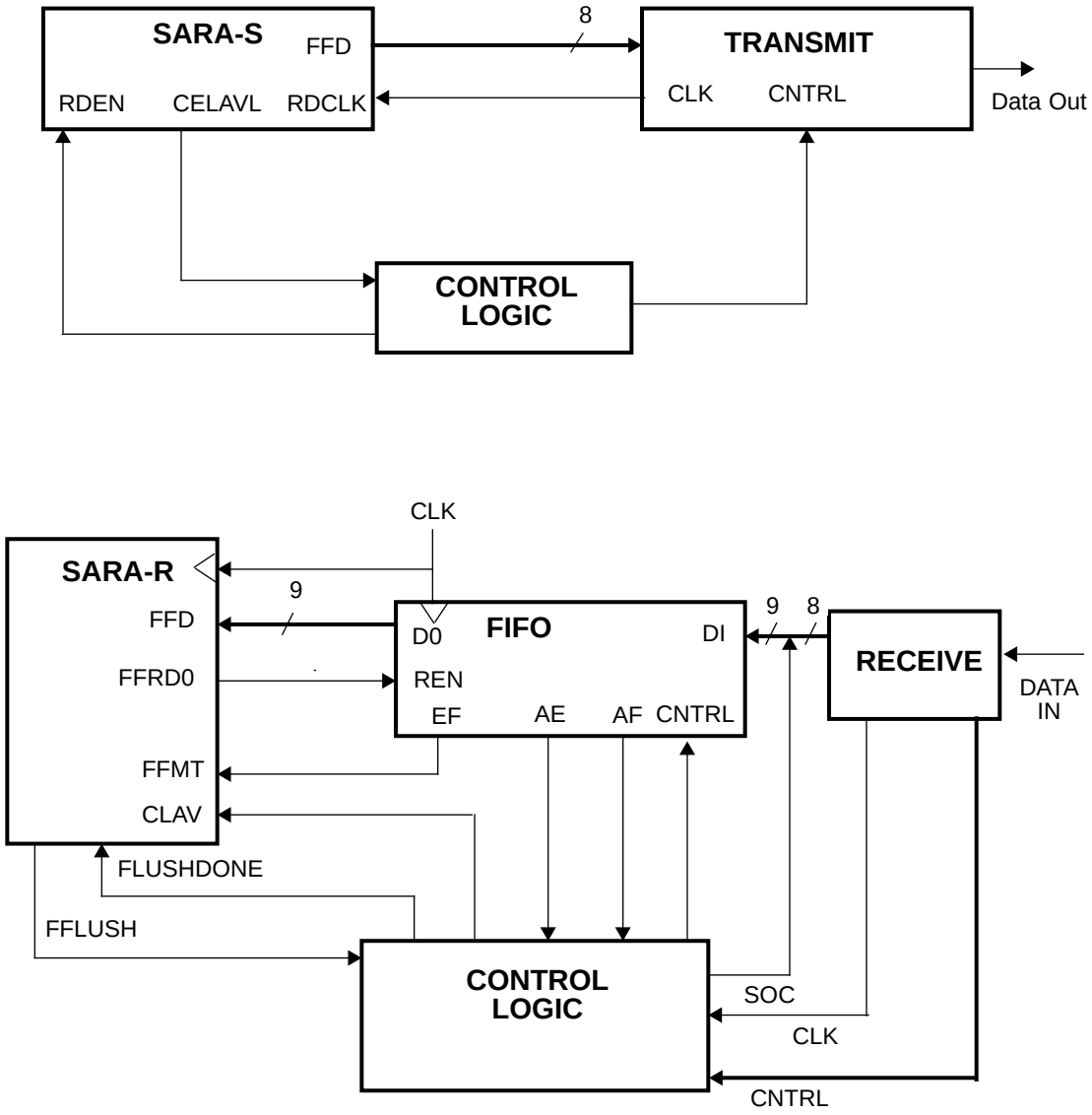
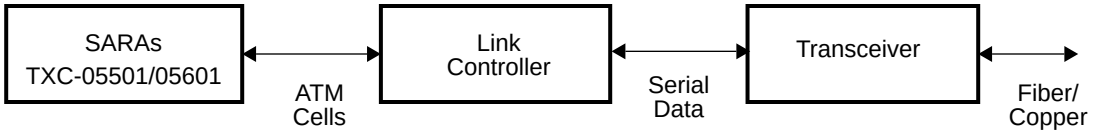
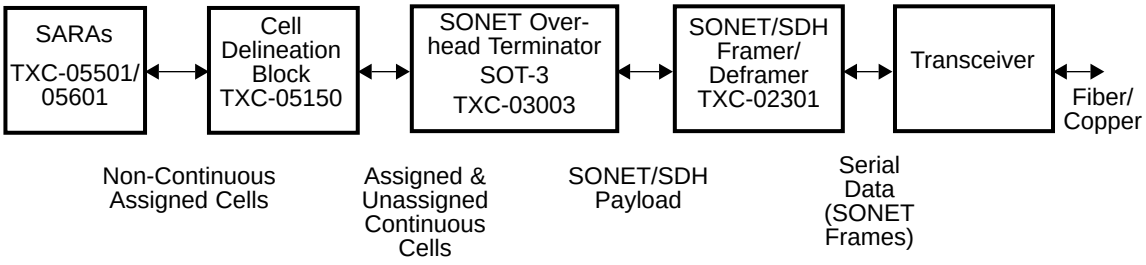


Figure C-1. Transmit and Receive Link Interface Example

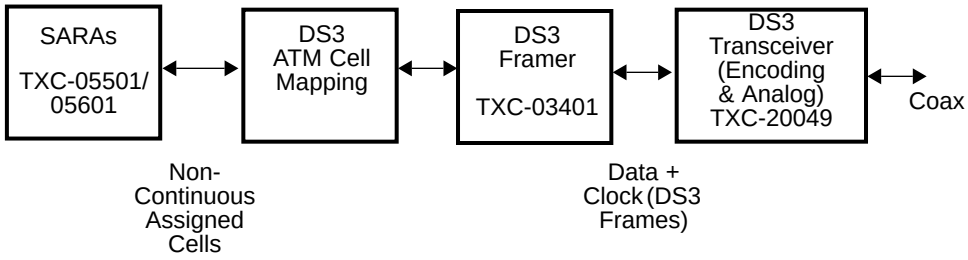
Figure C-2 shows block diagrams of the SARA chips in four different applications.



