Train Communication Network

MVBC 01

Multifunction Vehicle Bus Controller

Data Sheet

Document: 3EHL 420 441 / BAE 3010/0007 Revision C/BAL-8104

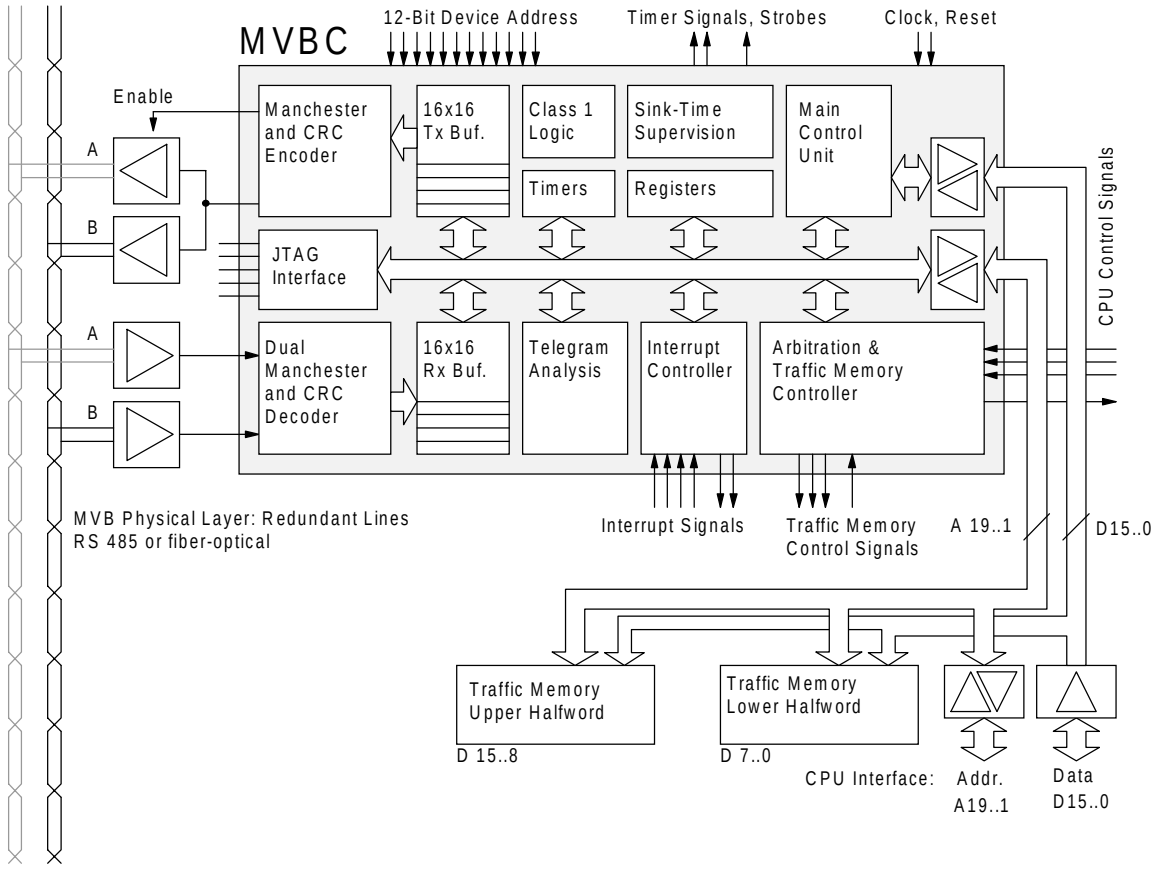


ABB Daimler-Benz Transportation (Switzerland) Ltd

ADtranz

REVISION HISTORY

Revision	Author	Dept.	Date	Status
Preliminary	zur Bonsen	BAEI-1	92-10-15	Draft Release
Prelim. Rev. A	zur Bonsen	BATC	93-10-07	Draft Release
Prelim. Rev. B	zur Bonsen	BATC	94-04-26	Draft Release
Revision C	zur Bonsen	BALT	97-07-25	Official Release

DISCLAIMER

ABB Daimler-Benz Transportation (Switzerland) Ltd. makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The company named above assumes no responsibility for any errors that may appear in this document. No liability is assumed arising out of application or use of this product described herein.

The company named above reserves the right to make changes to this product herein to improve reliability, functionality or design. The information in this document and the product described by this document is subject to change without prior notice.

QUALITY ASSURANCE

The Vehicle Control family has been developed and is produced under the requirements ruled by the ISO 9001 quality assurance standards. ABB Daimler-Benz Transportation (Switzerland) Ltd. is certified for complying with the ISO 9001 standards.

TABLE OF CONTENTS

1 INTRODUCTION

1.1 Abstract

The Multifunction Vehicle Bus (MVB) is a standard communication medium to transport and exchange data among attached devices. These devices, which are physically connected to the bus, may vary in function, size, performance and at the physical layer level. In order to allow these various devices to communicate with each other a common communication interface has been designed which is independent of the chosen physical layer and functions associated with each device.

The Multifunction Vehicle Bus Controller (MVBC) is the interface component between the MVB independent circuits and the actual physical layer of the MVB (excluding physical layer drivers). The MVBC, when configured accordingly, can be used in Class 1, 2, 3, 4 devices as defined in the IEC TCN Standard [1]. The MVBC is available in form of a 100-pin Application Specific Integrated Circuit (ASIC).

1.2 Intended Audience

Persons who are involved in hardware or software development projects where the MVBC ASIC is applied.

1.3 Scope

This document provides detailed information about the MVBC ASIC. Information, which are not directly related to the MVBC, for example MVB protocol specifications, are not part of this document. Further Information about the MVB can be found in [1].

1.4 Definitions and Abbreviations

All MVB-related terms are defined in [1]. Other terms can be referenced in the Index (Appendix K: Index).

Not included:

Common computer science and electrical engineering terms;
names of external I/O pins; see section 2.5.2 instead;
timing parameters; see section chapter 7 instead.

Abbreviations:

ALO	Active Level Overbalance (bit in PCS)
AMFX	All Master Frames Transmitted (interrupt)
ARB*i*	Arbitration Strategy Selection bits (1..0; part of SCR)
BA	Bus Administrator
BAC	Bus Arbitration Chip, co-processor unit for BAP15-2/3
BAP	Procontrol Bus Access Processor (BAP15-2/3): Predecessor of MVBC
BAS	Bus Administrator Software
BNI	Bus Not Idle (bit in PCS)
BUSY	Busy Indicator (MCU busy handling telegrams, bit in MR)
BT	Bit Time for signals transfer (1 BT = 666 ns = 16 clock cycles)
BTI	Bus Timeout Interrupt
CM	Communication Mode (part of 1st word of Message Data)
CPE*i*	Clear Pending Event (0 or 1; part of PCS)
CRC	Cyclic Redundancy Error indicator (bit in PCS)
CS	Check Sequence (8-bit Cyclic Redundancy Check)
CSMF	Cancel Sending Master Frames (bit in MR)
DA	Device Address (12-bit value)
DAOK	Device Address Override Key (Register)
DAOR	Device Address Override Register

DEC Disable / Enable Counter (part of PCS)
DGA Device Group Address (12-bit value)
DMF Duplicate Master Frame (interrupt)
DNR Device Not Ready (bit in DSR)
DP Data Pointer (queue data structure)
DPR Dispatch Pointer Register
DPR2 Secondary Dispatch Pointer Register
DR Decoder Register
DSF Duplicate Slave Frame (interrupt)
DSR Device Status Report (Slave Frame body for Device Status Poll)
DTI*i* Data Transfer Interrupts (7..1)
EA*i* Event Announced (0 or 1; bit in MR)
EC Error Counter (Register)
EC*i* Event Cancellation (0 or 1, bit in MR)
EF0 Event Frame Source Port for Event Type 0 (Physical Port)
EF1 Event Frame Source Port for Event Type 1 (Physical Port)
EFS Event Frame Sink Port (Physical Port)
EM Event Mode
EMF Erroneous Master Frame (interrupt)
ESF Erroneous Slave Frame (interrupt)
ERD Extended Reply Time (bit in DSR)
ET Event Type
F-Code Function Code
FC Frame Counter (Register)
FC8 Mastership Offer Source Port (Physical Port)
FC15 Device Status Port (Physical Port)
FD Force Data Pattern (part of Force Table)
FE Forcing Enabled (bit in PCS)
FEV Frames Evaluated Interrupt
FM Force Mask Pattern (part of Force Table)
FRC Some data is forced (bit in DSR); related to data forcing
IAV Interrupt Available (bit in IVR*i*)
IE*i* Input Enable (0..2, part of PCS)
IEC International Electrotechnical Commission
IL*i* Initialization Level bits (1..0; part of SCR)
IM Intel/Motorola Mode (bit in SCR)
IMR*i* Interrupt Mask Register (0 or 1)
IPR*i* Interrupt Pending Register (0 or 1)
IR JTAG Instruction Register
ISR*i* Interrupt Status Register (0 or 1)
IVR*i* Interrupt Vector Register (0 or 1)
JTAG IEEE 1149.1: Joint Test Action Group
LA Logical Address (12-bit value)
LAA Line A Active (or *trusted*; bit in DR and DSR)
LLR Linked List Record (queue data structure)
LS Line Switchover (bit in DR)
MBC Message Broadcall (bit in SCR)
MCM Memory Configuration Mode (part of MCR)
MCR Memory Configuration Register
MCU Main Control Unit (MVBC function block)
MD Message Data
MF Master Frame (communication packet)
MFC Master Frame Checked (interrupt)
MFR Master Frame Register
MFRE Master Frame Register duplicated for Exceptions
MFS Master Frame Slot (External Register)
MO*i* Master Frame Table Offset (1..0, part of MCR)
MOS Mastership Offer Sink Port (Physical Port)
MR Master Register
MR2 Secondary Master Register
MSNK Message Sink Port (Physical Port)
MSRC Message Source Port (Physical Port)

MVB	Multifunction Vehicle Bus
NP	Next Pointer (queue data structure)
NUM	Numeric Data transferred (bit in PCS)
PAR*i*	Pending for Arbitration (0 or 1, bit in MR)
PCS	Port Control and Status Register (Set of 4 words per port)
PD	Process Data
PI	Port Index (retrieved value from PIT)
PIT	Port Index Table
PP	Physical Ports (inside Service Area)
PTD	Port temporarily disabled (bit in PCS)
QA	Queues attached (bit in PCS)
QDT	Queue Descriptor Table (External Registers)
QFPP	Quad Flat Package (chip package)
QO*i*	Queue Offset (1..0, part of MCR)
QUIET	Disable Encoder activity (bit in SCR)
RCEV	Receive Events (bit in SCR)
RLD	Redundant Line Disturbed (bit in DR and DSR)
RQ	Receive Queue
RQC	Receive Queue Complete (interrupt)
RQE	Receive Queue Exception (interrupt)
RS*i*	Reset Timer (1 or 2, part of TCR)
RTI	Reply Timeout Interrupt (typically 42.7 µs)
RXB	Receive Buffer (MVBC function block)
SCR	Status Control Register
SDD	Some Device Disturbance (bit in DSR)
SER	Service Reservation (bit in DSR)
SF	Slave Frame (communication packet)
SFC	Slave Frame Checked (interrupt)
SI*i*	Sink-Time Supervision Interval (3..0, part of STSR)
SINK	Active Sink (bit in PCS)
SLM	Single Line Mode (bit in DR)
SMF*i*	Send Master Frame Command (bit in MR, MR2)
SMFA	Send Master Frames Automatically (code for SMF*i*)
SMFE	Send Empty Master Frame Table (code for SMF*i*)
SMFM	Send Master Frames Manually (bit in MR, MR2)
SMFT	Send Master Frames Timed (code for SMF*i*)
SQE	Signal Quality Error (bit in PCS)
SDD	Some Device Disturbance (bit in DSR)
SRC	Active Source (bit in PCS)
SSD	Some System Disturbance (bit in DSR)
STO	Slave Timeout (bit in PCS)
STSR	Sink-Time Supervision Register
TA*i*	Timer Active (1 or 2, part of TCR)
TACK	Transfer Acknowledge
TC*i*	Timer Counter Register (0 or 1)
TCN	Train Communication Network (IEC Standard)
TCR	Timer Control Register
TERR	Telegram Error (bit in PCS)
TI*i*	Timer Interrupt (1 or 2)
TM	Traffic Memory (in some documents also known as Traffic Store)
TMC	Traffic Memory Controller (MVBC function block)
TMO*i*	Timeout Selection bits (1..0; part of SCR)
TQC*i*	Transmit Queue Complete (0 or 1, interrupt)
TQE	Transmit Queue Exception (interrupt)
TR*i*	Timer Reload Register (0 or 1)
TSNK	Test Sink Port (Physical Port)
TSRC	Test Source Port (Physical Port)
TWCS	Transmit with Check Sequence (bit in PCS)
TXB	Transmission Buffer (MVBC function block)
UTQ	Use Test Source Ports (bit in SCR)
UTS	Use Test Sink Ports (bit in SCR)
VEC*i*	Interrupt Vector (3..0, bit in IVR*i*)

VP Valid Page (Page Pointer; bit in PCS)
WA Write Always (bit in PCS)
WS*i* Waitstate Selection bits (1..0; part of SCR)
XI*i* External Interrupt (3..0)
XQ*i* Transmit Queue; *i* = 0 or 1
XSYN External Synchronization (bit in TCR)

Combinations of Abbreviations:

DA-PCS See DA, PCS
DA-PIT See DA, PIT
LA-FRC See LA, FRC
LA-PCS See LA, PCS
LA-PIT See LA, PIT
PP-PCS See PP, PCS

Definition of Terms:

Please refer to Appendix K: Index

1.5 Conventions

1.5.1 Bit and Byte Order

Word:	Understood as a 16-bit word unless specified otherwise (i.e. "32-bit word")
Bit Order:	Bit 15 = Most significant bit (MSB)
	Bit 0 = Least significant bit (LSB)
Byte Order:	Two orders: Big Endian and Little Endian Format
Halfword:	Half part of a 16-bit word. Size: 1 byte.
Upper Halfword:	Bits 15..8 of a word (Numeric data: 215..28)
Lower Halfword:	Bits 7..0 of a word (Numeric data: 27..20)
Low Address Byte:	Byte inside word addressed with A0=0 (even addresses)
High Address Byte:	Byte inside word addressed with A0=1 (odd addresses)

Attention: Low (or High) Address Byte does not automatically imply Upper (or Lower) Halfword. This merely depends on the byte ordering scheme (Endian format) dictated by the supported host CPU or MCU.

Big Endian (i.e. Motorola 680x0s series):
Upper Halfword = Low Address Byte
Lower Halfword = High Address Byte

Little Endian (i.e. Intel 80x86 series):
Upper Halfword = High Address Byte
Lower Halfword = Low Address Byte

1.5.2 Symbols

The vertical bar seen on the right side indicates differences in the contents as compared to the previously released revision of this document (if available). Minor changes (i.e. spelling) are not marked with this bar.

1.5.3 Stylistics

An example of a register description is shown below:

Generic Register (GR):

Brief Description: The Generic Register is not part of the MVBC. This name is used for this example only.

Generic Register (GR): Address 0yFFFFH

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	-												AB	CD	EF	GH
Init. Value:	All 0												1	0	0	0
CPU Acc.:	-												R	RW0	RW1	RW
MVBC Acc.:	-												rw	rw	rw	r

1st Row: Bit assignment. In the normal case, a separate table provides a detailed description for each bit. A lower-case "x" indicates an unused bit. This bit cannot be used for general-purpose data storage.

2nd Row: Indicates the initial values after power-up and/or asynchronous MVBC reset. If an 'x' is found on the 1st row, then the specified value will always be read out (permanently wired to '1' or '0'). Reading different values reflect a chip malfunction.

A notice "Initial Value: ..." may substitute this row.

3rd Row: Indicates access modes provided to the CPU (Upper-case symbols):

- Bit not existing or supported
R Read Access
R0 Read zero all time
R1 Read one all time
W Write Access
W0 Write access, only 0 affects register and/or system state (i.e. Reset)
W1 Write access, only 1 affects register and/or system state (i.e. Preset)

Combinations of the above are allowed.

A notice "CPU Access: ..." may substitute this row.

4th Row: Indicates access modes provided to the MVBC:
Possible symbols: -, r, r0, r1, w, w0, w1 and combinations of them
These lower-case symbols carry equivalent functions as the upper-case symbols found in the 3rd row.

A notice "MVBC Access: ..." may substitute this row.

1.6 List of Illustrations

1.7 List of Tables

1.8 Document Structure

Chapter 2: MVBC Overview
A full description of the MVBC and its interface to the physical layer MVB drivers and via Traffic Memory to the CPU are elementary parts of this chapter.

Chapter 3: Behavioral Overview
The behavioral aspects, such as data transfer procedures, port processing for all defined Function Codes, event arbitration and Master Frame Dispatching.

Chapter 4: Hardware Application Suggestions
Few suggestions are given how to interface the MVBC into a given application environment.

Chapter 5: Programming Guidelines
This chapter summarizes easy "plug-in-and-play" instructions to program the MVBC in order to obtain successful results within a short period of time.

Chapter 6: Technical Data
DC electrical characteristics, mechanical and environmental data are summarized.

Chapter 7: Timing All timing diagrams and tables are located here.

Chapter 8: Operation in System: Mounted in System, Reliability, Availability information.

Chapter 9: Miscellaneous
Ordering information, chip handling and packing information are summarized.

Appendices: Function Code summary plus corresponding programming information, detailed description of access sequence to Traffic Memory, pad characteristics and reset behavior.