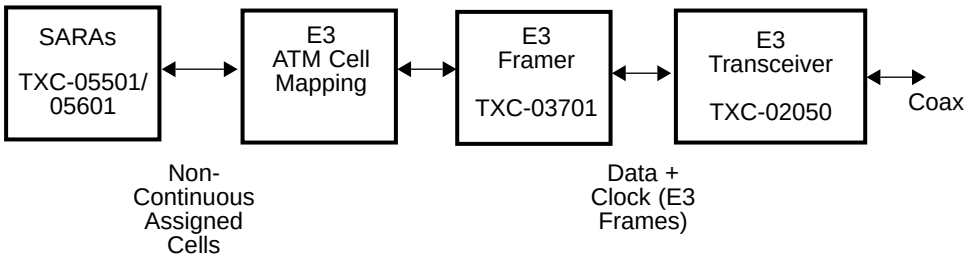
LOCAL 100 MBPS MULTIMODE FIBER/COPPER INTERFACE



SONET/SDH INTERFACE



DS3 INTERFACE



E3 INTERFACE

Figure C-2. Alternative Interface Applications

Appendix D. Control Memory Example

Figure D-1 shows a specific example of the SARA and a microprocessor (µP), both of which interface to control memory. The Arbitration and Control block arbitrates between the microprocessor and the SARA. It also generates the appropriate control signals for the microprocessor, SARA and control RAM.

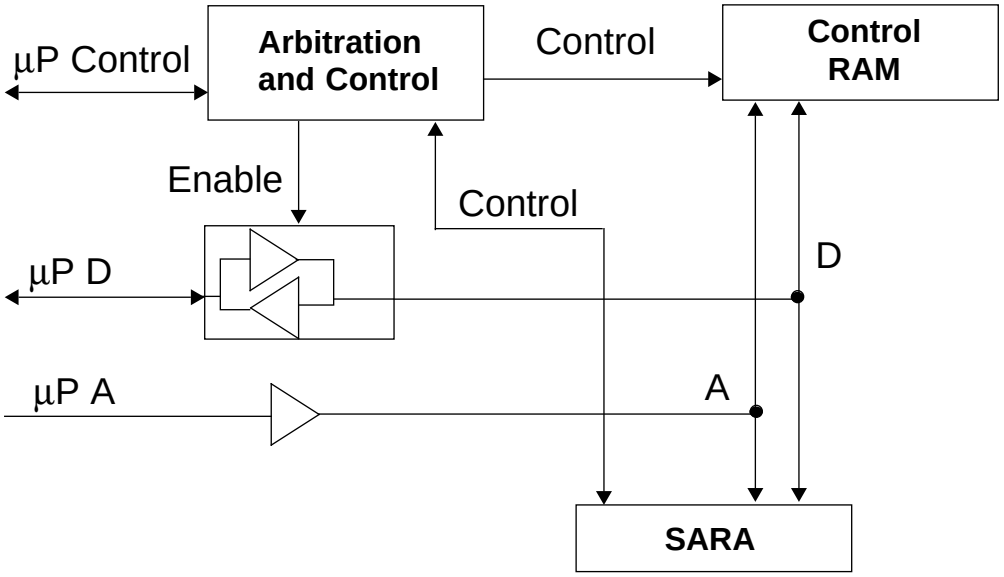


Figure D-1. Control Memory Example

Figure D-2 shows the state machine inside the arbitration and control block. The SARA access has higher priority over the processor access to control memory. The state diagram is drawn for a single-cycle access to control RAM (SRAM) with the control memory write signal programmed in the non-early write mode.

Table D-1 shows the meaning of the signals used.

Symbol	Signal or Meaning
A/B	Condition “A” results in State “B”
SREQ	SARA Request
SGRT	SARA Grant
µPREQ	µProcessor Request
µPRDY	µProcessor Ready
SMULR	SARA Multiple Request
SCYCST	SARA Cycle Start
µLAST	µProcessor Last Access (read or write)

Table D-1. Explanation of Symbols in the Control Memory State Diagram

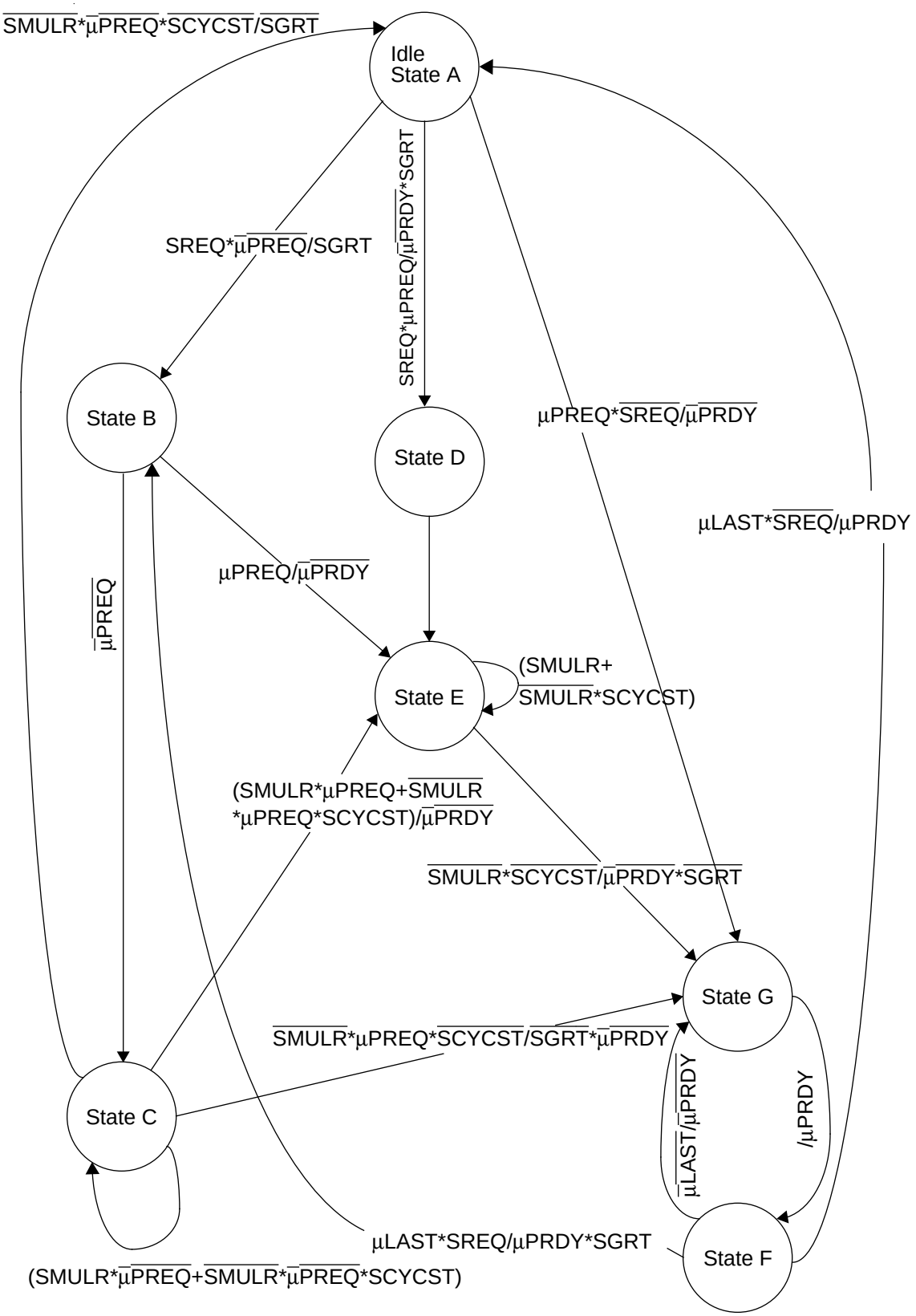


Figure D-2. Control Memory State Diagram

State A: Idle State

State B: Transition state; SGRT has been asserted in response to SREQ and in the next cycle (state C or state D depending on processor request) SCYCST will be active. No request from processor.

State C: SARA access (SCYCST or SMULR active). No request from processor.

State D: Transition state; request from SARA and processor. Priority is given to SREQ by asserting SGRT. In the next cycle, SCYCST will be active. Processor request is pending.

State E: SARA access (SCYCST or SMULR active). Processor request is pending.

States F & G: Processor Access.

Thus, the transitionary states B and D are similar except for processor request. States C and E are similar except for the state of the processor request.

Appendix E. SARA Performance Examples

Table E-1 and Table E-2 show typical performance for the SARA-S and SARA-R respectively, with the conditions under which such performance was measured.

Cell Interface	20 MHz Link Clock	12.5 MHz Link Clock
8-bit	147 Mbps	95 Mbps
16-bit	154 Mbps	150 Mbps

Table E-1. SARA-S Throughput Performance for AAL 3/4

Conditions:

1. SARA-S clock is 20 MHz.
2. 0-wait state for control and packet memory data transfers.
3. One clock cycle to obtain the bus for control and packet memory, i.e. no contention.
4. 10-cell packet segmented.
5. Packet memory in 32-bit mode.

Cell Interface	20 MHz SARA-R Clock
8-bit	117 Mbps
16-bit	181 Mbps

Table E-2. SARA-R Throughput Performance for AAL 3/4

Conditions

1. SARA-R clock is 20 MHz.
2. 0-wait state for control and packet memory data transfers.
3. One clock cycle to obtain the bus for control and packet memory, i.e. no contention.
4. Five cell packet received.
5. Packet memory in 32-bit mode.

Appendix F. Congestion Control in SARA

The SARA supports Backward Explicit Congestion Notification (BECN) scheme for congestion control. When congestion is experienced in the network, the network may issue congestion notification cells. Congestion notification cells are those identified either by the most significant bit of the first byte in the cell header (MSB of the GFC field) being set to ‘‘1’’ or by special OAM F5 end-to-end cells (PTI field in the cell header is 101). The OAMCC_CHK register is used to check the first byte of an OAM F5 end-to-end cell to determine if the cell is a congestion notification cell. This check is enabled by the CC_OAMF5_EN bit in the SARA-R Mode Register 1. Congestion notification using GFC bit is enabled by setting to ‘‘1’’ CC_GFC_EN bit of the SARA-R Mode Register 1.

When the SARA-R receives a congestion notification cell, it stores the 4-byte cell-header (the HEC byte is dropped) and the 48-byte payload of the congestion control cell in the raw cell queue. The SARA-R transfers the congestion control information to the SARA-S on the congestion control interface (CCXFER, CCHLD and CCDATA pins). If the raw cell queue is full or if the SARA-S is not ready to receive the congestion control information (CCHLD is 1), the congestion notification cell is dropped. The storing of congestion notification cells in the raw cell queue is enabled by setting the CC_XFER_EN bit in the SARA-R Mode Register 1. The transfer of the congestion control information to the SARA-S will not be affected when storing of the cells in the raw cell queue is disabled.

The SARA-S takes action on the congestion control information received from the SARA-R if bit 3 of the SARA-S Mode Register 1 is set to ‘‘0’’.

When a congestion notification cell is received, if CCHLD is inactive, the SARA-R drives CCXFER active and shifts out 26 bits of serial data on the CCDATA pin. Once the SARA-S receives the congestion control information, it forces CCHLD active until it has processed the information and is ready to receive new congestion control information. The bits that are transferred from the SARA-R to the SARA-S are shown in Figure 3-15. If the circuit is set up to reassemble packets using VCI and MID, then the 16-bit VCI should match the VC Table Index (the SARA-S descriptor table entry). If the circuit is setup to reassemble packets using VPI and VCI fields, then bits 16 to 1 of the 24-bit reassembly table pointer shown in Figure 5-11 should match the VC Table Index for that particular circuit.

Example: A simple example on how to setup the VP/VC Tables is given below (note that this is not the only way to set up the tables):

For VP/VC reassembly:

VPI=00000110 VCI= 0000 0001 1001 1001

VP Table entry: 6 LSB’s of VPI, 6 bits (arbitrary), 0001 (to use 10 VCI bits for reassembly)

000110 000000 0001

Let Reassembly base register = 00010000. The Reassembly table pointer is

0001000 000110 0110011001 0

VC Table Index: Match bits 16-1 of reassembly table pointer.

0001 1001 1001 1001

For VCI/MID reassembly:

VPI=00110001 VCI= 0000 0001 1001 1001 MID=0000 0010 11

VC Table entry: 12 LSB’s of VCI, 0011 (to use 9 MID bits for reassembly)

000110011001 0011

The reassembly table pointer constitutes 7 MSB’s from the Reassembly Base register, 7 LSB’s from VC Table Entry and 9 LSB’s from MID.

Assuming Reassembly Base register = 00010000, the reassembly table pointer is

0001000 0001100 000001011 0

VC Table Index: match the 16-bit VCI

0000 0001 1001 1001

The congestion code is that which is carried in the second byte of the 48-byte congestion control cell payload. If the congestion code is 01 H, the congestion is considered moderate and the segmentation peak rate on that VC is throttled by one *notch*. Each notch reduces the peak rate on the VC by one half of the peak rate of the previous notch. The fifth notch stops segmentation on that particular VC. If the congestion code is 02 H, the congestion is considered extreme and the segmentation of packets on that particular VC is immediately stopped (drops down to the fifth notch).When no congestion cells arrive for a throttled VC during a recovery time-out period, the SARA-S will increase the peak segmentation rate to the previous notch. If no further congestion cells are received, this process will continue until the VC is operating at its programmed full peak rate. The CCMODE bits (see Figure 6-5 and Table 6-1) in the VC Table define the rate of recovery of the segmentation peak rate. These mode bits define the number of cell transfer periods (based on the reduced peak rate) that must pass without encountering a congestion notification cell to cause recovery of the segmentation peak rate.

Under all circumstances, cells are not discarded - only segmentation is stopped. This happens gracefully at cell boundaries.

Appendix G. SARA Time-Out Operation

The SARA-R packet time-out function allows for packets undergoing reassembly to be monitored with respect to their aging prior to completion of reassembly. The purpose for this function is to terminate the reassembly of those packets which don't complete reassembly within an expected duration. In the event that the source of traffic on a given virtual circuit failed during the middle of a packet transmission, the receiving device should determine that such was the case and terminate the reassembly of that packet, freeing up the associated space in packet RAM.

The SARA-R provides for time-out capability through the use of four 16-bit registers:

PKT_TM_CNTPacket Time-out and Count Register (section 5.20)

INTRVL_CNTRPacket Aging Interval Counter (section 5.21)

TMOUT_INDXTime-out Index Register (section 5.22)

TMOUT_RANGETime-out Index Range Register (section 5.23)

The packet time-out function will age all packets undergoing reassembly at the same rate regardless of the individual packet size or rate of cells received. Thus all packets are allowed the same amount of time to complete reassembly before they are terminated.