2 MVBC OVERVIEW

2.1 Included Chip Features

- Full Compliance with the IEC TCN Standard, Part 3: MVB [1]

  - Process Data (PD)
  - Message Data (MD)
  - Supervisory Data (Event Polling, Device Status Polls, Mastership Offer Polls)
  - Bus Administrator and monitoring functions

- 0.8 µm ASIC (Sea of Gates)
- Easy-to-use universal hardware interface
- 100 pin plastic quad flat pack (QFPP)
- Supports computers without real-time capabilities

Communication:

- 1.5 Mbit/s data rate
- Signal Quality Checking
- Powerful error and collision detection
- Full 16-bit support
- Max. 4095 ports for logical addressed telegrams
- Class 1 Mode: Support of 16 ports @ 1-16 word
- Automatic telegram analysis and evaluation
- Event Polling over two priorities
- Manchester Biphase-L coding
- Hamming Distance: 8
- Detection of permanent transmitters (jabber hold)
- 16 K - 1 M Byte Traffic Memory
- Max. 4095 ports for device addressed telegrams
- Automatic Message Queue handling
- Timeout mechanisms
- Device Address modifiable by software

Advanced Communication Functions:

- Bit-wise data forcing capabilities
- Automatic data Comparison mechanism
- Synchro port for synchronization and data strobing
- User-supplied Check Sequences

Master Functions:

- Transmission of individual Master Frames (MF)
- Timed MF transmission at precise intervals
- Autom. Transmission of max. 32 MFs from a table
- MF-tables are cascadable with *advance requests*

Testability:

- JTAG Boundary and Internal Scan
- On-Line test on Traffic Memory possible
- High degree of ad-hoc testability and observability
- Internal RAM cells isolatable for direct RAM tests

Miscellaneous:

- Two Universal Timers generating interrupts
- Sink-Time Supervision
- Intelligent Interrupt Logic, supports vectors
- 4 external interrupt inputs supported

- External High-Precision synchronization of one Universal Timer
- Synchronized MF Transmission over multiple MVBCs controlling multiple MVBs

2.2 Device Interface Symbol

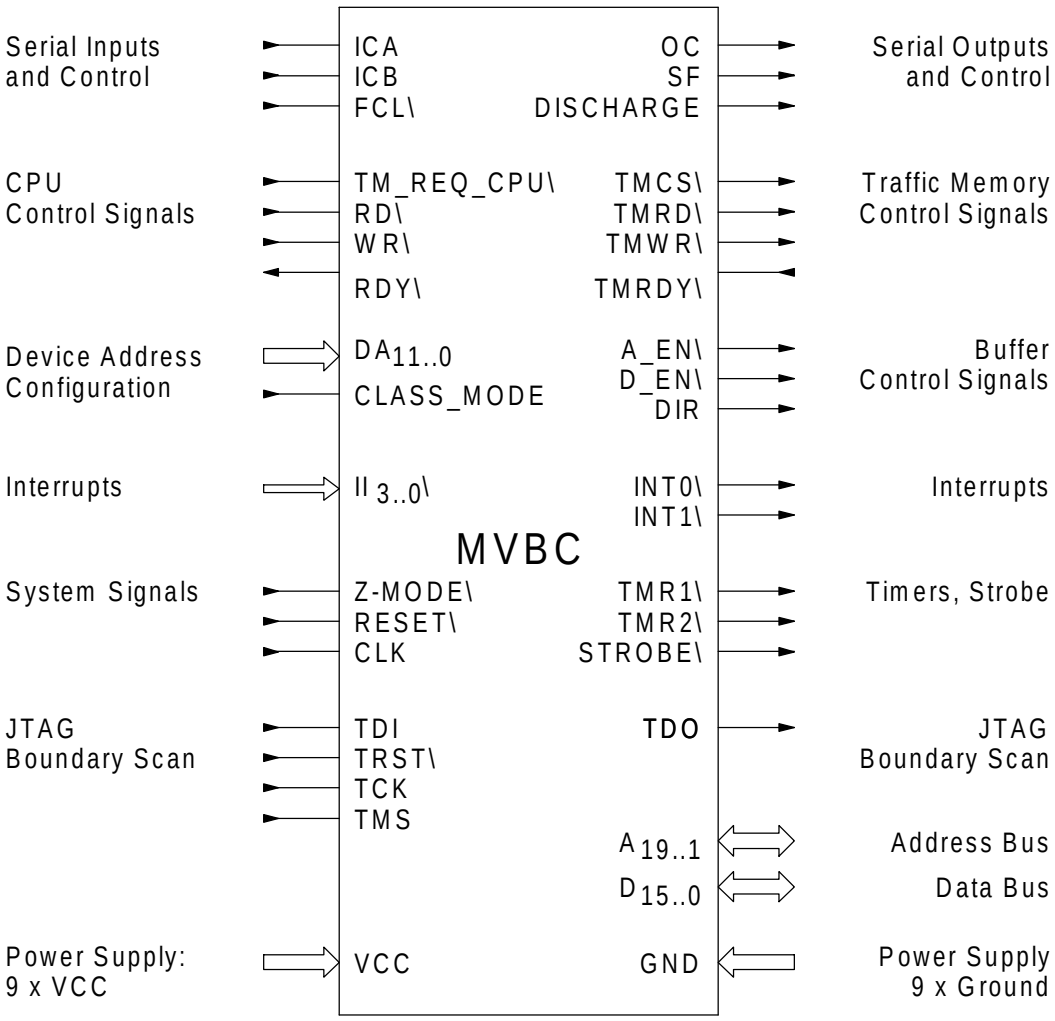


Figure 2.1: Device Interface Symbol

2.3 Systems Supported

The MVBC can either operate with a single host CPU or with assistance of a dedicated communication microprocessor to handle bus administration and upper level communication protocols. Class 1 mode allows autonomous operation without requiring a CPU or a microcontroller.

The functional characteristics of the MVBC can be modified via hardware configuration pins and via configuration registers that are set by software. The MVBC supports all device classes listed below:

- Class 1 Devices:
  - Simple slave boards without µP or µC (Actuators, Sensors)
  - No Traffic Memory
  - Slave boards with 8-bit or bigger µPs or µCs, with or without built-in I/O ports

- Class 2 Devices:
  - Slave boards with 16-bit or bigger µPs or µCs
  - Minimum 16 K Bytes Traffic Memory
  - 8-bit µPs or µCs can be attached using external logic

- Class 3 Devices:
  - See Class 2 Devices

- Class 4 Devices:
  - Boards with 16-bit µPs
  - Minimum 32 K Bytes Traffic Memory (Recommended: 256 K Bytes)
  - Capable to operate as a Bus Administrator

- Bus couplers, Gateways, etc.

2.4 MVBC versus BAP 15-2/3 with BAC

The Bus Access Processor (BAP 15-2/3) [2] was the very first MVB Controller designed by Brown Boveri (BBC). The Bus Arbitration Chip (BAC) [3] is a supplementary ASIC to handle automatic transmission of Master Frames and Telegram Analysis working together with the BAP 15-2/3.

2.4.1 Comparison Table

Traffic Memory (TM) and Addressing:

Function	**MVBC**	**BAP 15 2/3 with BAC**
TM and Data bus width	16 bits	8 bits
I/O Signal Levels	HCMOS (out), TTL (in)	TTL
Access arbitration to TM	Controlled by MVBC. MVBC can guarantee indivisible data transfers from/to TM. Realized with RDY\-Signal to add waitstates.	Simple DMA-like arbitration using HOLD/HLDA signals
Internal Registers	Accessible in a window inside Service Area in Traffic Memory space	Separate chip-select signal required
Traffic Memory (TM) Size	16 K, 32 K, 64 K, 256 K, 1 M Byte	8 K, 16 K Byte, expandable when using Bus Cycle Indicator (BZA: Buszyklusanzeige)
Maximum number of ports for logical-addressed Process Data	256 - 4096, depends on TM size	151 (8 KB TM) - 256 (16 KB TM)
Maximum number of ports for incoming device-addressed data	0 - 4095, depends on TM size	BAP alone: 0 BAP with BAC or external decoding logic to combine BZA: 256 ports.
Supervisory Frames (Function codes 8,9,12,13,14)	All frames are received in the Service Area inside the TM	Some frames must be explicitly retrieved from the internal Receive Register.
Location to place or retrieve Check Sequences (CRC) if requested	One word of the Port Control and Status Register (PCS)	Combination of address and BZA to create distinct addresses, i.e. by using BAC

Internal Features:

Function	**MVBC**	**BAP 15 2/3 with BAC**
Intermediate Buffer	Transmit and Receive Buffer, size: 16 x 16 bit words each	16-bit Transmit and Receive Registers
Data Transfer Interrupts	7	4 (2 of them combined to clear pending events)
External Interrupt Inputs	8	2
Bus Timeout (1.3 us), Silence (1.4 us)	Both supported by MVBC	Silence (1.4 us) is supported to switch incoming MVB line
Universal Timers	2 (selectable: 125 ns - 650 ms)	1 (selectable: 1 ms - 6.66 ms)
Bus Administrator Capabilities	Manual (1 frame at time) and automatic (up to 32 frames)	BAP alone: manual. Automatic method possible with BAC only.
Telegram Analysis	Inside MVBC	Requires BAC (error counter)
Bit width of TACK bits	16	8
Support for non-real-time host computers	Yes	No

Protocol Features:

Function	**MVBC**	**BAP 15 2/3 with BAC**
Automatic forcing of source and sink data	Supported	Not supported
Data comparison mechanism	Supported (interrupts on changing data)	Not supported
Message queues	Handled by MVBC (2 transmit queues and 1 receive queue)	Not supported
Sink-time supervision	Handled by MVBC	Requires external logic or must be handled by SW.
Event arbitration	Two Event Types: Event is signed off with CPE1 or CPE0 (Clear Pending Event)	Two Event Types: Event is signed off with interrupt IE1 or IE0.
Selectable Reply Timeout	21.3 / 42.7 / 64.0 / 85.4 us	33 / 37 / 41 / ... / 61 us (requires BAC)

Table 2.1: MVBC versus BAP/BAC

2.4.2 Compatibility: Frame Overlapping

If the skew between the frames on the incoming MVB lines A and B is too big, the BAP may transmit the next frame after the entire frame has arrived on line A, but not yet on line B. This results to frame overlapping. On the other hand, the MVBC will wait until the complete frame has been received on both lines.

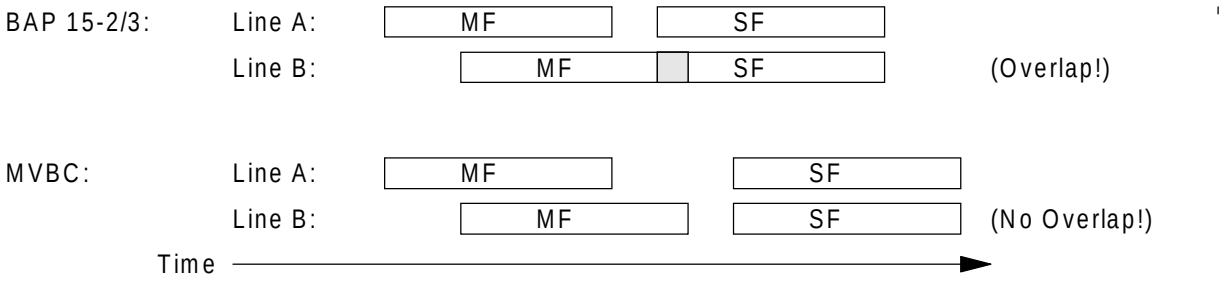


Figure 2.2: Frame Overlapping

2.4.3 Compatibility: Telegram Report

In the BAP15-2/3, the telegram status bits (TERR, STO, etc.) are handled accumulatively. When communication errors appear, then these bits are set to '1' but never reset to '0' after the next successful data transfer. In this case, software assistance is required to clear these bits. In the MVBC, the telegram status bits contain only up-to-date information relating to the last transferred frame.

2.4.4 Compatibility: TACK Bits at Message Transfers

The MVBC will no longer check the TACK bits before receiving messages (F-Code=12). The BAP15-2/3 does not receive any further messages while the most significant TACK bit is at '1'. The queue mechanism can be used on the MVBC to receive consistent messages instead.

(Space below has been left blank intentionally)

2.5 Input / Output Pins

2.5.1 Pinout Diagram

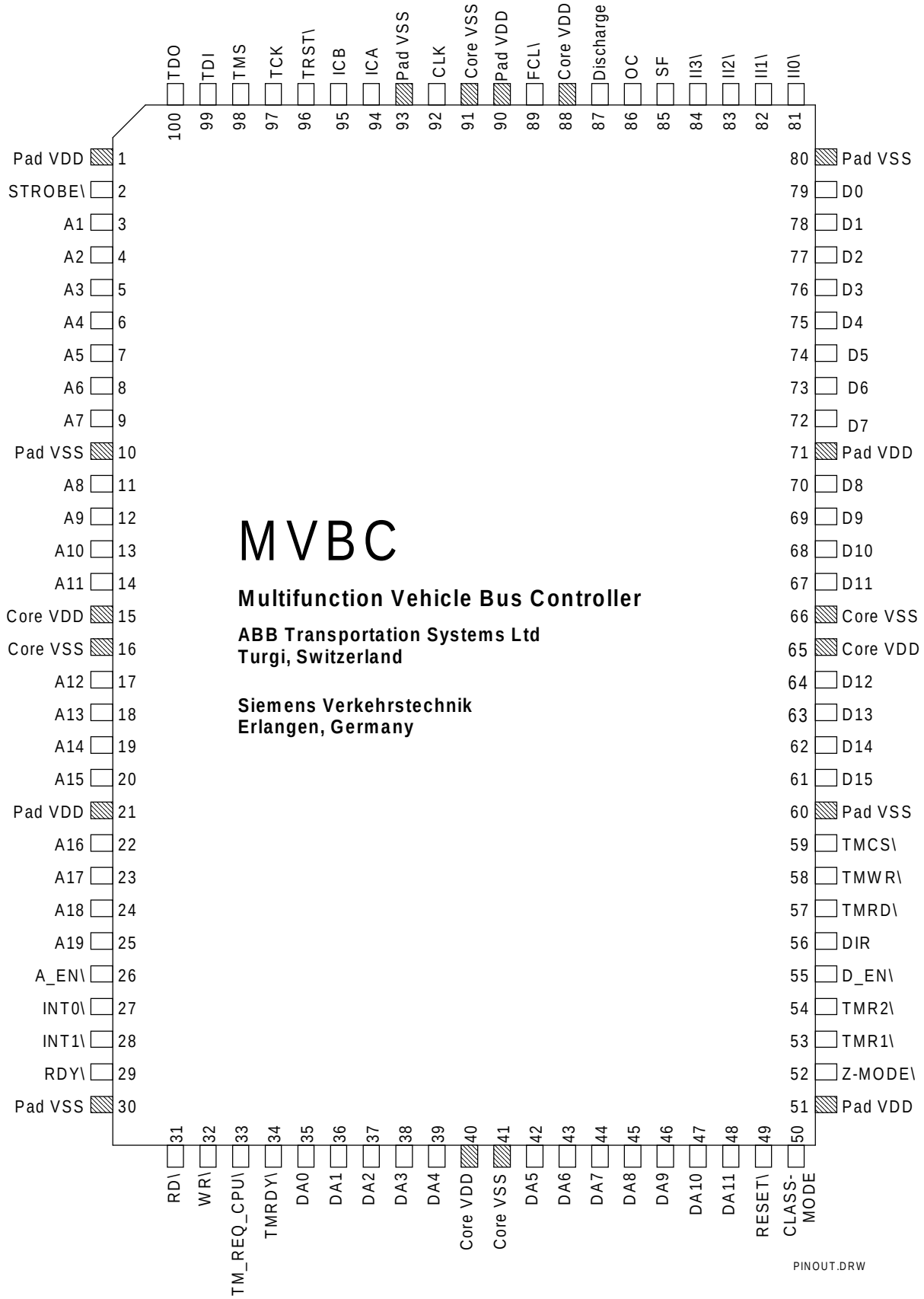


Figure 2.3: Pinout (Top View)

2.5.2 Pin Description

Pin number	Pin name / *Initial Level*	Dir. *Sense*	Description *Short summary*
61 - 79 [1]	$D_{15..0}$ *High Impedance*	I/O *Level*	Data Bus *The bidirectional lines are connected directly to the 16 data lines of the Traffic Memory and to the data transceivers which provide a link to the data bus of the host CPU.*
3 - 25 [1]	$A_{19..1}$ (see [2]) *High Impedance / High Level* [3]	I/O *Level*	Address Bus These lines address the Traffic Memory when the MVBC makes accesses. Also used as address inputs when CPU accesses MVBC or Traffic Memory.
35 - 48 [1]	$DA_{11..0}$ (see [2]) -	Input *Level*	Device Address Inputs *The Device Address input lines are directly connected the component that generates the Device Address bits (i.e. DIP switches). The DA must be valid during entire operation.*
31	RD\ -	Input *Level*	Read Signal from Host CPU *The CPU intends to perform a read-access from the TM or MVBC Internal Registers if both RD\ and TM_REQ_CPU\ are active.*
32	WR\ -	Input *Level*	Write Signal from Host CPU *The CPU intends to perform a write-access to the TM or MVBC Internal Registers if both WR\ and TM_REQ_CPU\ are active.*
33	TM_REQ_CPU\ -	Input *Level*	Traffic Memory or MVBC Request from Host CPU (Chip Select) *By activating this signal the CPU requests that it wishes to access the TM or registers*
29	RDY\ *Low*	Output -	MVBC Ready Signal to Host CPU *indicates that the data bus $D_{15..0}$ contains valid data.*
57	TMRD\ *High*	Output -	MVBC Read Signal to Traffic Memory *TMRD\ is used together with TMCS\ to perform read-accesses from the TM.*
58	TMWR\ *High*	Output -	MVBC Write Signal to Traffic Memory *TMRD\ is used together with TMCS\ to perform write-accesses to the TM.*
59	TMCS\ *High*	Output -	MVBC Chip-Select Signal to Traffic Memory *TMCS\ is used in combination with TMRD\ or TMWR\ to access the TM.*
34	TMRDY\ -	Input *Level*	Traffic Memory Ready Signal to MVBC *It is used to delay accesses made by/via MVBC to TM.*
26	A_EN\ *High*	Output -	Traffic Memory Address Buffer Enable *Enables the address buffer that is used to isolate the CPU from the Traffic Memory. Attention: A_EN may be at '0' if the MVBC is started up with TM_REQ_CPU_N='0'.*
55	D_EN\ *High*	Output -	Traffic Memory Data Transceiver Enable *Enables the data transceiver that is used to isolate the CPU from the TM.*
56	DIR *High*	Output -	CPU <-> MVBC/TM Data Flow Direction 0 = From CPU to MVBC/Traffic Memory 1 = From MVBC/Traffic Memory to CPU
2	STROBE\ *High*	Output -	Strobe Signal *Active for 3 clock cycles (125 ns) if a write access is made to the Synchro Port.*
92	CLK -	Input ↑ *Edge*	24 MHz Master Clock Input *Do not use any different frequencies since they change data rate and timeouts on the MVB.*
49	RESET\ -	Input *Level*	Asynchronous Reset *Initializes the MVBC to a known reset condition.*
50	CLASS_MODE\	Input *Level*	Specifies Class Mode the MVBC must operate in *0 = Class 1 Device 1 = Class 2/3/4 Device*
53, 54	TMR1,2\ *High*	Output -	Universal Timer 1 and 2 Outputs *Active for 3 clock cycle (125 ns) when respective counter reaches zero.*
81 - 84	$II_{0..3}$\ (see [2]) -	Input *See* [4]	External Interrupt Inputs *These inputs can be used to collect external interrupt sources and process them in the MVBC along with the internal interrupts.*

Continued on next page.

Continued from previous page.

Pin number	**Pin name / *Initial Level***	**Dir. *Sense***	**Description *Short summary***
27, 28	INT0..1\ *High*	Output -	Interrupt Outputs *Each Interrupt output cover up to 16 different interrupt sources.*
94	ICA -	Input *Level*	MVB Input Data Channel A *MVB format input data from physical layer MVB receiver*
95	ICB -	Input *Level*	MVB Input Data Channel B *MVB format input data from physical layer MVB receiver*
86	OC *Low*	Output -	MVB Output Data Channel *MVB format output data to physical layer transmitter*
87	(Reserved) *Low*	Output -	Reserved for future use.
85	SF *Low*	Output -	Send Frame *Indicates that a telegram is being sent via OC. This signal is available to enable physical layer drivers.*
89	FCL\ -	Input *Level*	Force Constant Light *Active FCL\ forces OC high. Used for fiber optical line strength calibration.*
52	Z-MODE\ -	Input *Level*	High-Impedance Mode *Allows all outputs to be switched to high Impedance to allow for easy in-circuit testing.*
96	TRST\ -	Input *Level*	Test Reset (JTAG Pin) *Must be active at power up to initialize JTAG Boundary Scan Hardware, similar as RESET\.*
97	TCK -	Input ↑ *Edge*	Test Clock (JTAG Pin)
98	TMS -	Input *Level*	Test Mode Select (JTAG Pin)
99	TDI -	Input *Level*	Test Data In (JTAG Pin)
100	TDO *High*	Output -	Test Data Out (JTAG Pin)
15, 40, 65, 88	VDD (Core) -	Power -	4 x +5 V Pins for core
1, 21, 51, 71, 90	VDD (Pads) -	Power -	5 x +5 V Pins for pads
16, 41, 66, 91	VSS (Core)	Power -	4 x Ground Pins for core
10, 30, 60, 80, 93	VSS (Pads)	Power -	5 x Ground Pins for pads

[1] These pin number range specifies where these signals can be found. Other pins (i.e. control signals, VDD, VSS) may lie inbetween. Check the pinout diagram for detailed pin assignment information.

[2] Different signal assignments apply when the MVBC operates in Class 1 Mode (CLASS_MODE input is at '0'). See section 2.9.14.4 for details.

[3] Wherever two initial values are given and separated with the slash character '/', the left value denotes to Class 2/3/4 Mode, the right value to Class 1 Mode.

[4] The interrupt is triggered after the transition from '1' to '0'. Before the transition, the signal must have been at '1' for at least 1 clock cycle (41.7 ns). After the transition, the signal must remain stable for at least one clock cycle again.