Since a given packet may begin reassembly at any time (whenever the first cell of such a packet is received), it is necessary to associate a different aging counter with each packet. This is accomplished with an 8-bit Packet Time-out Count field in the SARA-R buffer descriptor table entry allocated in control memory for each packet undergoing reassembly. This field is initialized by the SARA-R at the start of packet reassembly with the value found in the upper 8 bit of the PKT_TM_CNT register. Hence each packet begins its reassembly with the same age value. Each 8-bit Packet Time-out Count field in control memory is incremented periodically by the SARA-R. If a packet time-out count field overflows, the reassembly associated with that packet is terminated. The SARA-R will set the PTE bit to '1' in the descriptor status field and return the descriptor to the processor by writing it into the packet complete queue.

The INTRVL_CNTR is a free-running 12-bit counter and is used to age packets, or in other words, is used to increment the Packet Time-out Count fields for each packet. Since too many control memory accesses would be required to access all descriptor table entries at once, the SARA-R increments one Packet Time-out Count field at a time in a round robin fashion. Each time the INTRVL_CNTR overflows, the SARA-R increments the Packet Time-out Count field in the descriptor table entry which is currently pointed to by the TMOUT_INDX register. Each time the INTRVL_CNTR rolls over, the TMOUT_INDX register is also incremented. Thus, the next time the INTRVL_CNTR overflows, the SARA-R ages the next packet. The lower 8-bits of the PKT_TM_CNT register is used as a reset value of the upper 8 of 12 bits of the INTRVL_CNTR. This controls how often the 12-bit counter rolls over.

Note that the aging will occur if a packet is active in the reassembly table. If the TMOUT_INDX register is pointing to a virtual circuit which is not currently active, then no packet is aged. The TMOUT_INDX register is simply incremented and the cycle begins again. The TMOUT_INDX scans the full range, i.e., it counts from 0 - (64K-1). This allows for a range independent time-out function. Only the VCs that fall in the range specified by the TMOUT_RANGE register are subject to the aging process.

The aging function is independent of the size of the reassembly table which must be stepped through to age each packet. The TMOUT_RANGE register defines the dimension of the reassembly table which is to be aged. Figure 5-13 shows how the lower 8-bits and upper 8-bits of the TMOUT_RANGE register respectively form a portion of the starting and ending addresses of the reassembly table. Since the two bytes constitute bits 9-16 of the respective addresses, increments of 256 locations may be used.

To summarize, the PKT_TM_CNT provides the set-up values for the packet time-out operation. The lower 8-bits refresh the INTRVL_CNTR and therefore determine the frequency of aging events in total. The upper 8-bits set the beginning age of each packet and therefore determine how many aging events for that particular packet are required for termination.

Aging event duration = (256-*value1*) x 2^4 x clock period

where *value1* is the number programmed in the lower byte of the PKT_TM_CNT register, and (256-*value1*) x 2^4 is the number of clock cycles needed to overflow the INTRVL_CNTR from its reset value.

Packet aging event duration =Time-out Index range roll over time = 65536 x (aging event duration)

Packet time-out duration = (256-*value2*) x (Packet aging event duration)

where *value2* is the number programmed in the upper byte of the PKT_TM_CNT register.

Example: (the numbers are not realistic)

For a 20 MHz clock, the SARA clock period is 50 ns.

Let each packet age 100 seconds before termination. Set the packet time-out count field in each descriptor field such that it overflows after 100 increments (arbitrary). This results in a packet aging event duration of 1 second. The value to be programmed in the upper byte of PKT_TM_CNT register is (256 - 100 = 156 = 9CH).

Program the interval counter to overflow 64K times per second => all packets time-out count field is incremented once each second. After 100 increments (equivalently 100 seconds) all packets have timed out. The interval counter has overflowed 6400K times.

The value to be programmed in the lower byte of PKT_TM_CNT register (same as the reset value of the upper 8 bits of the INTRVL_CNTR) is

$256\text{-}1.0/ (65536 \times 2^4 \times 50 \times 10^{-9}) = 256\text{-}19.07 = 236.93 = EDH$

Therefore, PKT_TM_CNT = 9CEDH

Given the above register values,

Packet aging event duration = $65536 \times (256\text{-}237) \times 2^4 \times 50 \times 10^{-9} = 0.996$ seconds

Packet time-out duration = 0.996 x 100 = 99.6 seconds

Assume a time-out range of 512 packets. Note that this has nothing to do with the duration of the packet aging since the TMOUT_INDX register scans the full 64K range.

TMOUT_RANGE = FFFE H (512 entries from FE00H to FFFFH)

Note: Since the aging process is asynchronous to the start of reassembly of a packet, using the above values would result in a packet time-out duration between 98.6 to 99.6 seconds. To ensure a minimum time-out duration of 99.6 second, the PKT_TM_CNT register in the above example should be programmed to 9BEDH (as opposed to 9CEDH). This would result in a packet time-out duration between 99.6 seconds to 100.6 seconds.

Appendix H. SARA Chipset Deviation List

The following deviations describe differences, as of the publication date of this Manual, between the actual functional performance of the identified SARA chips and the specified functional performance in this Technical Manual. Contact the TranSwitch Applications Department for current information.

SARA-S (For products marked TXC-05501-ACPQ or TXC-05501-BCPQ)

1. Packet-length count in the buffer descriptor is incorrectly decremented for AAL5 & OAM cells.

Explanation: The packet-byte count field in the descriptor is decremented by 52 instead of 48 for AAL5 packets and OAM cells.

Work Around: Load the packet-byte count field in the descriptor field with: [L*(52/48)-4] where L = the actual CPCS-PDU length. (Note: AAL5 CPCS-PDUs are always a multiple of 48 bytes so this calculation always equals an integer value.)

2. When both cell output buffers are full in the SARA-S and the header of the cell has not been sent to the output cell interface, the header of the first cell may become corrupted.

Normal Operation: After the cells are segmented they are sent to the output cell payload buffers where they will be appended with the cell header.

Deviation: Under certain conditions when the external interfaces can not keep up with the bursty cell output rate of SARA-S and both output cell buffers are full the SARA-S may corrupt the cell's header. The cause of the problem is that there is only one cell header buffer in the SARA-S, but two payload buffers. If a cell is ready for transmission by SARA-S, in which case Cell Available is asserted by SARA-S but transmission of the cell is delayed because Read Enable is not asserted by the external logic, then SARA-S may begin processing next packet cell and overwrite the cell header buffer, causing the problem.

Work Around: The external logic should disable either CGRT or CRDY when CELAVL is active and the RDEN signal is inactive.

Note: Many systems use different clocks for the SARA-S cell interface and System Clock, so in the external logic, be sure to resynchronize the new signal that causes the Control Memory to be not ready to System Clock.

3. Preempting packet memory interface in sync mode at the last access of cells.

Explanation: The packet memory data transfers can be terminated externally by de-asserting the PGRT signal. This will result in the SARA-S completing the current word transfer and requesting the bus.

If the SARA-S is preempted while the last data transfer is in progress, the SARA-S will re-request the bus and keep transfer the data without stopping.

Work Around: If this poses a problem with the hardware of the packet memory, then preempt the interface by de-asserting the PRDY* signal and de-asserting the PGRT after the next clock cycle.

4. Duplicate descriptor numbers may be returned.

Explanation: When the SARA-S chip is operated in the mode that segments multiple packets concurrently in the same rate queue, it returns duplicate descriptor numbers to the Transmit Complete Queue (see Section 4.11).

Work Around: Bit 14 of Mode Register 1, which is shown in Section 4.1.2 as a Reserved bit that must be set to "0", should instead be set to "1". When this bit is set to "1", linking of new descriptors takes place only after all vertically linked descriptors in the rate queue have one cell transmitted. During this processing, a new descriptor presented in the Packet Ready Queue must wait to be linked. Such new descriptors are held until all the vertically linked descriptors have one cell transmitted. This wait time may affect the latency for segmentation of new descriptors, depending on the depth of the vertical link of the rate queue. However, the overall device performance will not be substantially affected, since this work around causes only a slight delay in the linking of new descriptors.

SARA-R (For products marked TXC-05601-ACPQ or TXC-05601-BCPQ)

1. Preempting packet memory interface in sync mode to do single word transfers.

Explanation: The packet memory data transfers can be terminated externally be deasserting the PGRT signal. This will result in the SARA-R completing the current word transfer and re-requesting the bus.

If the SARA-R is preempted while the last data transfer is in progress, the SARA-R will complete the transfer, but will re-request the bus and repeat the last word transfer. If this repeat transfer is preempted, another repeat cycle is generated and so on. If the SARA-R is preempted using the PGRT signal during every transfer (single cycle access), it will cause the SARA-R to repeat the last transfer indefinitely (lockup).

Work Around: If this poses a problem with the hardware of the packet memory, then preempt the interface by deasserting the PRDY* signal and deasserting the PGRT after the next clock cycle.

2. Cell received counter interrupt occurs at 16 bit overflow rather than 32.

Explanation: The receive cell-counter overflow bit in the Interrupt Status Register is set when the lower 16 bits of the counter roll over instead of when the 32 bits roll over.

Work Around: When this interrupt occurs, read the cell-counters and update the s/w counters, if any. Do not add any 2^{32} to the s/w counters as one would if the 32-bit counter overflow worked correctly.

3. CBR mode on SARA-R may corrupt the first two bytes of 48-byte payload.

Explanation: The corruption may happen if a cell on a VCI setup for CBR traffic is received, followed by another cell on any VCI. The SARA-R reads in the header and the first two bytes of the next cell even if these bytes of the previous (CBR) cell have not yet been transferred to the packet memory. The first two payload bytes of the new cell overwrite the first two payload bytes of the CBR cell resulting in the above bug.

Work Around: The CLAV signal should not be asserted under any of the following circumstances:

1. While FFRD(0) is asserted.
2. Until the first word of the current cell is transferred to the packet memory.

3. Until there is no activity on the packet memory interface for five clock periods after the previous cell was read in (i.e., FFRD(0) signal was deasserted). This is to cover for cells that are not transferred to the packet memory (i.e., idle cells, cells with bad HEC, and exception error).

4. TMOUT_RANGE register is incorrectly read.

Explanation: The TMOUT_RANGE register is written correctly, but read incorrectly.

Work Around: None.

5. Packet memory interface 16-bit, Big-Endian mode puts out words in incorrect order.

Explanation: When the packet memory interface is 16-bit and Big-Endian mode, the SARA-R puts out word1 first and then word0.

Work Around: Use packet memory data lines PD (31:16) instead of PD (15:0) and parity lines PD (35:34) instead of PD (33:32). No work around is necessary for the 32-bit (Big-Endian or Little-Endian modes) or 16-bit Little-Endian mode. This error is only in the SARA-R, not in the SARA-S.

6. Deactivating CGRT may not preempt SARA-R data transfer to/from control memory.

Explanation: Under the Data Transfer heading in section 3.3.2.4 on page 3-22, the fifth paragraph indicates that the SARA devices can be preempted on the control memory interface by deactivating the CGRT input signal. This is not always true for the SARA-R device. If the CGRT input of the SARA-R is deactivated before both its CCYCST* and CMULR outputs are inactive the SARA-R may not release the data bus. In addition, the address bus signals sent by the SARA-R could become corrupted when the SARA-R device again requests access to the control memory.

Work Around: Preemption of the SARA-R device after it has commenced a data transfer should not be attempted. It is necessary to wait until the control memory cycle is completed. The memory cycle is complete when both CCYCST* and CMULR have become inactive.

7. Early write mode of control memory is disabled by packet-aging process.

Explanation: The early write mode of the control memory, which is described in the final paragraph of the Data Transfer heading in section 3.3.2.4, is disabled in the SARA-R device due to operation of the packet-aging process (described in sections 2.5.2, 5.20 and 5.21). When the mode bit CM_EARLY_WR is set to a “1”, the CWRT pin will not be forced to a “0” in the clock cycle before CCYCST* becomes inactive (as shown in Figures 3-5 and 3-7) but in the same clock cycle (as shown in Figure 3-8 for CM_EARLY_WR=0).

Work Around: None.

Standards Documentation Sources

Telecommunication technical standards and reference documentation may be obtained from the following organizations:

ANSI (U.S.A.):

American National Standards Institute (ANSI)
11 West 42nd Street
New York, New York 10036
Tel: 212-642-4900
Fax: 212-302-1286

The ATM Forum (U.S.A.):

ATM Forum World Headquarters
303 Vintage Park Drive
Foster City, CA 94404-1138

Tel: 415-578-6860
Fax: 415-525-0182

ATM Forum European Office
14 Place Marie - Jeanne Bassot
Levallois Perret Cedex
92593 Paris France
Tel: 33 1 46 39 56 26
Fax: 33 1 46 39 56 99

Bellcore (U.S.A.):

Bellcore
Attention - Customer Service
8 Corporate Place
Piscataway, NJ 08854
Tel: 800-521-CORE (In U.S.A.)
Tel: 908-699-5800
Fax: 908-336-2559

ETSI (Europe):

European Telecommunications Standards Institute
ETSI, 06921 Sophia - Antipolis
Cedex France
Tel: 33 92 94 42 00
Fax: 33 93 65 47 16

IEEE (U.S.A.)