Table 2.2: Pin Description

2.6 Physical Layer Interface to the MVB

The physical layer interface suits to both copper twisted pair medium and fiber optical medium. The MVBC provides one common output (OC) and two redundant inputs (ICA, ICB). The transmitted and received signals are active high:

High = +5 V
Low = 0 V

The SF-signal must be used to enable the drivers. This signal is activated 125 ns before the beginning and 125 ns after the end of every frame. This time overhead assures correct data transmissions by prevent undesired transients at the frame boundaries.

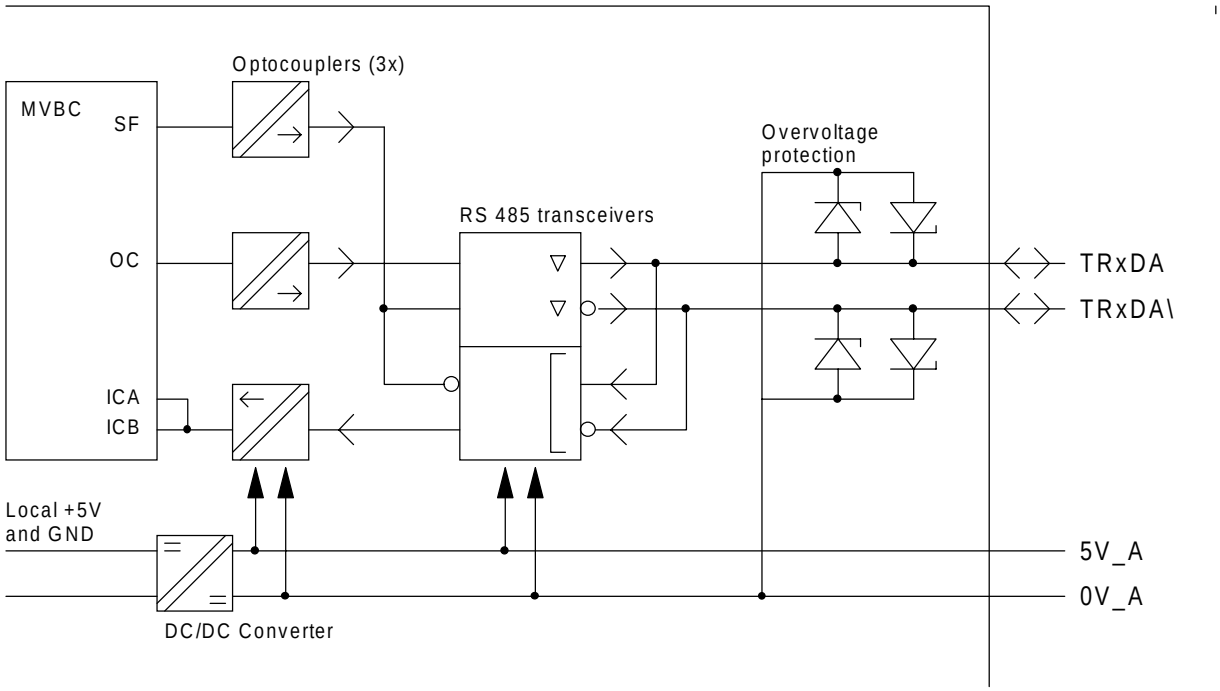


Figure 2.4: Physical layer interface to copper medium

The following timing diagram illustrates the behavior of the SF signal:

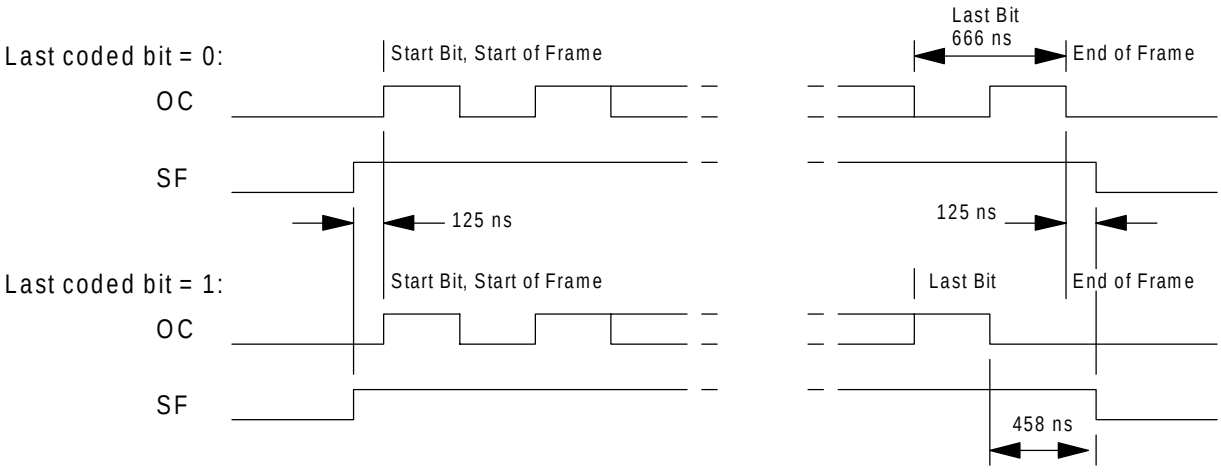


Figure 2.5: SF-Signal Behavior

2.7 Physical Interface to the Traffic Memory and CPU

2.7.1 Physical Interconnection to Traffic Memory

The TM access control signals are organized in order to connect static RAM chips directly. Support logic is necessary when more complex memory structures (i.e. DRAMs) are chosen.

The TMRDY\-Signal must be tied to '0' if the Traffic Memory can return data within a definite time period. In this case, the number of waitstates must be configured inside the MVBC.

Buffers and bi-directional transceivers must be used to isolate the MVBC address and data buses from the main CPU address and data buses. The following figure illustrate the basic interconnection between the MVBC, TM and host CPU.

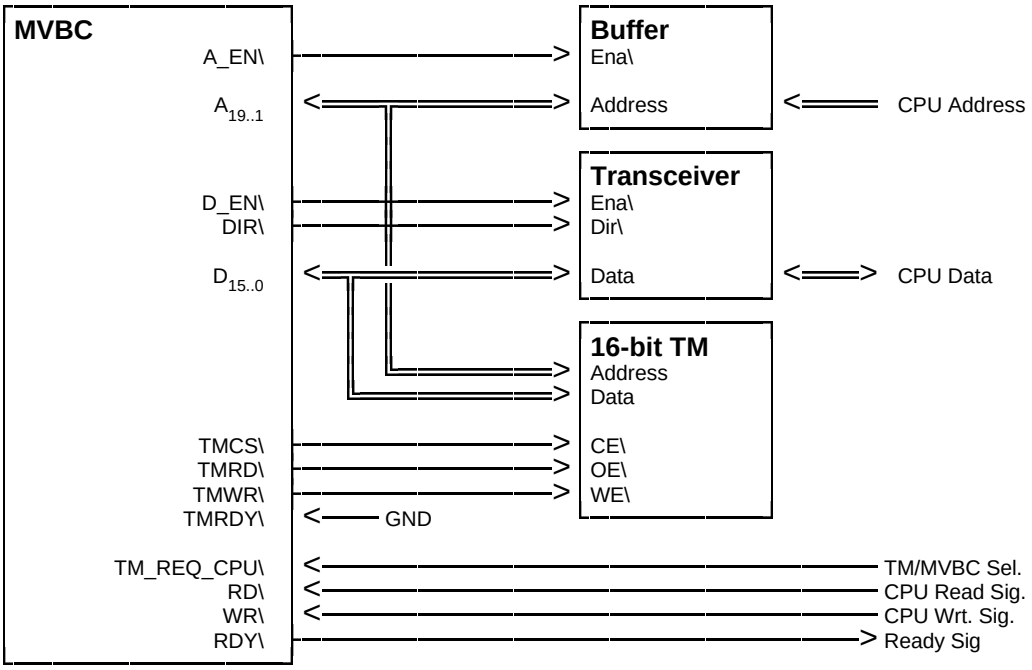


Figure 2.6: Simple CPU / MVBC / TM Interconnection Diagram

2.7.2 Physical Interface to host CPU

The MVBC handles all host CPU accesses made to the Traffic Memory or Internal Registers in the MVBC. The CPU control signals (TM_REQ_CPU\, RD\, WR\) are connected directly to the MVBC. The MVBC takes care of arbitration if both CPU and MVBC intend to access the Traffic Memory simultaneously.

In addition, the two interrupt request signals (INT0\, INT1\) must be connected directly or via an interrupt controller to the CPU.

2.7.3 Word and Byte Access Restrictions

The MVBC supports 16-bit Traffic Memory architectures. No explicitly 8-bit accesses will be performed. However, custom glue logic can be introduced to support 8-bit read and write accesses.

2.7.4 Byte Order

The Byte Order issue must be considered carefully. Consider following cases: Motorola (Big Endian format) and Intel (Little Endian format). The left column in the following figure describes non-numeric data (character strings, stored in memory with incrementing address order). The right column describes numeric data (upper halfword contains obviously the more significant digits) words. The processors write the data (and as well the Force Mask and Force Data) into the Traffic Memory using their native Endian formats:

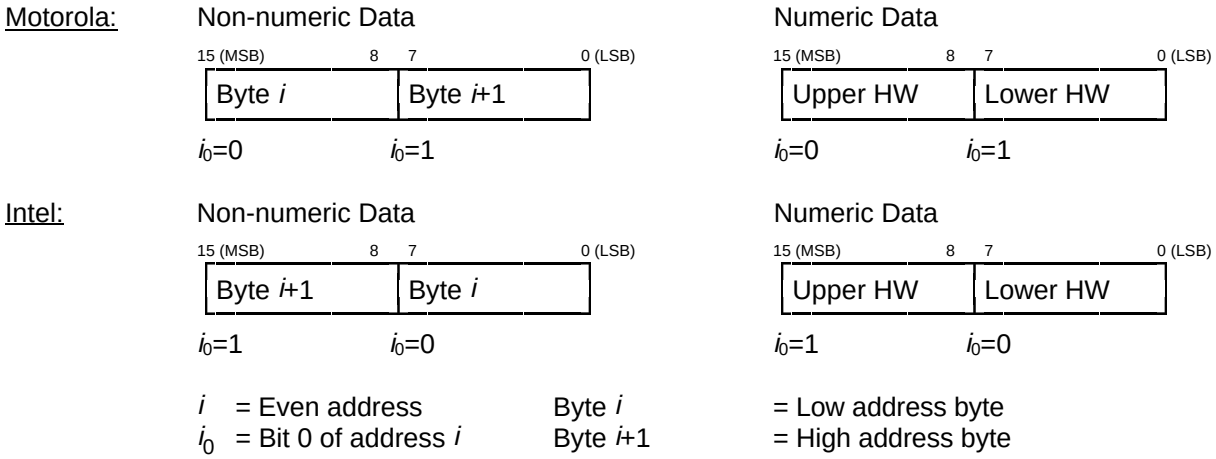


Figure 2.7: Data Format inside Traffic Memory

The MVBC will load these words from Traffic Memory (or internal Receive Buffer) to the internal Transmit Buffer (or Traffic Memory) without performing any byte swaps. However, the following order is required when transferring numeric and non-numeric data over the MVB:

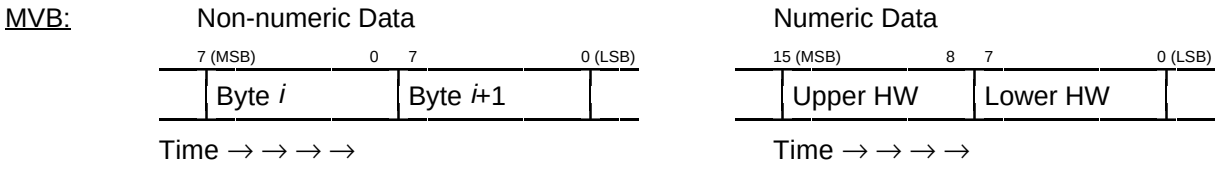


Figure 2.8: Data Format on the MVB

Since the MVBC transmits 16-bit words starting with the most significant bit first, no problems arise when transmitting numeric data. In addition, no problem should occur when transmitting non-numeric data if the host processor uses Big Endian format (like the Motorola 680x0 series). A byte swap is necessary when transmitting non-numeric data while the host processor operates with Little Endian format (like the Intel 80x86 series).

Approach: The MVBC shall be informed whether to perform a byte swap or not. First, an Intel/Motorola bit (IM in SCR, see section 2.9.7) is available to enable byte swapping. If the MVBC operates in "Intel" mode, byte swapping is permitted (does not mean "enabled"). Otherwise, "Motorola" allows no swapping at all. Byte swapping is performed only if the data to be transferred is declared as "non-numeric data" (See PCS, section 2.8.3.1).

In the normal case, all data being transferred are considered as numeric data except messages (see section 3.7). Word order for 32-bit or bigger numbers is not handled by the MVBC. The software shall assure that the upper 16-bit word is stored at address *i* and the lower 16-bit word at address *i*+2 (*i* = an even address).

Attention: In class '1' mode, all data being transferred is considered as numeric data. No byte swapping will take place at all. If 8-bit data are exchanged (i.e. between MVBC and an A/D-converter), the user shall make a decision himself whether to use the upper or lower eight bits and declare the respective data as numeric or non-numeric in MVBCs of the other bus participants.

2.7.5 Class 1 Mode Addressing

While the MVBC operates in Class 1 Mode, the addressing range is limited to sixteen 16-bit locations. Sixteen

bits ($A_{15..1}$ and A_{16}) of the address bus are decoded into active-low *chip-select* signals. The remaining three bits will stay at '1' permanently. Details about operation in Class 1 Mode are given in section 2.9.14.

2.8 Software Interface

All information and data pertaining to the MVBC are found in the Traffic Memory address space. This space is visible to both host CPU and MVBC. The Traffic Memory is divided into following partitions:

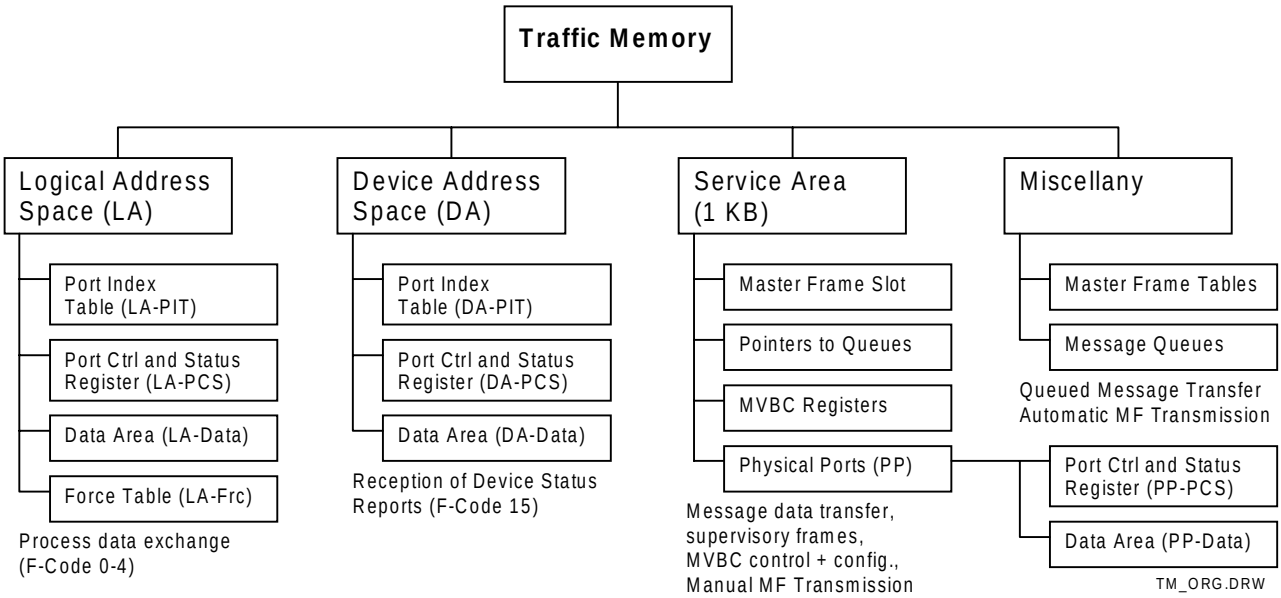


Figure 2.9: TM Organization

Depending on the hardware application, the user may consider to choose a Traffic Memory with a specific size. In order to utilize the Traffic Memory efficiently, five Memory Configuration Modes (MCM) have been introduced.

MCM	TM Size	LA Ports	DA Ports	Unused	Service Area (address range)
Class 1	-	16	0	-	No Service Area
0	16 KB	256	0	1 KB	03C00H - 0x03FFFH
1	32 KB	256	256	7 KB	07C00H - 0x07FFFH
2	64 KB	1024	256	1 KB	0FC00H - 0x0FFFFH
3	256 KB	4096	2048	31 KB	0FC00H - 0x0FFFFH
4	1 M	4096	4096	751 KB	0FC00H - 0x0FFFFH
4 (see [1])	256 KB	4096	3008	8 KB	0FC00H - 0x0FFFFH

[1] MCM=4 operating with 256 KB TM: The DA Data Area at location 40000H-50000H maps into 00000H-100000H. In order to avoid overlapping into the Port Index Tables and Service Area, the range is limited to 47 KB (04000H-0FC00H). Consequently, only 3008 DA ports can be supported. The vacant 8 KB region originates from the PCS space due to 1088 unused ports.

Table 2.3: Memory Configuration Modes (MCM)

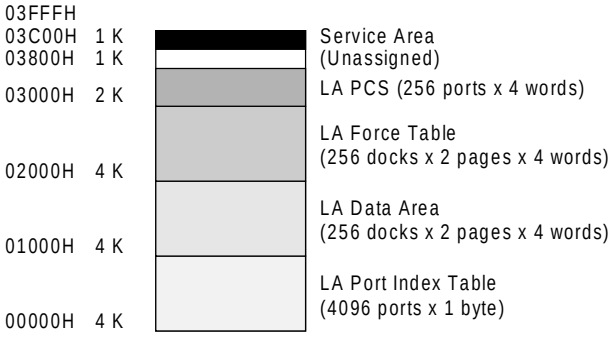
2.8.1 Traffic Memory Maps

The following pages illustrate the Traffic Memory Maps for all Memory Configuration Modes. All *unused* memory fragments can be used for data structures which are not bound to any specific location: Message Queues and

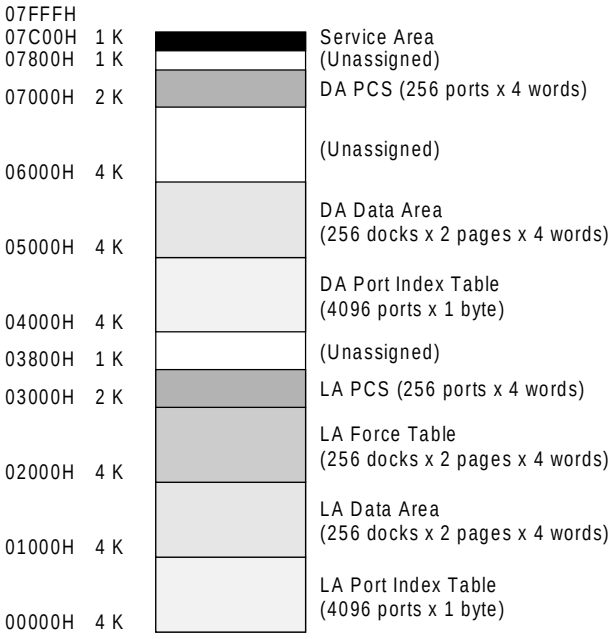
Master Frame Tables.

Mode 0 is effective after power-up or an asynchronous or synchronous reset has been issued, given the MVBC does not operate in class 1 mode. The start address to the Service Area is 03C00H. Mode 0 supports no device-addressable ports which are necessary to receive Device Status Reports.