The Institute of Electrical and Electronics Engineers, Inc.
Customer Service Department
445 Hoes Lane
P. O. Box 1331
Piscataway, NJ 08855-1331
Tel: 800-7014333 (In U.S.A.)
Tel: 908-981-0060
Fax: 908-981-9667

ITU-T (International):

Publication Services of International Telecommunication Union (ITU)
Telecommunication Standardization Sector (T)
Place des Nations
CH 1211
Geneve 20, Switzerland
Tel: 41-22-730-5285
Fax: 41-22-730-5991

TTC (Japan):

TTC Standard Publishing Group of the
Telecommunications Technology Committee
2nd Floor, Hamamatsucho - Suzuki Building,
1 2-11, Hamamatsu-cho, Minato-ku, Tokyo
Tel: 81-3-3432-1551
Fax: 81-3-3432-1553

List of Technical Manual Changes

This change list identifies those areas within the updated SARA Manual that have significant differences relative to the superseded SARA Manual:

Updated SARA Manual: Edition 6, April 1996

Superseded SARA Manual: Edition 5B, April 1995

The page numbers indicated below of the updated manual include changes relative to the superseded manual.

Page Number of Manual	Summary of the Change
Cover	Changed edition number to Edition 6 and issue date to April, 1996. Added TranSwitch E-Mail and Web addresses.
All	Changed edition number to Edition 6.
All	Removed Preliminary.
i	Modified Notice section.
ii	Changed edition number and issue date at top of page. Modified text.
iii - xiv	Updated Table of Contents, List of Figures and List of Tables.
xv	Added second paragraph to About This Manual section.
xvi	Modified List of Technical Manual Changes section.
1-1	Added the first two paragraphs.
3-41 - 3-65	Changed sections 3.4 (DC Characteristics) and 3.5 (AC Characteristics), including addition of modified (B) versions of Table 3-22 and Figures 3-20 through 3-28 that show the different performance of the “B” version devices.
H-2	Modified Deviation No. 4 in SARA-S (For products marked TXC-05501-ACPQ or TXC-05501-BCPQ) section.
Sources-1 - Sources-2	Updated Standards Documentation Sources.
Changes-1	Updated List of Technical Manual Changes for Edition 6.
Glossary-5	Added ITU to Glossary.

Glossary

10BASE5	An 802.3 standard: 10 Mbps transmission BASEband with 500 meters per coax segment. Standard physical layer option for CSMA/CD.
10BASE2	An 802.3 standard: 10 Mbps transmission BASEband with 185 meters per thin (RG-58A/U coax segment. Standard physical layer option for CSMA/CD.
10BASE-T	An 802.3 standard: 10 Mbps transmission BASEband over Twisted pair. Standard physical layer option for CSMA/CD.
AAL	ATM Adaptation Layer. The ATM standards that specify the procedures to be followed to segment variable length data packets into cells for transport through an ATM network and then to reassemble as they exit the network. The AAL is subdivided into the SAR and CS sublayers.
ANSI	American National Standards Institute. United States' representative to the CCITT.
ARP	Address Resolution Protocol. The Internet protocol used to dynamically map Internet addresses to physical (hardware) addresses on LANs. Limited to local area networks that support hardware broadcast.M
ATDM	Asynchronous Time Division Multiplexing. A multiplexing technique in which a transmission capability is organized in a priori unassigned time slots. The time slots are assigned to cells upon request of each application's instantaneous real need.
ATM	Asynchronous Transfer Mode. A specific form of fast packet switching technology and named by CCITT during the development of B-ISDN to distinguish it from STM. Aka "cell relay." A specific packet-oriented transfer mode using asynchronous time division multiplexing technique: the multiplexed information flow is organized in fixed blocks called cells. It is "asynchronous" in the sense that the recurrence of cells containing information from an individual user is not necessarily periodic. Contrast with CTM and PTM.
ATM Deterministic	A mode of ATM in which a constant information transfer capacity expressed in terms of a predetermined limiting value for a given service is provided to the user throughout a call.
ATM Statistical	A mode of ATM in which the information transfer capacity specified for a given service provided to the user throughout a call is expressed in terms of values of parameters such as mean, peak, standard deviation.
AUI	Attachment Unit Interface. Ethernet/802.3 transceiver host interface.
Bandwidth	A measure of information-carrying capacity.
BER	Bit Error Rate.
B-ISDN	Broadband ISDN. See ISDN.

Bridge	A device that connects two or more physical networks and forwards packets between them. Bridges can usually be made to filter packets, i.e., to forward only certain traffic. Related devices are *repeaters* which simply forward electrical signals from one cable to another, full-fledged *routers* which make routing decisions based on several criteria, and *gateways* which interconnect two networks with different communications protocols. In OSI terminology, a bridge is a *Data Link Layer Intermediate System*. See gateway, repeater, and router.
Broadcast	A packet delivery system that delivers a copy of a given packet to all hosts that attach to it is said to "broadcast" the packet. Broadcast may be implemented with hardware or software.
Brouter	A hybrid consisting of a bridge and router. See bridge and router.
CBR	Constant Bit Rate. See VBR. Constant bit rate service is a type of telecommunication service characterized by a service bit rate specified by a constant value.
CCITT	Comité Consulatif International Télégraphique et Téléphonique. Aka International Telegraph and Telephone Consultative Committee. A unit of the International Telecommunication Union (ITU) of the United Nations. An organization with representatives from the PTTs of the world. CCITT produces technical standards, known as "recommendations," for all internationally controlled aspects of analog and digital communications.
Cell	The ATM Layer PDU. A short, fixed-length packet used in the ATM high-speed packet switching technique. 53 bytes: 5 of header and 48 of payload. The header fields are: GFC (User-Network Interface only), VPI, VCI, PT, CLP, and HEC.
Cell Relay	Aka ATM.
CER	Cell Error Rate.
CLNS	ConnectionLess Network Service. See Connectionless.
CLP	Cell Loss Priority. A cell header field that is used to provide guidance to the network in the event of congestion.
Connectionless	The model of interconnection in which communication takes place without first establishing a connection. Sometimes (imprecisely) called *datagram*. Connectionless service treats each packet or datagram as a separate entity that contains the source and destination address. Examples are LANs, Internet IP and OSI CLNP, UDP, and ordinary postcards.
Connection-oriented	The model of interconnection in which communication proceeds through three well-defined phases: *connection establishment*, *data transfer*, and *connection release*. Examples are X.25, Internet TCP and OSI TP4, and ordinary telephone calls.
CONS	Connection Oriented Network Service. See Connection-oriented.
COS	Class Of Service. A parameter associated with a virtual circuit that indicates delay sensitivity and/or loss sensitivity of the connection.
CPE	Customer Premise Equipment. Generic name for transmission devices that are located in, and owned by, the public service customers.