Memory Configuration Mode 0 (16 KB)



Memory Configuration Mode 1 (32 KB)



Memory Configuration Mode 2 (64 KB)

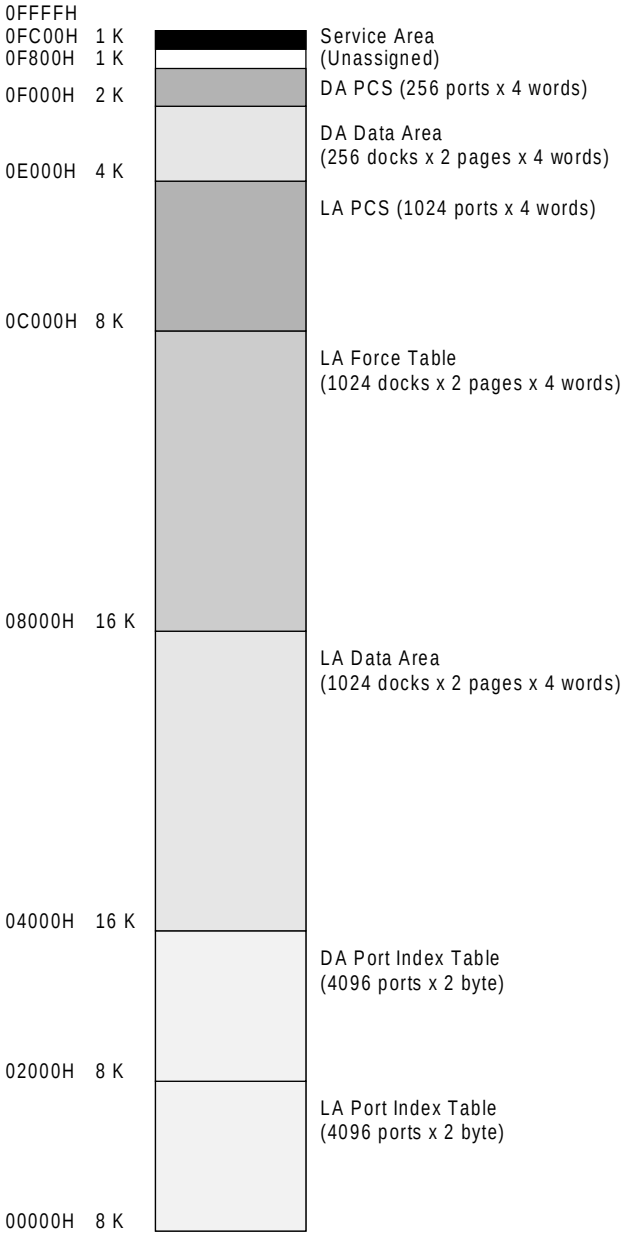
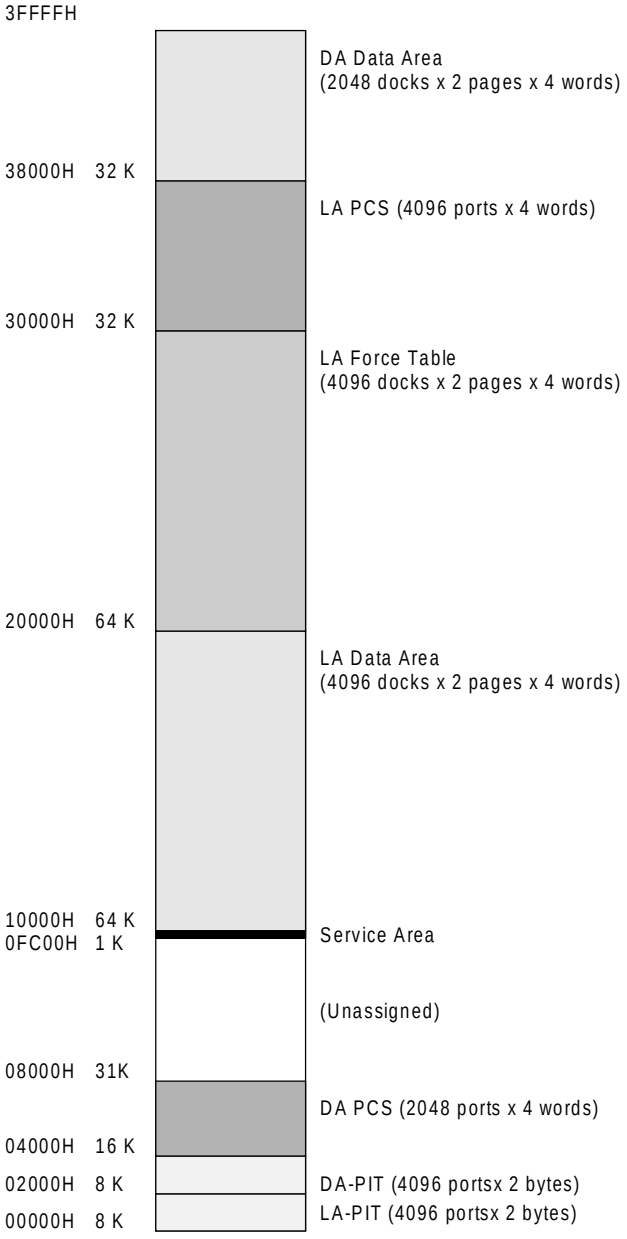


Table 2.4: Traffic Memory Maps

Memory Configuration Mode 3 (256 KB)



Memory Configuration Mode 4 (1 MB)

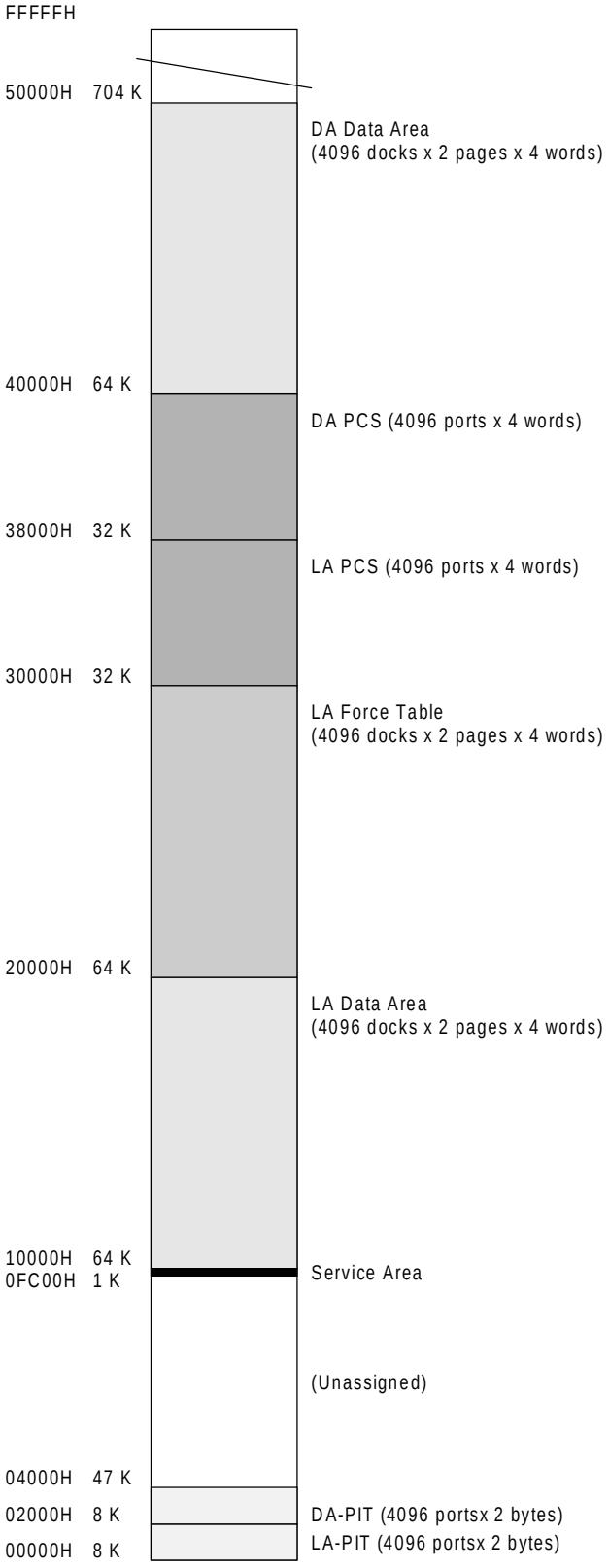


Table 2.4: Traffic Memory Maps (continued)

2.8.2 Port Index Tables (PIT)

The Traffic Memory provides two Port Index Tables, one for

- Logical Addressed Data (Used for Process Data)
- Device Addressed Data (Used to receive Device Status Reports)

The Port Index Tables link the Ports to the Logical and Device Addresses which are specified in the Master Frames. Port Index PI=000_{hex} is used to assign all unused Logical and Device Addresses.

If the Memory Configuration Mode (MCM) is zero, then no device addressable Port Index Table and Ports exists. If MCM={0,1}, every 16-bit word in the Port Index Table contains two 8-bit Port Indexes. The low (high) address bytes contain even (odd) numbered Port Indexes. Since the native byte order of the host CPU is used, the MVBC must be informed about the selected byte order (see SCR, IM-bit, section 2.9.7). If MCM={2,3,4}, then every 16-bit word contains one 12-bit Port Index, covering bits 11..0. Bits 15..12 must be zero.

Size:	1 Word / Port (if MCM · 2), else ½ Word / Port
Start Address:	LA-PCS: 00000H
	DA-PCS: 04000H (if MCM = 1) or 02000H (if MCM · 2)

PIT Word Format for MCM ≤ 1:

Motorola Mode:

15 … 8	7 … 0
Port 4094 :	Port 4095 :
Port 0	Port 1

Intel Mode:

15 … 8	7 … 0
Port 4095 :	Port 4094 :
Port 1	Port 0

Figure 2.10: PIT Organization

2.8.2.1 Address Evaluation from Port Index

The Port Index is used to compute the effective TM addresses to the following memory blocks:

- PCS
- Data Area
- Force Table (For logical address space only)

Figure 2.11 on the next page shows the algorithm which is used to compute the effective TM addresses.

Address Evaluation Example:

Consider an MF containing 0234H (F-Code=0, Port Address = 234H). The MCM is set to 2 (64 KB Traffic Memory). First, the TM address 0468EH is evaluated from the Port Address to read the Port Index Table (PIT). Assume, the Port Index contains 00FH. The resulting addresses are computed as follows:

Addr(PCS) = Start_addr(PCS) + (00F shl 3) = 0C000 + 078 = 0C078

Map(PI=00F, VP=0) = 00D8H; Map(PI=00F, VP=1) = 00F8H

Addr(Data_Area, VP=0) = 04000H + 00D8H = 040D8H
Addr(Data_Area, VP=1) = 04000H + 00F8H = 040F8H
Addr(Force_Table, Data pattern: VP=0) = 080D8H
Addr(Force_Table, Mask pattern: VP=1) = 080F8H

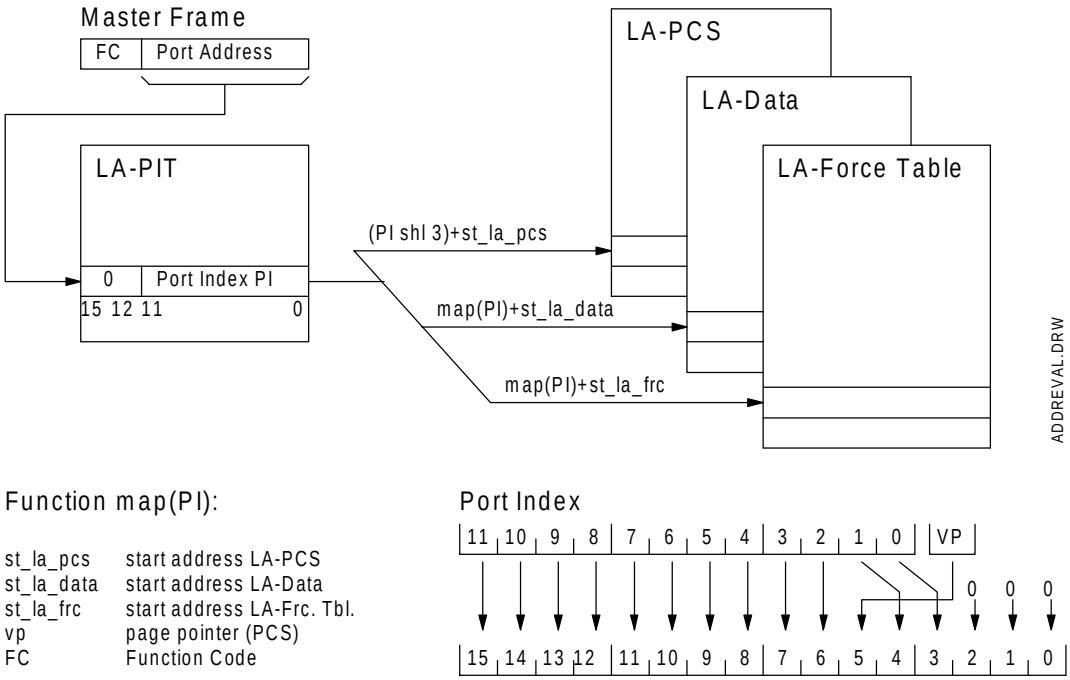


Figure 2.11: Address Evaluation from Port Index

2.8.3 Port Control and Status Register (PCS)

The Port Control and Status Register contains all the relevant information pertaining to one port. This information is used by MVBC to determine how it should handle the related port. The PCS includes following information:

- Port-related information
  - Function Code
  - Port description (source vs sink port, queuing and forcing en-/disabled)
  - Specifications of interrupts to occur when transfer has completed
  - Event arbitration
- Data consistency check
  - Valid Page pointer
  - Port Disabling mechanism (used to support host systems which do not comply with any real-time requirements)
- Telegram report
  - Indicates type of communication error or timeout occurred
- Transfer acknowledge bits
  - Acknowledges successful data transfer
  - Intended for sink-time supervision
- Check Sequences
  - Used if data is transferred with software-defined Check Sequences

Each PCS is a four-word record. The bit description is summarized next:

PCS Word 0: Port Description
PCS Word 1: Telegram Report, Page Pointer, Disabling Mechanism
PCS Word 2: Transfer Acknowledge Bits
PCS Word 3: Check Sequences

Symbols in	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Word 0	$F\text{-}Code_{3..0}$				SRC	SINK	TWCS	WA	$IE_{2..0}$			$CPE_{1..0}$		QA	NUM	FE
Word 1	$DEC_{7..0}$								PTD	VP	CRC	SQE	ALO	BNI	TERR	STO
Word 2	$TACK_{15..0}$															
Word 3	$CS1_{7..0}$								$CS0_{7..0}$							

2.8.3.1 PCS Word 0: Port Description

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$F\text{-}Code_{3..0}$				SRC	SINK	TWCS	WA	$IE_{2..0}$			$CPE_{1..0}$		QA	NUM	FE
Init. Value:	This location in the Traffic Memory must be initialized by the host software															
CPU Acc.:	Read and write access															
MVBC Acc.:	Read-only															

The MVBC accesses this port every time after a Master Frame has been received and an appropriate port has been selected.

Symbol	Description
$F\text{-}Code_{3..0}$	Function Code (F-Code) The MVBC compares the F-Code in the PCS with the F-Code received in the Master Frame. If they are identical, then further actions are taken. Otherwise, the frame is ignored and no further action is taken. Description of F-Codes: See appendix A Exception: The F-Code is not checked during event arbitration. Reason: The ports EF0, EF1 and EFS (See appendix A) must be accessible with three different F-Codes: 9, 13 and 14. A value of 9 is suggested.
SRC	Port is active Source 0 Port is passive source 1 Port is active source (regardless if SINK=0 or 1)
SINK	Port is active Sink 0 Port is passive sink 1 Port is active sink (provided that SRC=0)
TWCS	TWCS Transfer with Check Sequence 1 Transfers data with 8-bit Check Sequence between the Traffic Memory and the MVBC. The Check Sequence is loaded/stored in word 3 of the PCS. For transferring queued Message Data: The Check Sequences are not queued. They are handled in the same manner as for any regular data transfer.
WA	Write Always 0 Received Slave Frames with errors will not be written into the Data Area of the TM (or appended to the Message Queue if F-Code=12). VP remains unaffected. This mechanism prevents sink ports from becoming contaminated with erroneous data. 1 All correct Slave Frames, as well as erroneous Slave Frames where no frame size discrepancy has been detected will be stored in the Traffic Memory. This type of erroneous data is known as recoverable data. Erroneous data with mismatching frame size (i.e. due to an signal loss) will not be stored. This feature may be useful for bridges which do intentionally forward erroneous frames (using false CRCs with TWCS=1).

Continued on next page.

Continued from previous page.

Symbol	Description
$IE_{2..0}$	Enable Interrupts A general-purpose Data Transfer Interrupt (DTIi) is generated upon successful (no errors, or recoverable data after an error given WA-bit is set) processing of the port. If an error is detected, then no interrupt is asserted because no data transaction is made to the port. The software developer is free to use these interrupts for his application needs. 0 0 0 No interrupt is generated 0 0 1 DTI1 is asserted (see also section 2.9.15.2) 0 1 0 DTI2 is asserted 0 1 1 DTI3 is asserted 1 0 0 DTI4 is asserted 1 0 1 DTI5 is asserted 1 1 0 DTI6 is asserted 1 1 1 DTI7 is asserted (If DTI7 is enabled and the port is a sink, then an Automatic Comparison Mechanism is enabled. No interrupt occurs if the port is configured as a source. See section 3.9.4 for details.)
$CPE_{1..0}$	Clear Pending Event of Type 1, or 0 1 The announced event of type ET = 1 or 0 is cleared upon any data transfer using this port by setting EAi and PARi (see SCR, section 2.9.7) to zero. These bits are used for event arbitration and event data transfers, regardless if the transfer turned out successfully or not. Exception: If the QA-bit (see below) is active, then EAi will be cleared only if no more messages are left in the queue.
QA	Queue Attached to Port 1 Message Queues are attached. If SINK=1, the Receive Queue is attached. If SRC=1 and SINK=0, the Transmit Queues are attached. Transmit Queue 0 will be checked before Transmit Queue 1. 0 No queue is attached. All data transfers are made with Data Area (inside Traffic Memory) instead. This bit applies for message transfers only (F-Code=12). However, the user is required to keep this bit at zero in PCS with other F-Codes.
NUM	NUM Numeric Data 1 The port contains 16-bit or longer numeric data. This information is required in order to transmit numeric data in the correct byte sequence (see section 2.7.4). 0 The port contains non-numeric data (i.e. character sequences, messages) NUM has no effect if the MVBC is running in Motorola-Mode (see section 2.7.4), but the user is strongly recommended to declare numeric data anyway. Reason: The byte ordering is correct for both numeric and non-numeric data. Attention: The MVBC handles byte order of 16-bit words only. It does not handle word order of 32-bit or larger words.
FE	Data Forcing Enabled 0 Data Forcing is disabled. The MVBC does not access the Force Table. 1 Forcing is enabled. Access to Force Mask and Data Area are made before transmission or reception of every Process Data word. Attention: Forcing requires two additional TM accesses for each data word.

Table 2.5: PCS Word 0: Port Description

2.8.3.2 PCS Word 1: Telegram Report, Page Pointer

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$DEC_{7..0}$								PTD	VP	CRC	SQE	ALO	BNI	TERR	STO
Init. Value:	This location in the Traffic Memory must be initialized by the host software															
CPU Acc.:	Read and write access															
MVBC Acc.:	Read-only								rw	rw	w	w	w	w	w	w

This word is updated every time a data transfer has taken place (both successful or erroneous), or a Reply Timeout has occurred (default value: 42.7 µs). This word is not updated if a Master Frame has been received while expecting a Slave Frame, or if the port is inactive. The six error bits CRC through STO are not cumulative. If the next data transfer was successful, then all of these bits return to '0'.