CRC	Cyclic Redundancy Check. An error detection scheme in which the block check character is the remainder after dividing all the serialized bits in a transmission block by a predetermined binary number—or a polynomial based on the transmitted data.
CS	Convergence Sublayer. One of two AAL sublayers. Provides AAL service at the AAL-SAP.
CSMA/CD	Carrier Sense Multiple Access with Collision Detection. A contended access method in which stations listen before transmission, send a packet, and then free the line for other stations. With CSMA, although stations do not transmit until the medium is clear, collisions still occur. The access method used in Ethernet and IEEE 802.3.
CSU	Channel Service Unit. A digital DCE unit for DDS lines; interfaces with DSU on customer premises. Performs functions such as line conditioning (or equalization), signal reshaping, and loopback testing.
CTM	Circuit Transfer Mode. A transfer mode in which transmission and switching functions are achieved by permanent allocation of channels/bandwidth between the connections. Contrast with ATM and PTM.
Datagram	An abbreviated, connectionless, single-packet message from one station to another; rarely, if ever, implemented on current PDNs.
DCE	Data Communications Equipment or Data Circuit-terminating Equipment. In common usage, synonymous with modem; the equipment that provides the functions required to establish, maintain, and terminate a connection as well as the signal conversion required for communications between the DTE and the telephone line or data circuit.
DMA	Direct Memory Access. A fast method of moving data between two processor subsystems without processor intervention.
DQDB	Distributed Queue Dual Bus. IEEE 802.6 defined cell-relay standard for HSLANs/MANs.
DRAM	Dynamic Random Access Memory. See SRAM.
DSU	Digital Service Unit. DCE common equipment used to connect a customer's DTE to public network facilities or into CSU-equipped facilities.
DTE	Data Terminal Equipment. The equipment serving as the data source, the data sink, or both to the network DCE.
EIA	Electronics Industry Association. A standards group within ANSI for the electronics industry. Known for RS232C and RS422 standards that specify the electrical characteristics of interconnections between terminals and computers or between two computers.
ES	End System. Aka "host." A machine intended for running user application programs and connected to a network. In an ATM network, where an ATM connection is terminated or initiated.
ET	Exchange Termination. The part of the port interface that handles the cell processing in B-ISDN. See LT.
Ethernet	A de facto standard, developed first by Xerox and then sponsored by DEC, Intel, and Xerox (DIX). An Ethernet LAN uses coaxial cables and CSMA/CD. Ethernet is similar to an IEEE 802.3 LAN.
FDDI	Fiber Distributed Data Interface. A high-speed networking standard. The underlying medium is fiber optics, and the topology is a dual-attached, counter-rotating Token Ring. Supports only packet-switched traffic types.

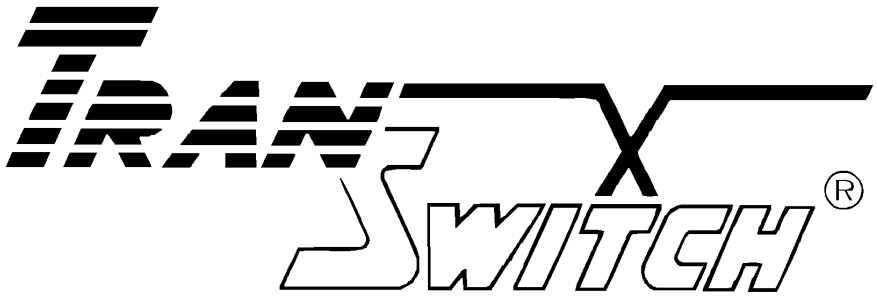
FDM	Frequency Division Multiplexing. The method of passing multiple, independent signals across a single medium by assigning each a unique carrier frequency. Hardware to combine signals is called a multiplexor; hardware to separate them is called a demultiplexor. See TDM.
Fiber Channel	Fiber-based point-to-point data communication standard.
FIFO	First-In/First-Out queue.
Flow Control	Control of the rate at which hosts or gateways inject packets into a network or internet, usually to avoid congestion. Also the procedure for controlling the transfer of messages or characters between two points in a data network—such as between a protocol converter and a printer—to prevent loss of data when the receiving device's buffer begins to reach its capacity.
Frame	Variable length, addressed data unit identified by a label at layer 2 of the OSI reference model. Aka "packet" or "message." Literally, it is a packet as it is transmitted across a serial line. See packet and PDU.
Frame Relay	A private networking packet delivery interface with an historical base in X.25.
Gateway	A special purpose, dedicated computer that attaches to two or more networks with different communications protocols and routes packets from one to the other. Gateways route packets to other gateways until they can be delivered to the final destination directly across one physical network. Gateways operate at the 4th through 7th layers of the OSI model. See bridge, repeater, and router.
GFC	Generic Flow Control. A cell header field that is used for multiplexing for access to an ATM network.
HDLC	High-level Data Link Control. A link level protocol standard by ISO.
HEC	Header Error Control. A cell header CRC field that can be used to correct single-bit errors in the header and to detect multiple-bit errors.
Header	The bits within a cell allocated for functions required to transfer the cell payload within the network.
Host	See ES.
HSLAN	High Speed Local Area Network.
IBCN	Integrated Broadband Communication Network. Aka BISDN.
ICMP	Internet Control Message Protocol. See TCP/IP.
Internet	A collection of packet switching networks interconnected by gateways along with protocols that allow them to function logically as a single, large, virtual network. When written in the upper case, Internet refers specifically to the Defense Advanced Projects Research Agency (DARPA) Internet and the TCP/IP protocols.
Internet address	The 32-bit address assigned to hosts that want to participate in the Internet using TCP/IP. Internet addresses are the abstraction of physical hardware addresses just as the Internet is an abstraction of physical networks. Actually assigned to the interconnection of a host to a physical network, an Internet address consists of a network portion and a host portion. The partition makes routing efficient.
Interoperability	The ability of software and hardware on multiple machines from multiple vendors to communicate meaningfully.
IPC	Inter-Process Communication.

IP	Internet Protocol. See TCP/IP.
ISDN	Integrated Digital Services Network. An emerging technology that is beginning to be offered by the telephone carriers of the world. ISDN combines voice and digital network services in a single medium making it possible to offer customers digital data services as well as voice connections through a single “wire.” The standards that define ISDN are specified by CCITT.
ISO	International Standards Organization. An international body that drafts, discusses, proposes, and specifies standards for network protocols. ISO is best known for its 7-layer reference model that describes the conceptual organization of protocols.
ITU	International Telecommunication Union (parent of CCITT, which is now known as its Telecommunication Standardization Sector, ITU-T).
LAN	Local Area Network. A data communications network confined to a limited geographic area (up to 6 miles or about 10 kilometers) with moderate to high data rates (100 Kbps to 100 Mbps). The area served may consist of a single building, a cluster of buildings, or a campus-type arrangement.
LT	Line Termination. The part of the port interface that handles the physical level transmission line interface in B-ISDN. See ET.
MAC	Media Access Control.
MAN	Metropolitan Area Network.
MAU	Medium Access Unit. An transceiver for 802.3 10BASE5 and 10BASE2.
Message	A complete transmission; used as a synonym for “packet,” but a message is often made up of several packets.
MIB	Management Information Base. A collection of objects that can be accessed via a network management protocol. See SMI.
MIC	Media Interface Connector. Duplex fiber connector used for FDDI.
MID	Message IDentifier.
Multicast	A technique that allows copies of a single packet or cell to be passed to a selected subset of all possible destinations.
NMS	Network Management Station. The system responsible for managing a network or a portion of the network. The NMS talks to network management agents, which reside in the managed node, via a network management protocol.
NISDN	Narrowband ISDN.
NIC	Network Interface Controller. Circuitry, usually a PC expansion card, that connects a workstation to a network.
NIU	Network Interface Unit. An adapter card that performs the packet segmentation/reassembly (via SARA) and has an ATM link interface.
NMS	Network Management Station.
OSI	Open Systems Interconnection. Refers to a seven-layer hierarchical reference structure developed by the ISO for defining, specifying, and relating communications protocols. In the OSI model, groups of communications protocols are arranged in layers.
Packet	An information block identified by a label at layer 3 of the OSI reference model. It is the unit of data sent across a packet switching network. See frame and PDU.