Symbol	Description
$DEC_{7..0}$	Disable / Enable Counter. See 3.9.2. This counter is used in association with the PTD (Port Temporarily Disabled) bit to disable data reception in order to allow host computers, which do not fulfill real-time requirements, to retrieve data from this port. = 0 The port remains active according to the information's given in PCS Word 0. Whenever the MVBC writes PCS Word 1 back to the Traffic Memory, PTD will be set to '0' if this is not yet done so. > 0 The port will disable automatically after the next transfer. Whenever the MVBC writes PCS Word 2 back to the Traffic Memory, PTD will be set to '1'.
PTD	Port Temporarily Disabled. See 3.9.2 0 The port is enabled. The MVBC will set PTD to '0' if it detects a zero Disable/Enable Counter. 1 The port is disabled Attention: PTD applies to sink ports (SINK=1, SRC=0) only. It is ignored when the port is used as a source.
VP	Valid Page Pointer The Page Pointer indicates which page of the Traffic Memory data section is active. This pointer is used to maintain data consistency. If the port is active source (SRC='1', SINK=don't care): The MVBC reads data from the page pointed by VP. The MVBC does not change the value of VP. The host must write data to the page pointed by Inv(VP). After data has been written, the host must invert VP in order to make new data visible to the MVBC. If the port is active sink (SRC='0', SINK='1'): The host reads data from the page pointed by VP. The host must not change the value of VP. The MVBC writes all incoming data to Inv(VP). After all words of the telegram have been written, the MVB inverts VP in order to make the new data visible to the host. Attention: VP is not inverted if erroneous data has been received and the MVBC will therefore not store the received data. Exceptions apply when the WA-bit is active. See WA-bit in PCS Word 0.
CRC	Mismatching Check Sequence 1 Frame with mismatching CRC has been detected. If WA=1, and the port is a sink, then the erroneous frame is written to the Traffic Memory. If this bit is active, then TERR will also be active.

Continued on next page.

Continued from previous page.

Symbol	Description
SQE	Signal Quality Error 1 Bad signal quality has been detected. This bit is enabled by the Decoder Unit (see 2.9.4) if the double-sampling on the incoming line gives different results. If this bit is active, then TERR will also be active.
ALO	Active Level Overbalance, commonly found when collisions have occurred. 1 This signal is active if following two cases are met: ▪ A garbled frame with valid delimiter has been detected (probably a weak overlapping collision) Here: STO = 0, TERR = 1 ▪ A garbled frame with garbled delimiter has been detected (probably a strong overlapping collision) Here: STO = 1, TERR = 0 If this bit is active, then TERR will also be active.
BNI	Bus not Idle 1 The Bus Not Idle Bit will go active when after receiving a Master Frame a pulse of minimum 83 nsec (2 clock cycles à 24 MHz) is seen on the active MVB transmission line. BNI is affected when port is active source or active sink. Exception: BNI is not set to '1' after successful data transfers. If this bit is active, then TERR and/or STO will also be active.
TERR	Telegram Error Bit 1 Frame error has been detected. This bit can only become active if a valid Master or Slave Delimiter has been received and missing or garbled data follows. The following error types can set this bit to '1': ▪ Mismatching CRC (CRC bit will also be active) ▪ Bad Manchester bit ▪ Signal quality error (SQE bit will also be active) ▪ Frame length error ▪ Active Level Overbalance (ALO bit will also be active) TERR is affected when port is active source or active sink.
STO	Slave Frame Reply Timeout If no valid Slave Delimiter (SD) is received within reply time (default value: 42.7 µs) of receiving a valid Master Delimiter (MD), the timeout bit is set.

Table 2.6: PCS Word 1: Telegram Status, Page Pointer

2.8.3.3 PCS Word 2: Transfer Acknowledge Bits

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$TACK_{15..0}$															
Init. Value:	This location in the Traffic Memory must be initialized by the host software															
CPU Acc.:	Read and write access															
MVBC Acc.:	Read-only (if sink-time supervision disabled) or read/write access (if enabled)															

This word is updated every time data has been transferred from the MVBC to Traffic Memory or vice versa.

Symbol	Description
$TACK_{15..0}$	Transfer Acknowledge The TACK bits will be set after a valid transfer from the MVBC to the TM or in the other direction. The complete word is set to FFFFH when this transfer is complete. The software or the Sink-Time Supervision Logic on the MVBC (see section 2.9.17) can decrement this counter in order to perform sink time supervision. TACK is not affected when the port is inactive.

Table 2.7: PCS Word 2: TACK Bits

2.8.3.4 PCS Word 3: Check Sequences

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$CS1_{7..0}$								$CS0_{7..0}$							
Init. Value:	This location in the Traffic Memory must be initialized by the host software															
CPU Acc.:	Read and write access															
MVBC Acc.:	Read-only (source ports) or write access (sink ports)															

Symbol	Description
$CS0_{7..0}$, $CS1_{7..0}$	8-bit Check Sequence for data page 0 (CS0) and for page 1 (CS1) If TWCS (Transfer with Check Sequence) equals '1', then the Check Sequence (CS) corresponding to the data stored in pages 0 and 1 located in the Data Area are stored in CS0 and CS1 respectively. The page is pointed by VP. If the frame size exceeds 4 words so that multiple Check Sequence are required (up to 4), the other Check Sequence pairs are found in the corresponding PCS word 3 of the next higher (unused) port addresses. The following examples illustrates CS-storage for a 16-word PD variable:

Dock Nr.	Addr.	Contents	Usage
	i+1EH	PCS Word 3	CRC's for words 12-15
	i+1CH	PCS Word 2	(not used)
	i+1AH	PCS Word 1	(not used)
Dock 3	i+18H	PCS Word 0	(not used)
	i+16H	PCS Word 3	CRC's for words 8-11
	i+14H	PCS Word 2	(not used)
	i+12H	PCS Word 1	(not used)
Dock 2	i+10H	PCS Word 0	(not used)
	i+0EH	PCS Word 3	CRC's for words 4-7
	i+0CH	PCS Word 2	(not used)
	i+0AH	PCS Word 1	(not used)
Dock 1	i+08H	PCS Word 0	(not used)
	i+06H	PCS Word 3	CRC's for words 0-3
	i+04H	PCS Word 2	TACK Bits
	i+02H	PCS Word 1	Telegram Status, VP, etc
Dock 0	i+00H	PCS Word 0	Port Description

Table 2.8: PCS Word 3: Check Sequences

2.8.4 Data Areas

The Data Areas contain the data to be transmitted to or received from the MVB. Data Areas are available for logical addressed ports, device addressed ports and Physical Ports (located inside Service Area). The Data Area provides memory space for a specified number of docks. 2 pages of 4 words are assigned to each dock. For 1, 2 and 4-word telegrams, one dock is assigned to one port. For 8 and 16-word telegrams, two (or four) docks are grouped into one port. The page is pointed by the VP bit in the PCS.

Address	1 / 2 / 4 words: 1 Dock / Port	8 words: 3 Docks / Port	16 words: 4 Docks / Port
+38H ... +3FH	Port i+3, Page 1	(Dock 3)	(Dock 3)
+30H ... +37H	Port i+2, Page 1	Port i+2, Page 1	(Dock 2)
+28H ... +2FH	Port i+1, Page 1	(Dock 1)	(Dock 1)
+20H ... +27H	Port i+0, Page 1	Port i+0, Page 1	Port i+0, Page 1
+18H ... +1FH	Port i+3, Page 0	(Dock 3)	(Dock 3)
+10H ... +17H	Port i+2, Page 0	Port i+2, Page 0	(Dock 2)
+08H ... +0FH	Port i+1, Page 0	(Dock 1)	(Dock 1)
+00H ... +07H	Port i+0, Page 0	Port i+0, Page 0	Port i+0, Page 0

Table 2.9: TM Data Area

2.8.5 Force Table

The Force Table is used to override transmitted or received Process Data with forced data. The Force Table consists of two parts: Force Data bit pattern (FD) and Force Mask bit pattern (FM). The Force Data Pattern contains the data image (similar format as in a single page inside the Data Area entry) which will be partly or entirely used when forcing is enabled (See FE-bit inside PCS).

Attention: Forcing is not allowed on Message Data and Supervisory Data.

For every active Force Mask Bit ('1'), the corresponding bit to be transmitted or received is obtained from the local Force Data word. For every passive Force Mask Bit ('0'), the corresponding bit to be transmitted is obtained from the selected page (pointed by VP in PCS) in the Data Area. For data reception: If the corresponding Force Mask Bit is '0', then the data bit is obtained from the MVB.

Data-to-xmit = (DT[VP] & not FM) or (FD & FM)
Data-to-rcve = (MVB & not FM) or (FD & FM)

DT Data Area, pointed by Page Pointer VP

MVB Current word retrieved from MVB

The organization of the Force Table is similar to that of the Data Area. 4 words are assigned to each dock. For 1, 2 and 4-word telegrams, one dock is assigned to one port. For 8 and 16-word telegrams, two (or four) docks are grouped into one port. The page is pointed by the VP bit in the PCS. See table on next page.

Address	1 / 2 / 4 words: 1 Dock / Port	8 words: 3 Docks / Port	16 words: 4 Docks / Port
+38H ... +3FH	Port i+3, Mask	(Dock 3)	(Dock 3)
+30H ... +37H	Port i+2, Mask	Port i+2, Mask	(Dock 2)
+28H ... +2FH	Port i+1, Mask	(Dock 1)	(Dock 1)
+20H ... +27H	Port i+0, Mask	Port i+0, Mask	Port i+0, Mask
+18H ... +1FH	Port i+3, Data	(Dock 3)	(Dock 3)
+10H ... +17H	Port i+2, Data	Port i+2, Data	(Dock 2)
+08H ... +0FH	Port i+1, Data	(Dock 1)	(Dock 1)
+00H ... +07H	Port i+0, Data	Port i+0, Data	Port i+0, Data

Table 2.10: TM Force Table

Attention: Activating TWCS (in PCS Word 0) and FE when the port is used as a sink makes no sense. In this case, the received Check Sequence will no longer match with the data written into Traffic Memory.

2.8.6 Service Area

Depending on the selected Memory Configuration Mode MCM, the Service Area can take one of the following address spaces:

MCM = 0	03C00H - 03FFFH	(16 K Traffic Memory)
MCM = 1	07C00H - 07FFFH	(32 K Traffic Memory)
MCM = 2 / 3 / 4	0FC00H - 0FFFFH	(64 K / 256 K / 1 M Traffic Memory)

Attention: For improved clarity, the addresses to the Service Area are referred with 0yC00H - 0yFFFH where 'y' stands for 3, 7 or F (hex) respectively.

The Service Area is divided into following sections:

- Physical Ports (PP)
  - Port Control and Status Registers
  - Data Area
- MVBC External Registers
  - Master Frame Slot to send individual Master Frames
  - Queue Descriptor Table
- MVBC Internal Registers
  - Physically located inside the MVBC.

The following table specifies two addresses. Address 03xxxH applies if MCM=0, otherwise 0FxxxH applies.

Address	Size	Description	Comments
0yFFFH			
0yF80H	128 Bytes	MVBC: Max. 32 internal regs.	Located inside MVBC (32-bit word-aligned)
0yF00H	128 Bytes	MVBC: Max. 32 external regs.	Located in Traffic Memory (16-bit word-aligned)
0yE00H	256 Bytes	Physical Ports: PP-PCS	32 ports x 4 words (16 of them are in use)
0yC00H	512 Bytes	Physical Ports: PP-Data	32 ports x 4 words (16 of them are in use)

Table 2.11: Service Area

2.8.6.1 Physical Ports

The following table summarizes the available Physical Ports:

Port	F-Code	Direction	Description
FC8	8	Source	Mastership Offer Source Port This port is used to send 1-word Mastership Offer Frames.
MOS	8	Sink	Mastership Offer Sink Port This port is used to receive 1-word Mastership Offer Frames.
EF0	9, 13, 14	Source	Event Frame Source Port for Event Type 0 This port is used to send 1-word Event Frames if Event Type ET=0 is specified in the incoming Master Frame.
EF1	9, 13, 14	Source	Event Frame Source Port for Event Type 1 This port is used to send 1-word Event Frames if Event Type ET=1 is specified in the incoming Master Frame.
EFS	9, 13, 14	Sink	Event Frame Sink Port This port is used to receive 1-word Event Frames
FC15	15	Source	Device Status Port This port is used to send 1-word Device Status Reports.
MSRC	12	Source	Message Source Port This port is used to transmit Message Data. If queuing is enabled, then the Message Queue will be used instead of the MSRC Data Area.
MSNK	12	Sink	Message Sink Port This port is used to receive Message Data. If queuing is enabled, then the Message Queue will be used instead of the MSRC Data Area.

Continued on on next page.

Continued from previous page.

Port	F-Code	Direction	Description
TSRC	any	Source	Test Source Port This port is used instead of any other port as the source if the UTQ-bit is set (see SCR, section 2.9.7). This port shall be used for internal loop-back tests only. Attention: Memory location of MSRC and TSCR are the same.
TSNK	any	Sink	Test Sink Port This port is used instead of any other port as the source if the UTS-bit is set (see SCR, section 2.9.7). This port shall be used for internal loop-back tests only. Attention: Memory location of MSNK and TSNK are the same.

Table 2.12: Physical Ports: Description

2.8.6.2 Physical Ports, PCS Table

Address	Contents	Address	Contents
0yE3FH		0yE7FH	
0yE38H	Device Status Port (FC15)	0yE78H	Message / Test Sink Port (MSNK / TSNK)
0yE30H	Mastership Offer Sink Port (MOS)	0yE70H	(size: 16 words; covers 4 docks)
0yE28H	Event Frame 1 Source Port (EF1)	0yE68H	
0yE20H	Event Frame 0 Source Port (EF0)	0yE60H	
0yE18H	(Reserved)	0yE58H	Message / Test Source Port (MSRC / TSRC)
0yE10H	(Reserved)	0yE50H	(size: 16 words; covers 4 docks)
0yE08H	Event Frame Sink Port (EFS)	0yE48H	
0yE00H	Mastership Offer Source Port (FC8)	0yE40H	

Table 2.13: Physical Ports: PCS

Attention: Locations 0yE80H-0yEFFH are reserved for future use.

Attention: The Test Ports TSRC and TSNK overlay the Message Ports MSRC and MSNK. For clarity reasons, separate names are maintained throughout this document.

2.8.6.3 Physical Ports, Data Area

The Data Area of the Physical Ports is organized similarly as the data for logical addressed and device addressed ports. A similar memory assignment as found in the other Data Area is used.

Address	Contents	Address	Contents
0yC7FH		0yCFFH	
0yC78H	Device Status Port (FC15), Pg 1	0yC78H	Message / Test Sink Port (MSRC / TSRC) Pg 1 (size: 16 words; covers 4 docks)
0yC70H	Mastership Offer Sink Port (MOS), Pg 1	0yC70H	
0yC68H	Event Frame 1 Source Pt. (EF1), Pg 1	0yC68H	
0yC60H	Event Frame 0 Source Pt. (EF0), Pg 1	0yC60H	
0yC58H	Device Status Port (FC15), Pg 0	0yC58H	Message / Test Sink Port (MSRC / TSRC) Pg 0 (size: 16 words; covers 4 docks)
0yC50H	Mastership Offer Sink Port (MOS), Pg 0	0yC50H	
0yC48H	Event Frame 1 Source Pt. (EF1), Pg 0	0yC48H	
0yC40H	Event Frame 0 Source Pt. (EF0), Pg 0	0yC40H	
0yC38H	(Reserved)	0yC38H	Message / Test Source Port (MSRC / TSRC) Pg 0 (size: 16 words; covers 4 docks)
0yC30H	(Reserved)	0yC30H	
0yC28H	Event Frame Sink Port (EFS), Pg 1	0yC28H	
0yC20H	Mastership Offer Source Pt. (FC8), Pg 1	0yC20H	
0yC18H	(Reserved)	0yC18H	Message / Test Source Port (MSRC / TSRC) Pg 0 (size: 16 words; covers 4 docks)
0yC10H	(Reserved)	0yC10H	
0yC08H	Event Frame Sink Port (EFS), Pg 0	0yC08H	
0yC00H	Mastership Offer Source Pt. (FC8), Pg 0	0yC00H	

Table 2.14: Physical Ports: Data Area

Attention: Locations 0yD00H-0yDFFH are reserved for future use.

Attention: The Test Ports TSRC and TSNK are equivalent to the Message Ports MSRC and MSNK. For clarity reasons, separate names are maintained throughout this document.

2.8.6.4 MVBC External Registers

The External Registers are located in the Traffic Memory and are accessed by both CPU and MVBC:

Name	Description	Symbol	Address
Master Frame Slot	Used for outgoing Master Frames	MFS	0yF00H
Queue Descriptor Table (QDT), see 2.8.7.1	Pointer to Transmit Queue 0 Pointer to Transmit Queue 1 Pointer to Receive Queue	QDT[0] QDT[1] QDT[2]	0yF10H 0yF12H 0yF14H

Table 2.15: External Registers

MFS: The Master Frame is a 16-bit word which consists of an F-Code (4 bits) and an address (12 bits). The remaining locations are not used. They are reserved for future use and shall not be used as general-purpose data storage.

2.8.6.5 MVBC Internal Registers

The register addresses are 32-bit word aligned in order to simply operating with 32-bit processors.

MVBC Functional Unit	Register Name	Symbol	Address	Section
Status Control Register	Status Control Register	SCR	0yF80H	2.9.7
Address Logic	Memory Configuration Register	MCR	0xF84H	2.9.10.1
Decoder	Decoder Register	DR	0yF88H	2.9.4.5
Sink-Time Supervision	Sink-Time Supervision Register	STSR	0yF8CH	2.9.17.1
Telegram Analysis Unit	Frame Counter Error Counter Master Frame Register Master Frame Reg. Duplicate Exception	FC EC MFR MFRE	0yF90H 0yF95H 0yF98H 0yF9CH	2.9.6.4 " " 2.9.6.5
Main Control Unit	Master Register Secondary Master Register Dispatch Pointer Register Secondary Dispatch Pointer Register	MR MR2 DPR DPR2	0yFA0H 0yFA4H 0yFA8H 0yFACH	2.9.8.1 " 2.9.8.2 "
Interrupt Logic	Interrupt Pending Register 0 Interrupt Pending Register 1 Interrupt Mask Register 0 Interrupt Mask Register 1 Interrupt Status Register 0 Interrupt Status Register 1 Interrupt Vector Register 0 Interrupt Vector Register 1	IPR0 IPR1 IMR0 IMR1 ISR0 ISR1 IVR0 IVR1	0yFB0H 0yFB4H 0yFB8H 0yFBCH 0yFC0H 0yFC4H 0yFC8H 0yFCCH	2.9.15.3 " 2.9.15.4 " 2.9.15.5 " 2.9.15.6 "
Device Address	Device Address Override Register Device Address Override Key	DAOR DAOK	0yFD8H 0yFDCH	2.9.9.1 "
Universal Timer	Timer Control Register Timer Reload Register 1 Timer Reload Register 2 Timer Counter Register 1 Timer Counter Register 2	TCR TR1 TR2 TC1 TC2	0yFE0H 0yFF0H 0yFF4H 0yFF8H 0yFFCH	2.9.16.1 " " " "

Table 2.16: Internal Registers

The remaining addresses shown below are reserved for future use:

0yFD0H, 0yFD4H, 0yFE4H, 0yFE8H, 0yFECH

Attention: For register access, the MVBC will not decode address bits 0 and 1. Therefore, the same register is accessed without any word realignment if address 0yXXXXH, 0yXXXXH+1, 0yXXXXH+2 or 0yXXXXH+3 is used.

Example: A CPU attempt to access a 32-bit word from address 0yFFCH where the host-side logic splits it into two 16-bit accesses will automatically lead to fitting the 16-bit counter value twice into the 32-bit word.

2.8.7 Message Queues

The MVBC supports three Message Queues: two Transmit Queues with different priorities and one Receive Queue. The queues are organized as linked lists and may be located in any unused region in the Traffic Memory.

The Message Queues are used when the QA-bit in the PCS is active. The SRC and SINK-bits indicate whether the Transmit or Receive Queues are used. The following table summarizes the queue assignment:

SINK	SRC	QA	Description
X	X	0	No queue attached (set and reset by user only)
1	0	1	Receive Queue attached
X	1	1	Transmit Queues attached. Queue 0 (high priority) is always checked before queue 1 (low priority). Queue 1 is serviced only if queue 0 is empty or nonexistent.

Table 2.17: Conditions to Select Message Queues

Attention: High and low priority queues have nothing in common with the priorities related to different Event Types used in Event Arbitration.

If both QA and TWCS bits are active in the PCS, then the Check Sequences will still be read or written to their usual locations, namely PCS Word 3.

2.8.7.1 Queue Descriptor Table (QDT)

All three queues are accessible via the Queue Descriptor Table (QDT). The QDT lies in the Service Area of the Traffic Memory (Address 0yF10H, 0yF12H, 0yF14H) and contains the pointers to all three queues. A zero pointer indicates a *nonexistent queue*.

2.8.7.2 Queue Address Evaluation

The 19-bit (No A_0) address to the Traffic Memory is computed from any 16-bit queue pointer by shifting it 2 bits to the left and adding an offset (Valid offset values: 00000H, 40000H, 80000H or C0000H). The Queue Offset ($QO_{1..0}$) is defined in the Memory Configuration Register. Hence, the QDT, LLR and queue data blocks must be located within a selected 256 K block.

2.8.7.3 Linked List Records (LLR)

The pointer from the QDT points to the first Linked List Record (LLR). Each LLR consists of two 16-bit pointers: Data Pointer (DP) and Next Pointer (NP). The DP points to a 16-word message block. This data structure allows packet generation without physical data movement as long the data is already stored in the Traffic Memory. The NP points to the next LLR. *End of queue* is indicated with DP=NULL and NP=*Don't Care*. An *empty queue* consists of one LLR with DP = NULL. The following figure illustrates the data structure. DP must be aligned to 16-word blocks.

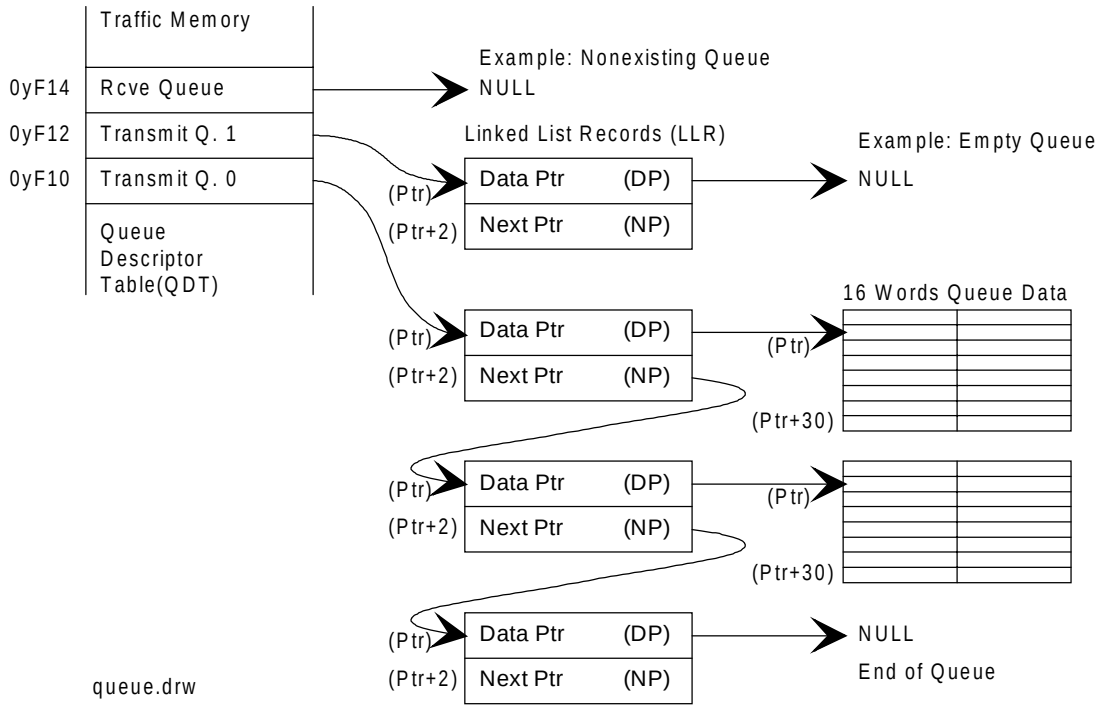


Figure 2.12: Queue Data Structure

The MVBC does not create any queue data structures by itself. The user must install a linked list structure of sufficient size for all three queues before queuing can be activated. The queue data records have a fixed size of 16 words. Following rules must be observed when installing a linked list for queuing:

- The queue structure must not overlap into any active TM regions (i.e. active port data, PCS, Service Area, Master Frame Tables)
- The start address to an LLR must be aligned to a 4 byte block (2 word block).
- The start address to a Data Record must be aligned to a 32 byte block (16 word block).
- The entire linked-list structure must be located within an aligned 256 KB block which is selected by the Queue Offset ($QO_{1..0}$).

2.8.8 Master Frame Tables (MF-Tables)

The MVBC provides an automatic Master Frame Dispatcher. Details about its operation are specified in section 3.10. Each Master Frame Table contains between 1 and 32 Master Frame Words. The Master Frame Tables may be stored in any yet unused location in the Traffic Memory. The user is responsible not to let the MF-Tables overlap any active TM areas (i.e. Port Index Table, active ports (PCS/Data/Force Table), Service Area, Message Queues).

The Master Frame Table is pointed by the 16-bit Dispatch Pointer Register (DPR) where the lower two bits are tied to zero. The table size is specified in the Master Register (MR). The pointer is shifted 2 bits to the left and a Master Frame Offset is added. Valid offset values are 00000H, 40000H, 80000H or C0000H. The Master Frame Offset ($MO_{1..0}$) is defined in the Memory Configuration Register. Following rules must be observed when defining Master Frame Tables:

- The Master Frame Tables must not overlap into any active TM regions (i.e. active port data, PCS, Queues)
- The start address to any Master Frame Table must be aligned to a 16 byte block (8 word block) if the Master Frame Table contains not more than 8 words.
- The start address to any Master Frame Table must be aligned to a 32 byte block (16 word block) if the Master Frame Table contains not more than 16 words.

- The start address to any Master Frame Table must be aligned to a 64 byte block (32 word block) if the Master Frame Table contains not more than 32 words.
- All Master Frame Tables must be located within an aligned 256 KB block which is selected by the Master Frame Offset ($MO_{1..0}$).

  In other words, smaller Master Frame Tables with 8, 16 or 24 Master Frame bodies may be defined in order to use the Traffic Memory more efficiently.

WRONG:

Address	Contents
0y1050H	
0y1040H	Table 2 (overrun)
0y1010H	
0y1000H	Table1

RIGHT:

Address	Contents
0y1050H	
0y1040H	Table 1
0y1010H	Table 2
0y1000H	

Best software approach: The Bus Administrator Software (BAS) should initialize the tables during start-up and assign pointers every time the next cyclic Master Frames are to be sent out. The plausibility of the tables can be tested with a common CRC algorithm which is typically used to test read-only memories.

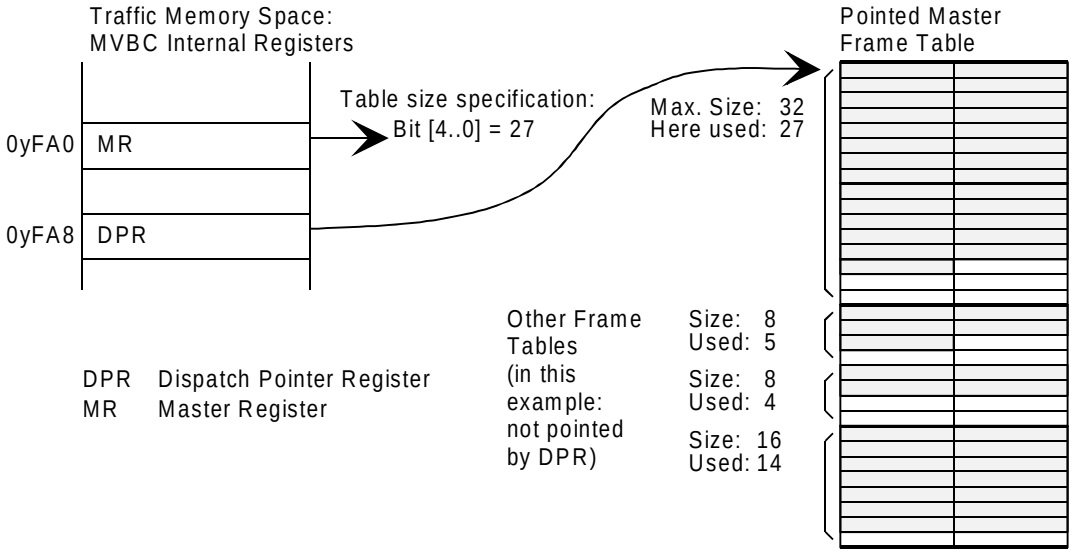


Figure 2.13: Master Frame Table Structure

2.8.9 Address Generation Tables

The following tables summarize the layout of the Traffic Memory by specifying the address bits.

Symbol	Description
$LA_{11..0}$	Logical Address (For MCM=0-1, the same word in PIT is accessed regardless if LA is even or odd)
$DA_{11..0}$	Device Address (For MCM=1, the same word in PIT is accessed regardless if DA is even or odd)
$PI_{11..0}$	Port Index
VP	Valid Page Pointer
$C_{4..0}$	Word Counter
$R_{6..1}$	Addresses Internal Registers (Service Area)
$QO_{1..0}$	Queue Offset (From Memory Configuration Register (MCR))
$MO_{1..0}$	Master Frame Offset (From Memory Configuration Register (MCR))
$P_{15..0}$	Master Frame and Queue Pointer bits
$T_{1..0}$	2-bit arithmetic sum of Port Index (PI) and Counter (C): $(T_1,T_0) = (PI_1,PI_0) + (C_3,C_2)$;
$X_{1..0}$	2-bit arithmetic sum of Pointer (P) and Counter (C): $(X_1,X_0) = (P_3,P_2) + (C_4,C_3)$;

Type	MCM	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1
LA-PIT	0,1 2,3,4	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 LA_{11}	LA_{11} LA_{10} LA_9 LA_8 LA_{10} LA_9 LA_8 LA_7	LA_7 LA_6 LA_5 LA_4 LA_6 LA_5 LA_4 LA_3	LA_3 LA_2 LA_1 LA_2 LA_1 LA_0
DA-PIT	1 2,3,4	0 0 0 0 0 0 0 0	0 1 0 0 0 0 1 DA_{11}	DA_{11} DA_{10} DA_9 DA_8 DA_{10} DA_9 DA_8 DA_7	DA_7 DA_6 DA_5 DA_4 DA_6 DA_5 DA_4 DA_3	DA_3 DA_2 DA_1 DA_2 DA_1 DA_0
LA-Data	0,1 2 3,4	0 0 0 0 0 0 0 0 0 0 0 1	0 0 0 1 0 1 PI_9 PI_8 PI_{11} PI_{10} PI_9 PI_8	PI_7 PI_6 PI_5 PI_4 PI_7 PI_6 PI_5 PI_4 PI_7 PI_6 PI_5 PI_4	PI_3 PI_2 VP T_1 PI_3 PI_2 VP T_1 PI_3 PI_2 VP T_1	T_0 C_1 C_0 T_0 C_1 C_0 T_0 C_1 C_0
DA-Data	1 2 3 4	0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0	0 1 0 1 1 1 1 0 1 PI_{10} PI_9 PI_8 PI_{11} PI_{10} PI_9 PI_8	PI_7 PI_6 PI_5 PI_4 PI_7 PI_6 PI_5 PI_4 PI_7 PI_6 PI_5 PI_4 PI_7 PI_6 PI_5 PI_4	PI_3 PI_2 VP T_1 PI_3 PI_2 VP T_1 PI_3 PI_2 VP T_1 PI_3 PI_2 VP T_1	T_0 C_1 C_0 T_0 C_1 C_0 T_0 C_1 C_0 T_0 C_1 C_0
LA-Force Table	0,1 2 3,4	0 0 0 0 0 0 0 0 0 0 1 0	0 0 1 0 1 0 PI_9 PI_8 PI_{11} PI_{10} PI_9 PI_8	PI_7 PI_6 PI_5 PI_4 PI_7 PI_6 PI_5 PI_4 PI_7 PI_6 PI_5 PI_4	PI_3 PI_2 VP T_1 PI_3 PI_2 VP T_1 PI_3 PI_2 VP T_1	T_0 C_1 C_0 T_0 C_1 C_0 T_0 C_1 C_0
LA-PCS	0,1 2 3,4	0 0 0 0 0 0 0 0 0 0 1 1	0 0 1 1 1 1 0 PI_9 0 PI_{11} PI_{10} PI_9	0 PI_7 PI_6 PI_5 PI_8 PI_7 PI_6 PI_5 PI_8 PI_7 PI_6 PI_5	PI_4 PI_3 PI_2 T_1 PI_4 PI_3 PI_2 T_1 PI_4 PI_3 PI_2 T_1	T_0 C_1 C_0 T_0 C_1 C_0 T_0 C_1 C_0
DA-PCS	1 2 3 4	0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1	0 1 1 1 1 1 1 1 0 1 PI_{10} PI_9 1 PI_{11} PI_{10} PI_9	0 PI_7 PI_6 PI_5 0 PI_7 PI_6 PI_5 PI_8 PI_7 PI_6 PI_5 PI_8 PI_7 PI_6 PI_5	PI_4 PI_3 PI_2 T_1 PI_4 PI_3 PI_2 T_1 PI_4 PI_3 PI_2 T_1 PI_4 PI_3 PI_2 T_1	T_0 C_1 C_0 T_0 C_1 C_0 T_0 C_1 C_0 T_0 C_1 C_0

Service Area (Start Address, Physical Ports, Master Frame Slot, Queue Descriptor Table, Int. Registers):

Type	MCM	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1
Start Address	0 1 2,3,4	0 0 0 0 0 0 0 0 0 0 0 0	0 0 1 1 0 1 1 1 1 1 1 1	1 1 * * 1 1 * * 1 1 * *	* * * * * * * * * * * *	* * * * * * * * *
PP-Data	all	* * * *	* * * *	* * 0 0	PI_3 PI_2 VP T_1	T_0 C_1 C_0
PP-PCS	all	* * * *	* * * *	* * 1 0	0 PI_3 PI_2 T_1	T_0 C_1 C_0
MFS	all	* * * *	* * * *	* * 1 1	0 0 0 0	0 0 0
QDT[0] QDT[1] QDT[2]	all all all	* * * * * * * * * * * *	* * * * * * * * * * * *	* * 1 1 * * 1 1 * * 1 1	0 0 0 1 0 0 0 1 0 0 0 1	0 0 0 0 1 0 1 0 0
Int.Regs	all	* * * *	* * * *	* * 1 1	1 R_6 R_5 R_4	R_3 R_2 0

Port Index Assignment (PI3..0) for Physical Ports (PP-Data/PP-PCS):

0 FC8	1 EFS	2 *Reserved*	3 *Reserved*
4 EF0	5 EF1	6 MOS	7 FC15
8 MSRC/TSRC	(4 docks)	C MSNK/TSNK	(4 docks)

Miscellaneous Assignment, Class 1 Mode:

Pointer / Mode	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1
LLR Queue Data MF-Tables Class 1 Mode	QO_1 QO_0 P_{15} P_{14} QO_1 QO_0 P_{15} P_{14} MO_1 MO_0 P_{15} P_{14} 1 1 1 S_0	P_{13} P_{12} P_{11} P_{10} P_{13} P_{12} P_{11} P_{10} P_{13} P_{12} P_{11} P_{10} S_{15} S_{14} S_{13} S_{12}	P_9 P_8 P_7 P_6 P_9 P_8 P_7 P_6 P_9 P_8 P_7 P_6 S_{11} S_{10} S_9 S_8	P_5 P_4 P_3 P_2 P_5 P_4 X_1 X_0 P_5 P_4 X_1 X_0 S_7 S_6 S_5 S_4	P_1 P_0 0 C_2 C_1 C_0 C_2 C_1 C_0 S_3 S_2 S_1

$S_{15..0}$ are active low decoded select signals. Only one signal may be active at a time.

Table 2.18: Address Generation Tables

2.9 Hardware Overview

The MVBC consists of the following functional blocks:

- Encoder - Generation of Manchester code and transmission of frames
- Transmission Buffer - 16 Words + 4 Check Sequences
- Decoder - Reception, Manchester decoding, data extraction and error checking
- Receive Buffer - 16 Words + 4 Check Sequences
- Telegram Analysis Unit
  - Detects Master Frame (MF) and Slave Frame (SF) timeouts
  - Frame level errors (duplicate/missing MF, SF), error statistics
- Status Control Registers - MVBC configuration: No. wait states, initialization level, etc.
- Main Control Unit
  - Supports MVBC functions to operate both as Master and Slave
  - Supports queued message transfers
- Dev. Addr Read & Store Unit - hardware-defined Device Address can be overridden by different value
- Address Logic
  - Decoder of CPU addresses applied to MVBC to select int. registers
  - Encoder to generate outgoing addresses to Traffic Memory
- Arbitration Controller
  - Governs access to Traffic Memory between MVBC and host CPU
  - Different arbitration modi are supported to assure data consistency
- Traffic Memory Controller
  - Controls access from CPU and MVBC to Traffic Memory
  - Built-in waitstate logic
- Bus Multiplexer / Forcing
  - Handles all internal data transfers in the MVBC
  - Enables data forcing, byte swapping
- Class 1 Logic
  - Permits operation without assistance of CPU or microcontroller
  - 16 ports à 16 bits are provided for Process Data
- Interrupt Logic
  - Interrupt vectors are provided for convenience
  - Generates interrupts on user-specified data transfers
  - Reports exceptions (data transfer errors)
  - Supports external incoming interrupt signals
- Two Universal Timers
  - Timer 1: 10 µs up to 650 ms, 10 µs resolution
  - Timer 2: 125 ns up to ca 8 ms, 125 ns resolution
  - Timer output signal (TMR1\,TMR2\) and interrupt capability
  - Multi-MVBC synchronization capabilities
- Sink-Time Supervision
  - Helps to detect irregularities while transferring Process Data
  - Any number of ports from 0 through 5096 can be supervised
  - Selectable supervision period from 1 ms, 2, 4, 8, ... 256 ms
- Clock Generator - Generates all clock and counter signals for the MVBC
- Test Support
  - Ad-Hoc: Internal loop-back, high degree of functional observability
  - JTAG Boundary Scan; Internal Scan; MUX-Isolation of internal RAMs

2.9.1 Block Diagram

See next page.