PCM	Pulse-Code Modulation. A modulation technique used to convert analog voice signals into digital form. Used for voice multiplexing on T1 circuits.
PDN	Public Data Network. Aka Packet Data Network. A network established and operated by a PTT, common carrier, or private operating company for the specific purpose of providing data communications services to the public. May be a PSDN or a DDS.
PDU	Protocol Data Unit. OSI terminology for "packet." A PDU is a data object exchanged by protocol machines (entities) within a given layer. See frame and packet.
POTS	Plain Old Telephone Service. Transported via PSTN and offers customers classical two way voice conversation.
PPS	Packets Per Second.
Protocol	A formal description of message formats and the rules two or more machines must follow to exchange those messages. Protocols can describe low level details of machine to machine interfaces (e.g., the order in which the bits from a byte are sent across a wire) or high level exchanges between application programs (e.g., the way in which two programs transfer a file across the Internet). Most protocols include both intuitive descriptions of the expected interactions as well as more formal specifications using finite state machine models.
PSDN	Public Switched Data Network. Transports computer data in the public domain based on X.25 protocols.
PSTN	Public Switched Telephone Network.
PT	Payload Type. A cell header field indicating the type of information in the cell payload.
PTM	Packet Transfer Mode. A transfer mode in which the transmission and switching functions are achieved by packet oriented techniques, so as to dynamically share network transmission and switching resources between a multiplicity of connections. Contrast with ATM and CTM.
PTT	Post, Telephone, and Telegraph Authority. The government agency that functions as the communications common carrier and administrator in many areas of the world.
PVC	Permanent Virtual Circuit. In a packet switched network, a fixed virtual circuit between two users; no call setup or clearing procedures are necessary; the PDN equivalent of a leased line. Contrast with SVC.
RACE	Research for Advanced Communication in Europe.
Repeater	A hardware device that propagates electrical signals from one cable to another without making routing decisions or providing packet filtering. In OSI terminology, a repeater is a *Physical Layer Intermediate System*. See bridge, gateway, and router.
Router	A system responsible for making decisions about which of several paths network traffic will follow. To do this, it uses a routing protocol to gain information about the network and algorithms to choose the best route based on several criteria known as "routing metrics." In OSI terminology, a router is a *Network Layer Intermediate System*. See bridge, gateway, and repeater.

RS232	A standard by EIA that specifies the electrical characteristics of slow speed interconnections between terminals and computers or between two computers. The specification limits speed to 20 Kbps and distance to 500 feet, but many manufacturers support speeds of 36.4 Kbps and/or longer distances.
SAR	Segmentation And Reassembly. One of two sublayers of the AAL with the functions of, at the transmitting side, the segmentation of higher layer PDUs into a suitable size for the information field of the ATM cell and, at the receiving side, the reassembly of the particular information fields into higher layer PDUs.
SARA	TranSwitch‘s Segmentation/Reassembly/DMA Engine. A two-chip chipset providing an interface from packet oriented equipment to ATM networks. The chipset supports the ATM AAL including CS and SAR functions. See SARA-S and SARA-R.
SARA-R	TranSwitch‘s Reassembly SARA. The reassembly device, one of two chips in the SARA chipset, that reassembles an incoming multiplexed stream of ATM cells into packets using either the VCI or the MID fields as the reassembly identifier. See SARA-S and SARA.
SARA-S	TranSwitch‘s Segmentation SARA. The segmentation device, one of two chips in the SARA chipset, that segments packets into cells and multiplexes those cells with cells from other packets that are undergoing segmentation. See SARA and SARA-R.
Service Bit Rate	The bit rate that is available to a user for the transfer of user information.
SCB	Shelf Control Bus.
SDH	Synchronous Digital Hierarchy. Europe’s version of SONET.
SMDS	Switched Multimegabit Data Service. A public packet switching service proposed by Bellcore.
SMI	Structure of Management Information. The rules used to define the objects that can be accessed via a network management protocol. See MIB.
SNMP	Simple Network Management Protocol. The network management protocol of choice for TCP/IP-based internets.
SONET	Synchronous Optical NETwork. The name for an advanced, fiber-based public network defined by a large family of related technical standards. See SDH.
SRAM	Static Random Access Memory. See DRAM.
STM	Synchronous Transfer Mode. A transfer mode that offers periodically to each connection a fixed-length word. Contrast with ATM.
SVC	Switched Virtual Circuit. In a packet switched network, temporary virtual circuit between two users. Contrast with PVC.
TCP/IP	Transport Control Protocol/Internet Protocol. TCP is the protocol that provides reliable, end-to-end stream transport. IP is the universal protocol of the Internet that defines the unit of transfer to be the IP datagram and provides the universal addressing scheme for hosts and gateways. Internet Control Message Protocol (ICMP), an integral part of IP, specifies error and control messages.
TDM	Time Division Multiplexing. A technique used to multiplex multiple signals onto a single hardware transmission channel by allowing each signal to use the channel for a short time before going on to the next one.

Transfer Mode	Aspects covering transmission, multiplexing, and switching in a telecommunications network.
UDP	User Datagram Protocol. The Internet standard protocol that allows an application program on one machine to send a datagram to an application on another machine. It uses IP to deliver the datagrams.
UNI	User-Network Interface.
VBR	Variable Bit Rate. See CBR. Variable bit rate service is a type of telecommunication service characterized by a service bit rate specified by statistically expressed parameters that allow the bit rate to vary within defined limits.
VC	Virtual Channel (or Virtual Circuit). One of two levels in the ATM layer of the ATM transport hierarchy. The basic unit of switching in B-ISDN. An individual logical connection. Describes the unidirectional transport of ATM cells associated by a common unique identifier value. See also VP.
VCI	Virtual Channel Identifier (or Virtual Circuit Identifier). A routing field in the header of a cell. Used to identify the virtual connection to which the cell belongs.
VP	Virtual Path. One of two levels in the ATM layer of the ATM transport hierarchy. A group of logical connections. A bundle of virtual channels that have the same endpoints. All of the cells flowing over all of the virtual channels in a single virtual path are switched together. Describes unidirectional transport of ATM cells belonging to virtual channels that are associated by a common identifier value. See also VC.
VPI	Virtual Path Identifier. A routing field in the header of a cell.
VRAM	Video Random Access Memory. A derivative of DRAM that has a separate serial access port(s) and is commonly used for video screen refresh. Sometimes used as dual-port RAMs for data buffers/storage.
WAN	Wide Area Network.



TranSwitch VLSI:
Powering Communication Innovation

TranSwitch Corporation • 8 Progress Drive • Shelton, CT 06484 • USA • Tel: 203-929-8810 • Fax: 203-926-9453