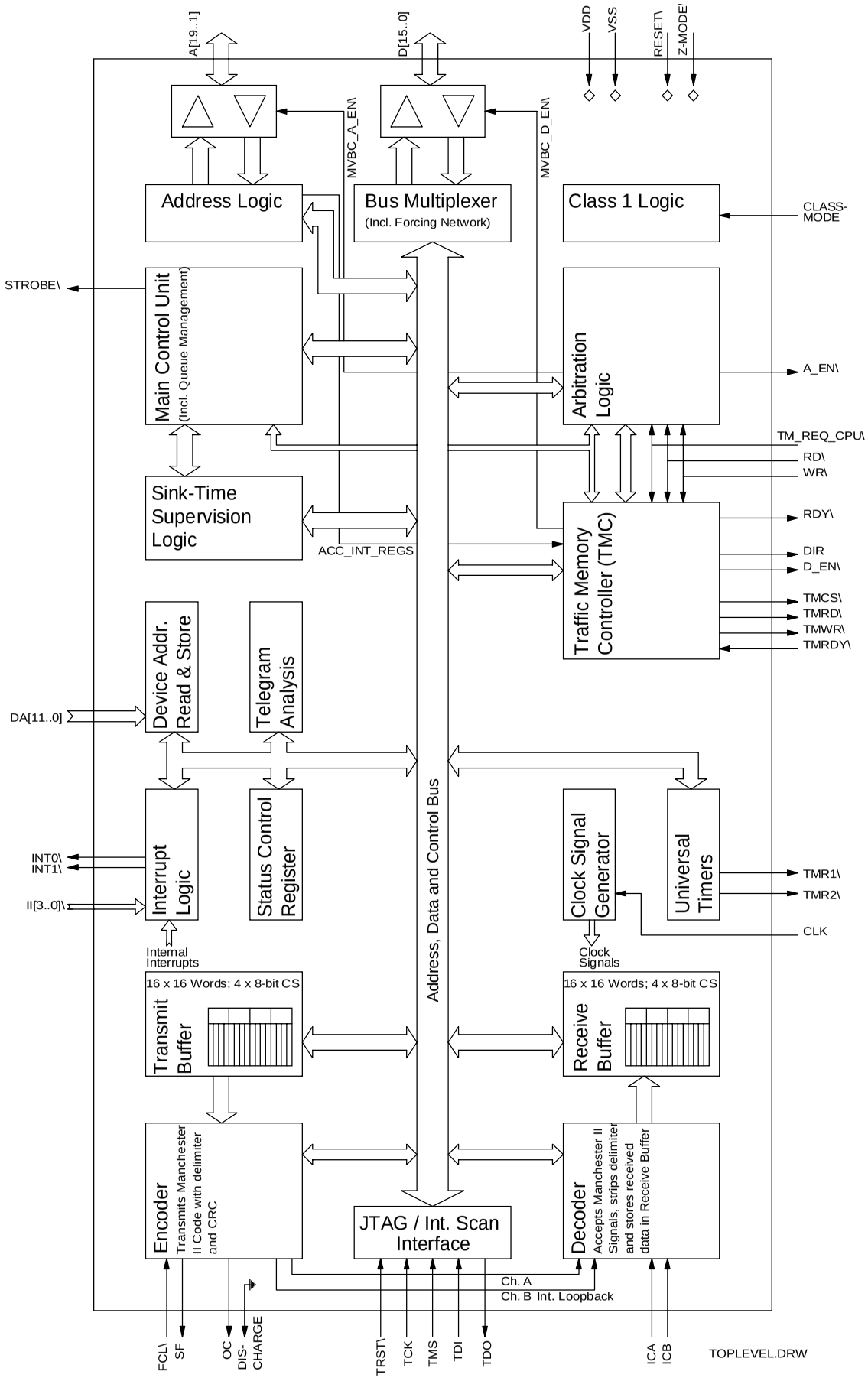


Figure 2.14: MVBC Top-Level Block Diagram

2.9.2 Encoder

The Encoder converts 16-bit data into a 1.5 Mbit/s serial Manchester Biphase L encoded data stream and transmits it over the common output pin OC. The data is supplemented with a start bit, Master or Slave Delimiter and with one or more 8-bit Check Sequences. The code complies with the IEC TCN Standard [1].

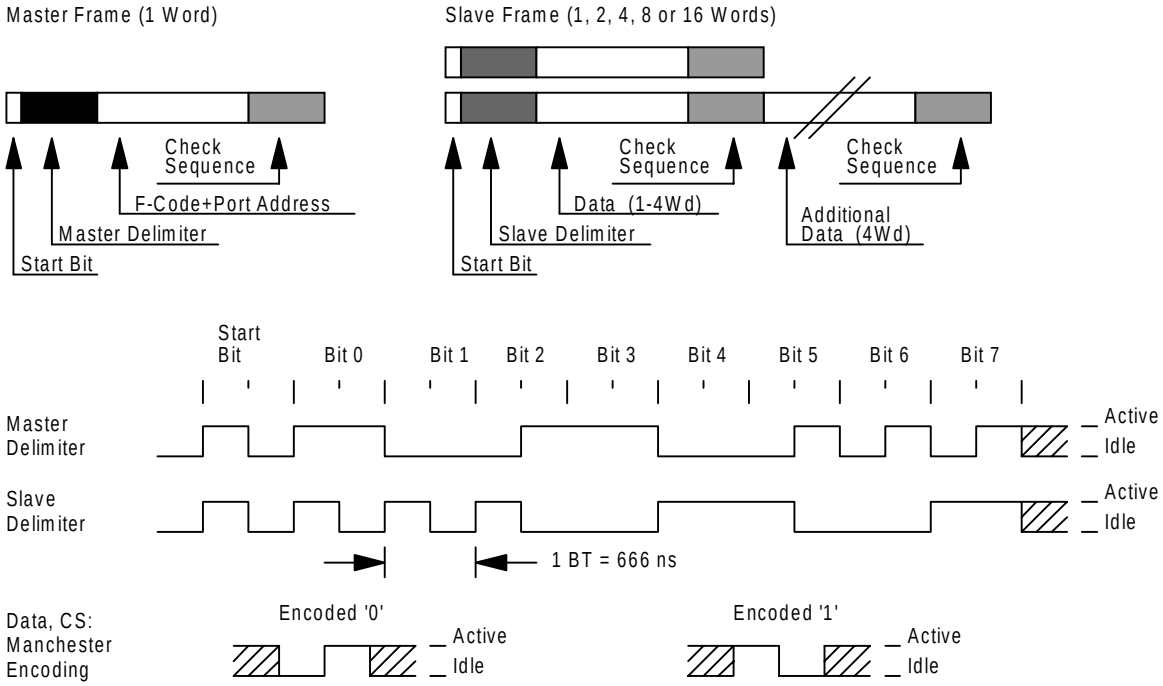


Figure 2.15: Manchester Coding Scheme

The CRC is evaluated from the generator polynomial shown below:

$$G(x) = X^7 + X^6 + X^5 + X^2 + 1$$

This polynomial guarantees a Hamming distance of 4. The overall Hamming distance of 8 is achieved by combining Manchester Biphase coding and CRC. The user can request to transmit user-supplied Check Sequences in the Slave Frames by activating TWCS in the PCS for selected ports. In this case, the same algorithm must be applied in order to assure correct data transfers. If the CRC mismatches, then the MVBC assumes that the user intends to forward garbled data intentionally.

Attention: All types of communication errors, including plain CRC mismatches, are treated as if they occurred on the MVB. Therefore, the bus participants may take actions such as line switchover.

2.9.2.1 Control Signals for Physical Layer Drivers

The serial data is output via the OC pin. The active high signal "Send Frame" (SF) stays active from 125 ns before the beginning and until 125 ns after the end of the transmitted frames.

An external input pin "Force Constant Light" (FCL\) is available to tie the output signal to high level in order to adjust signal driver strengths when using fiber optical medium.

A jabber-hold mechanism is available to suppress the serial outputs and keep the SF-signal disabled if the Encoder starts transmitting permanently due to a hardware fault. This mechanism is activated if the QUIET-bit in the SCR (See section 2.9.7) is active.

2.9.2.2 Minimum Frame Spacing Enforcement

The Encoder assures that minimum space requirements among Master and Slave Frames:

t_{MS} = 1.4 us (Space between end of Master Frame and beginning of Slave Frame)
t_{SM} = 4.0 us (Space between end of Slave Frame and beginning of Master Frame)

2.9.3 Transmission Buffer (TXB)

The Transmission Buffer (TXB) serves as an intermediate buffer to store the next frame to be transmitted. This buffer is not visible to the user and is exclusively controlled by the Main Control Unit (MCU).

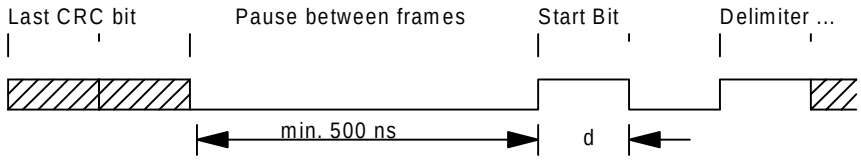
Capacity: 16 x 16-bit data words plus 4 x 8-bit Check Sequences

2.9.4 Decoder

The Decoder receives incoming Manchester biphase L signals over the *trusted* (or active) line and monitors the *observed* (or redundant) line for valid Manchester code. At power-up, the input pins ICA and ICB are assigned as the trusted and observed lines respectively. While frames are received, the Decoder identifies frame type (Master or Slave Frame) from the delimiter and stores the data and 8-bit CRC in the Receive Buffer (RXB). The built-in error detection mechanism guarantees a Hamming Distance of 8.

2.9.4.1 Signal Detection

A valid start bit, which occurs after a pause of at least 500 ns (0.75 bit times (BT)), must be detected in order to receive a frame. Shorter pauses cannot be detected. The start bit is recognized as long its duration (*d*) complies with the following limits:



d = minimum 125 ns = 1/2 BT - 5/16 BT
typical 333 ns = 1/2 BT
maximum 541 ns = 1/2 BT + 5/16 BT

Figure 2.16: Start Bit Detection

Frames with invalid start bits will be ignored and treated as noise. The Decoder synchronizes into the falling edge of the start bit and starts decoding the signal using mid-step signal detection and a sampling rate 16 x the data rate. At every transition, the Decoder performs self-synchronization and is able to adjust to jitter (± 125 ns, or ± 3/16 BT) and data rate deviations between sending and receiving devices (± 41.7 ns, or 1/16 BT). The shaded regions, as shown in figure 2.16, illustrate the regions where transitions are allowed. Outside these regions, known as *forbidden zones*, the signal will be sampled twice (rhombus symbols). This dual method assures reliable sampling and signal quality monitoring. If the edge is detected within forbidden zones (outside shaded areas), then a false sampling is made and a Signal Quality Error (SQE) will be reported.

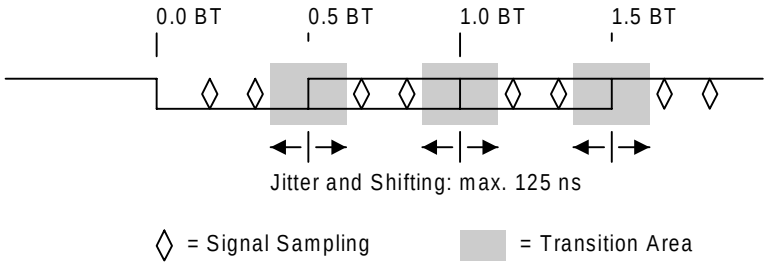


Figure 2.17: Signal Detection

2.9.4.2 Signal Monitoring

The Decoder monitors the incoming signals for following irregularities:

- Signal Quality Errors (SQE), see section 2.9.4.1.
- Manchester Code Violation
- Active Level Overbalance (ALO)
- Mismatching CRC
- Frame too short or too long
- Slave Frame Reply Timeouts

A Manchester Code Violation is reported when no transition is found in the middle of a received bit. However, the Decoder tolerates the violations defined in the Master and Slave delimiters (signal is stable for 3/2 BT).

The Manchester biphase L signal foresees a balance between "Low" and "High" signal level. An Active Level Overbalance (ALO) is reported if the difference between sampled "High" and "Low" half-bits exceeds +5.

The Decoder evaluates the CRC of the incoming frames according to the CRC algorithm specified in [1]. A CRC error is issued if the reconstructed CRC differs from the received CRC.

Frame size checking: The Decoder knows the expected frame length at the moment a valid delimiter has been received. Master Frames contain one word only. The expected size of a Slave Frame is derived from the F-Code of the previously received Master Frame.

If an error is detected inside the delimiter, then the Decoder ignores the frame (timeout!) and tries to re-synchronize on the start bit of the next frame. If an error is detected inside the data or CRC fields after receiving a correct delimiter, the Decoder reports the error to the Telegram Analysis Unit and to the PCS. If an error occurs, but the frame size remains unchanged, then the Decoder tries to recover the data if requested (active WA bit in the PCS).

2.9.4.3 Redundant Line Supervision

The trusted line is indicated by the LAA bit (Line A Active) in the Decoder Register (DR). Unless the Decoder operates in Single Line Mode (SLM bit in DR active), both ICA and ICB are sensitive and line switchover is enabled. Otherwise, if the user sets the SLM bit, the Decoder considers the trusted line (ICA or ICB, whichever is active at the moment) as the single input and line switchover is disabled. If ICB (or ICA) is the only connected line and LAA is active (or inactive), then the user must invoke a manual line switch before setting SLM.

The observed line is monitored for following types of errors:

- Signal Quality Errors (SQE)
- Manchester Code Violation
- Silence while a frame is received on the trusted line
- The frames on the trusted and observed line do not overlap at all.
- Permanent signal reception exceeding 1000...1250 µs

The RLD-bit (Redundant Line Disturbed) in the DR is set if one of the errors listed above occurs.

The Decoder will always wait until the complete frame has been received on both lines except when at least one of the following items apply:

- RLD-bit is set
- SLM-bit is set
- The observed line remained silent throughout the period the entire frame has been received over the trusted line.
- Permanent signal reception on the observed line

Line Switchover (applies if SLM = '0'):

A switch from the trusted line to the observed line takes place if one of the following conditions are met:

1. Erroneous frame or Reply Timeout on the trusted line, regardless if valid data has been detected on the observed line. Exception: No line switch will take place during Event Arbitration (F-Codes 9, 13, 14). The line switch takes place when nothing is received at the moment or at the next pause. RLD is set.

2. Silence: No valid Master Frame has been received within 1.4 ms. If both lines are silent, the MVBC will switch lines at 1.4 ms intervals until a Master Frame has been received again. RLD bit is set.

3. A Master Frame containing a Device Status Poll (F-Code=15) has been received and the Device Address matches. The addressed MVBC performs following steps:

   - If RLD is not set, then a line switch will be invoked
   - The RLD will be cleared
   - The Device Status Report will be transmitted (Class 1 Mode: contains old RLD)

4. User-invoked Line Switchover

   If the user activates the LS-Bit (inside DR), then the line switch will take place at the next pause or when the MVB is silent. The MVBC resets LS to zero after the line switch has taken place.

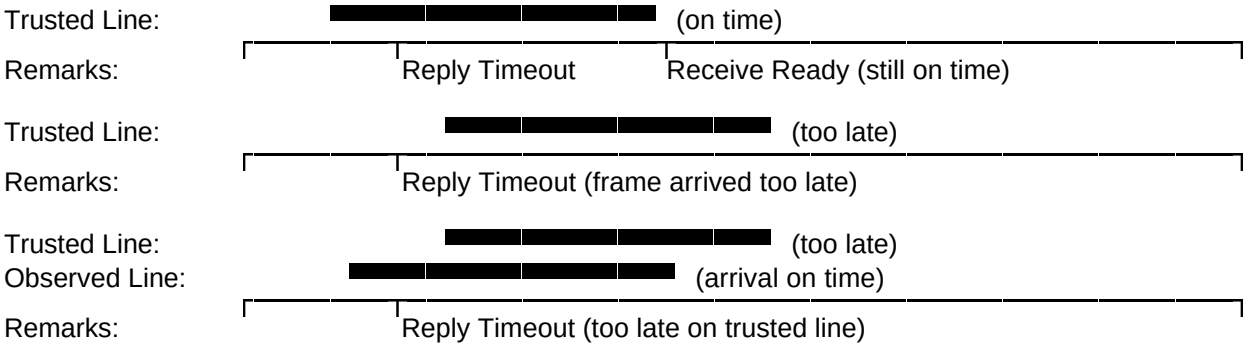
Attention: If RLD is active, then line switchover is restricted to the conditions stated in item 2.

2.9.4.4 Frame Reception Scenarios

The following diagrams illustrate the behavior of the Decoder for different scenarios while both lines are active (SLM=0).

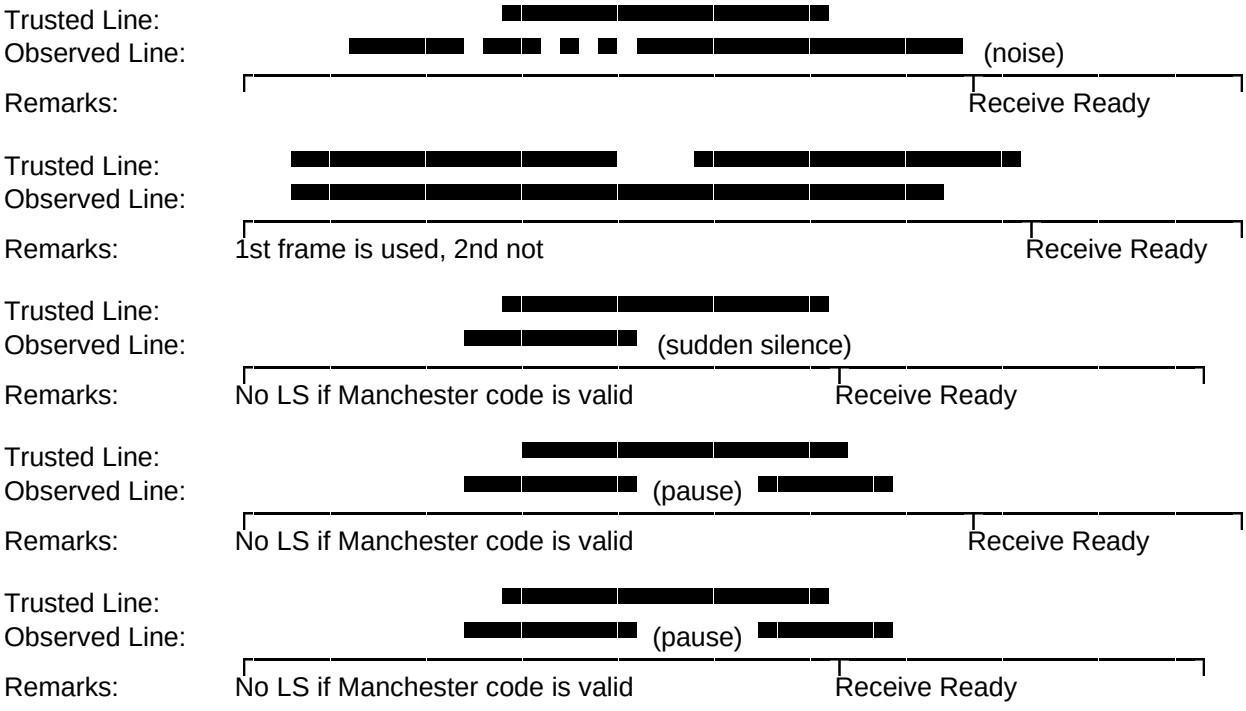
See figure 2.18 on the next page.

Timeouts:



Valid reception with skew:

Noise / Collisions:



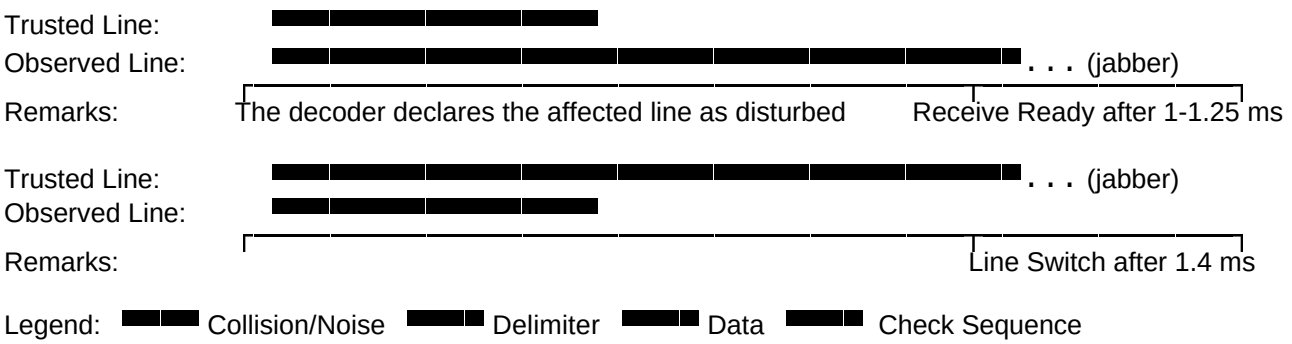


Figure 2.18: Communication Scenarios

2.9.4.5 Decoder Register (DR)

Decoder Register (DR): Address 0yF88H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	-												LAA	RLD	LS	SLM
Init. Value:	All 0												1	0	0	0
CPU Acc.:	-												R	RW0	RW1	RW
MVBC Acc.:	-												rw	rw	rw	r

Symbol	Description
LAA	Line A Active (trusted), inverted at every line switch 0 Line B active (Line A is observed) 1 Line A active (Line B is observed)
RLS	Redundant Line Disturbed 0 Observed line is O.K. 1 Bad data or silence detected on observed line Attention: This bit is cleared automatically when a Device Status Poll (F-Code 15) addresses this MVBC
LS	Activate Line Switchover 0 No line switchover in progress or filed 1 Line switchover in progress or filed
SLM	SLM Single Line Mode 0 Line switchover by MVBC is allowed 1 Only the line selected by LAA will be used to receive data. Line switchovers are suppressed.

Table 2.19: Decoder Register (DR)

2.9.5 Receive Buffer (RXB)

The Receive Buffer (RXB) serves as an intermediate buffer to store the last received frame. This buffer is not visible to the user and is exclusively controlled by the Main Control Unit (MCU).

Capacity: 16 x 16-bit data words plus 4 x 8-bit Check Sequences

2.9.6 Telegram Analysis Unit

The Telegram Analysis Unit pursues following tasks:

- Reporting incoming Slave Frames Section 2.9.6.1
- Timeout Mechanisms Section 2.9.6.2
- Telegram Error Handling Section 2.9.6.3
- Master Frame Registers Section 0
- Telegram Error Recordkeeping Section 2.9.6.5

2.9.6.1 Incoming Slave Frames

At the moment the Decoder reports that a valid or erroneous Slave Frame has arrived (at least the Slave Delimiter is valid) has been received, the interrupt

"Slave Frame Checked" (SFC)

is asserted.

2.9.6.2 Timeout Mechanisms

The Telegram Analysis Unit generates three different Timeout signals, as long the MVBC is operating in full-functional or loopback mode (Initialization Level 2 or 3, see SCR, section 2.9.7):

- Reply Timeout See below
- Bus Timeout 1.30 ms
- Line Timeout 1.42 ms

The Slave Frame Reply Timeout signal is used to signal a timeout if no Slave Frame arrives, or if the delay is too large. The signal is passed to the Decoder which checks whether a Frame is currently received or not. If not, then the interrupt

"Reply Timeout Interrupt" (RTI)

is asserted. This interrupt does not occur more than once after a Master Frame. The Timeout Coefficient is adjustable with the SCR bits TMO1..0. The initial value is 42.7 µs and complies with the MVB Specification [1]. All other values do not comply with the specifications and must not be used in open MVB systems.

- 21.3 µs (50% less)
- 42.7 µs (default value)
- 64.0 µs (50% more)
- 85.4 µs (100% more)

RTI can occur during a true timeout or a strong-overlapping collision where the delimiters have been garbled. If an active port is affected, then the PCS Word 1 will be updated with STO=1. BNI may be active if the line was not silent. The Bus Timeout occurs 1.3 ms after the end of the last Master Frame. If no Master Frame has been received within this period, then the interrupt

"Bus Timeout Interrupt" (BTI)

will be asserted. This timeout does not occur more than once after a Master Frame. The Line Timeout occurs 1.42 ms after the end of the last Master Frame. This timeout forces a line switch at the Decoder, given the Decoder does not operate in Single Line Mode (SLM=1). As long no Master Frame has been received, this timeout signal at 1.42 ms intervals in order to poll both signal inputs.

2.9.6.3 Telegram Error Handling

Error information's are obtained from the Decoder, processed and suggestions for further actions are passed to the Main Control Unit (MCU). The state diagram on Figure 2.19 illustrates how subsequent frames are handled if errors have occurred or not. If the delimiter has been recognized, but the received frame contains an error, then one of the following interrupts can occur:

"Erroneous Master Frame" (EMF)

"Erroneous Slave Frame" (ESF)

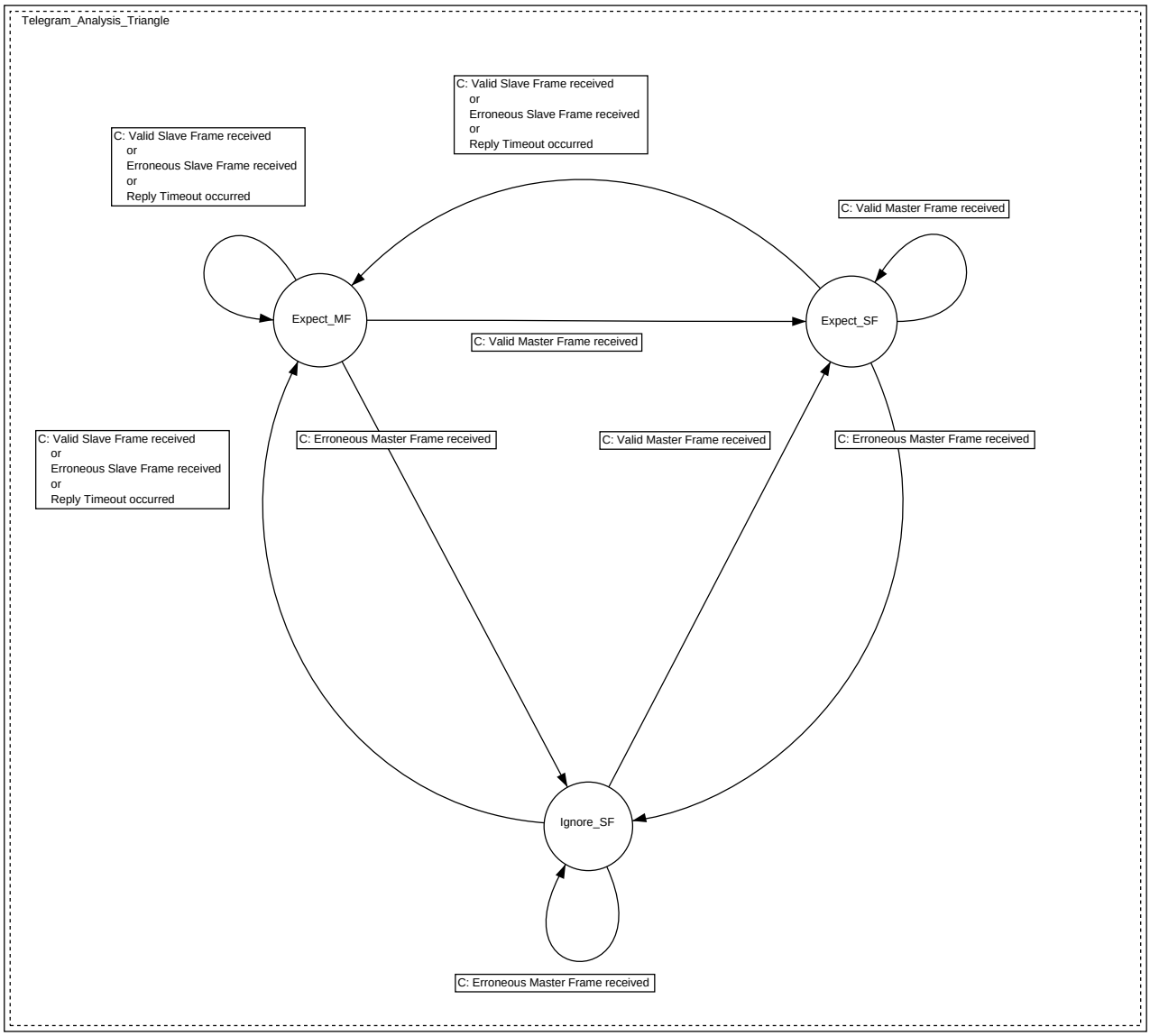


Figure 2.19: Telegram Analysis Principal States

If ESF occurs and an active port is affected, then the PCS Word 1 will be updated with TERR=1 and BNI=1. SQE, ALO and CRC contain detailed error information's. If EMF occurs, then no port will be affected. If two or more subsequent Master Frames are received before Reply Timeout occurs, then the last Master Frame applies. The previous Master Frames will be ignored. The interrupt

"Duplicate Master Frame" (DMF)

will be asserted. If two or more subsequent Slave Frames are received, the first Slave Frame will be handled. All other frames are ignored. This case also occurs if a valid Slave Frame is received after a Reply Timeout. The interrupt

"Duplicate Slave Frame" (DSF)

will be asserted.

2.9.6.4 Master Frame Registers (MFR, MFRE)

Two registers are foreseen for capturing the Master Frame:

- Master Frame Register (MFR)
- Master Frame Register Duplicate on Exception occurrence (MFRE)

The MFR contains the last Master Frame received. It also applies to Master Frames sent by this MVBC which returns back into the Decoder. The MFR serves for following purposes:

- F-Code Checking, Comparison of 12-bit address, Port Selection
- Frame size Comparison for the next arriving Slave Frame
- Evaluation of various Traffic Memory Addresses

If an interrupt due to an exception is generated, then the contents in MFR are copied into MFRE. Then, MFRE will be frozen until the interrupt has been serviced (corresponding Interrupt Status bits in ISR are cleared, see Interrupt Logic, section 2.9.15). MFRE will not be frozen if the affected interrupt is masked.

Following interrupts are considered as exceptions:

EMF, ESF, DMF, DSF, RTI, BTI

Both MFR and MFRE use the same 16-bit Master Frame Format:

Master Frame Registers (MFR): Address 0yF98H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$F\text{-}Code_{3..0}$				$Addr_{11..0}$											
Init. Value:	all 0				all 0											
CPU Acc.:	R				R											
MVBC Acc.:	rw				rw											

Master Frame Register Duplicate Exception (MFRE): Address 0yF9CH

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$F\text{-}Code_{3..0}$				$Addr_{11..0}$											
Init. Value:	all 0				all 0											
CPU Acc.:	R				R											
MVBC Acc.:	w				w											

Symbol	Description
$F\text{-}Code_{3..0}$	Function Code. See "Appendix A: Function Code Summary" for details.
$Addr_{11..0}$	Address as specified in the Master Frame. It depends on the F-Code: 0 - 4: Logical Address 8,12,14,15: Device Address 9: Parameters 13: Device Group Address

Table 2.20: Master Frame Registers (MFR, MFRE)

Attention: Whenever messages are received, the MFR is loaded with the first word of the Message Data. The first word contains the destination Device Address and the Communication Mode bits.

Attention: At EMF and DMF, the MFRE contains the previous Master Frame and not the one which caused the error.

2.9.6.5 Telegram Error Recordkeeping

In order to obtain quantitative results regarding the quality of the bus, the number of frames sent and received are recorded in the Frame Counter Register (FC). The number of erroneous frames (or Reply Timeouts) sent or received are recorded in the Error Counter Register (EC). When 65,535 frames have been received, the interrupt

"Frames Evaluated Interrupt" (FEV)

will be asserted. The user can retrieve the error count from the EC and reset the FC. The EC will be frozen when the FC has reached the maximum value of 65,535 (216-1). FC and EC will remain unchanged during event polls since these polls allow collisions and Reply Timeouts.

The following frames affect FC and EC:

- All Master Frames (including F-Codes 9, 13 and 14)
- Slave Frames (or Reply Timeouts) followed after Master Frames with all F-Codes except 9, 13 and 14.

The following frames do not affect FC and EC:

- Slave Frames followed after valid Master Frames with F-Code 9, 13 and 14: Arrival of good and erroneous frames (collisions) and timeouts.

Frame Counter (FC): Address 0yF90H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$FC_{15..0}$															
Init. Value:	all 0															
CPU Acc.:	RW															
MVBC Acc.:	rw															

The Error Counter counts the number of erroneous frames as described above. This value shall be reset by the user.

Error Counter (EC): Address 0yF94H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$EC_{15..0}$															
Init. Value:	all 0															
CPU Acc.:	RW															
MVBC Acc.:	rw															

2.9.6.6 Error Models

The table shown on the next page illustrates the error models and the actions taken:

Brief description of the column headers:

Counter: FC Frame Counter is incremented. Not affected during Event Polls (F-codes 9,13,14).

Counter: EC Error Counter is incremented. Not affected during Event Polls (F-Codes 9, 13, 14).

TACK: Transfer Acknowledge set to FFFFH (PCS Word 2)

CRC, SQE, ALO, BNI, TERR, STO:
Telegram Status Bits affected (PCS Word 2) if active port is processed

LS: Line switched (if RLD=0, SLM=0). No line switchover takes placing during Event Polls.

VP: Valid Page Pointer (PCS Word 2) toggled or unchanged if active port is processed

Interrupts: See section 2.9.15.

Error Situation, Possible Causes	Counter		Influences on the PCS, Word 1 (Telegram Status)								PCS Wd 1	Interrupts
	FC	EC	TACK	CRC	SQE	ALO	BNI	TERR	STO	LS	VP	(if enabled)
Valid Master Frame (no DMF):	+1	+0	-	-	-	-	-	-	-	No	-	- [3]
Valid Slave Frame (no DSF):	+1	+0	FFFFH	0	0	0	0	0	0	No	toggle	SFC+DTIi [2]
Entire Telegram (Valid MF + SF):	+2	+0	FFFFH	0	0	0	0	0	0	No	toggle	SFC+DTIi [2]
Erroneous Master Frame (EMF) detected (delimiter OK):	+1	+1	-	-	-	-	-	-	-	Yes	-	EMF [3]
Erroneous Slave Frame (ESF) detected (delimiter OK):												
Due to Signal Quality Error (SQE)	+1	+1	unchgd	0	1	0	1	1	0	Yes	See [1])	ESF
Due to Manchester Code Violation	+1	+1	unchgd	0	0	0	1	1	0	Yes	See [1])	ESF
Due to Invalid Check Sequence (CRC)	+1	+1	unchgd	1	0	0	1	1	0	Yes	See [1])	ESF
Frame Size Error (but correct transfer over the bus)	+1	+1	unchgd	0	0	0	1	1	0	Yes	unchgd	ESF
External Noise (notice combination of errors allowed)	+1	+1	unchgd	0 or 1	0 or 1	0 or 1	1	1	0	Yes	unchgd	ESF
Reply Timeouts (RTI):												
Late Slave Frame (SF is treated as DSF afterward)	+1	+1	unchgd	0	0	0	0	0	1	Yes	unchgd	RTI
Timeout due to missing Slave Frame (Silence)	+1	+1	unchgd	0	0	0	0	0	1	Yes	unchgd	RTI
Noisy bus, no delimiter detectable	+1	+0	unchgd	0	0	0 or 1	1	0	1	Yes	unchgd	RTI
Reception of two consecutive Master Frames (DMF):												
1st frame is ignored, 2nd frame is processed	+1	+1	-	-	-	-	-	-	-	No	unchgd	DMF [3]
Reception of two consecutive Slave Frames (DSF):												
1st frame is processed as good Slave Frame	+2	+0	FFFFH	0	0	0	0	0	0	No	toggle	SFC+DTIi [2]
2nd SF causes error	+1	+1	-	-	-	-	-	-	-	No	unchgd	DSF
Collisions:												
Strong overlapping: Both delimiter and data garbled:	+1	+1	unchgd	0	0	0 or 1	1	0	1	Yes	unchgd	RTI
Weak overlapping: 1 Delimiter OK, data and CRC garbled	+1	+1	unchgd	0 or 1	0 or 1	0 or 1	1	0 or 1	0	Yes	unchgd	ESF
No overlapping: 2 consecutive Slave Frames: At 1st Frame:	+1	+0	FFFFH	0	0	0	0	0	0	No	toggle	SFC+DTIi [2]
At 2nd Frame: DSF occurs	+1	+1	-	-	-	-	-	-	-	No	unchgd	DSF
Mixup: 1 good SF followed by 1 erroneous SF: At 1st Frame:	+1	+0	FFFFH	0	0	0	0	0	0	No	toggle	SFC+DTIi [2]
At 2nd Frame: ESF+DSF occur	+1	+1	-	-	-	-	-	-	-	Yes	unchgd	DSF+ESF
Bus Timeout (BTI):	+0	+0	-	-	-	-	-	-	-	Yes	unchgd	BTI
Frame spacing too small: (less than 500 ns)	-	-	Previous and current frames are treated as one large frame with illegal frame size, see *Frame Size Error*									

[1] Toggles if TWCS=='1' and an entire frame has been received (no frame size error)

[2] DTIi means "Data Transferred" Interrupt, see section 2.9.8.

[3] MFC "Master Frame Checked" is not included since it applies to sending Master Frames only. See section 2.9.8.

Table 2.21: Telegram Analysis Results

2.9.7 Status Control Register (SCR)

The Status Control Register contains MVBC configuration information. This register is used to force the MVBC into various operation and test modes. The test modes allow extensive testing of MVBC functionality without interfering with MVB traffic.

Status Control Register (SCR): Address 0yF80H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	IM	QUIET	MBC	-	$TMO_{1..0}$		$WS_{1..0}$		$ARB_{1..0}$		UTS	UTQ	MAS	RCEV	$IL_{1..0}$	
Init. Value:	0	0	0	0	01		11		00		0	0	0	0	00	
CPU Acc.:	RW	RW	RW	RW	RW		RW		RW		RW	RW	RW	RW	RW	
MVBC Acc.:	r	r	r	r	r		r		r		r	r	r	r	r	

Symbol	Description
IM	Intel/Motorola Mode (Assures correct order of data transmitted/received) 0 Motorola Mode (Big Endian type CPU connected to MVBC, i.e. Motorola 680x0 series) 1 Intel Mode (Little Endian type CPU connected to MVBC, i.e. Intel 80x86 series) IM affects byte order when transferring non-numeric data (see section 2.8.3.1, PCS bit "NUM" inactive) over the MVB and byte order of Port Index Tables when using MCM=0 or 1 (see section 2.8.2).
QUIET	Disables data transmission immediately (Jabber-Hold Protection), used by Encoder If the software got information that its own MVBC acts as a permanent transmitter (jabber activity), the Encoder can be shut off by activating this bit. The MVBC will still be able to receive frames and update sink ports. 0 Transmitter is active 1 Transmitter is shut off
MBC	Message Broadcall: Used by Main Control Unit 0 Messages (F-Code=12) are received only if target address matches with own DA. 1 All messages with valid CM values are received (suitable for bus monitoring).
$TMO_{1..0}$	Timeout Coefficient: Used by Telegram Analysis Unit 0 0 21.3 µs (50% less) 0 1 42.7 µs (Default and Initial value) 1 0 64.0 µs (50% more) 1 1 85.4 µs (100% more)
$WS_{1..0}$	Minimum number of Waitstates: Used by Traffic Memory Controller for all TM accesses made by host CPU and MVBC, as well as accesses to Internal Registers 0 0 0 waitstates inserted 0 1 1 waitstates inserted 1 0 2 waitstates inserted 1 1 3 waitstates inserted

Continued on next page.

Continued from previous page.

$ARB_{1..0}$	Arbitration Strategies: Used by the Arbitration Controller 0 0 The CPU has the highest priority 0 1 The MVBC has the highest priority 1 0 Traffic Memory is locked to allow CPU accesses only 1 1 The CPU has the highest priority except when contents of Transmission or Receive Buffer of the MVBC are transferred from/to the Traffic Memory.
UTS	Use Test Sink Port (TSNK) 1 The MVBC checks TSNK and no other port when a Slave Frame is received. Used for testing and diagnostics, i.e. transferring Process Data over internal loop-back. 0 Normal operation Attention: TSNK is equivalent to MSNK
UTQ	Use Test Source Port (TSRC) 1 The MVBC checks TSRC and no other port when a Slave Frame is sent. Used for testing and diagnostics, i.e. transferring Process Data over internal loop-back. 0 Normal operation Attention: TSRC is equivalent to MSRC
MAS	MVBC is permitted to send Master Frames 1 The MVBC is allowed to send Master Frames 0 The MVBC is not allowed to send Master Frames (Slave device only) Attention: If MAS is set from '1' to '0' while the MVBC is busy sending a manually initiated Master Frame (SMFM bit, see MR, section 2.9.8.1), the current Master Frame will still be sent. If automatic sending is in progress (SMFA or SMFT), then the current Master Frame will be transmitted and sending the remaining frames will be suspended until MAS changes back to '1'. Canceling sending Master Frames is accomplished by writing '1' to the CSMF-Bit in the MR.
RCEV	Receive Event Frames (allow participation) 1 The MVBC reacts to all MF containing Event Polls (F-Code=9,13,14). To avoid continuous message transmission, the bits EA0 and EA1 (See Master Register MR) shall be set to '0' and the event ports EF0 and EF1 shall be declared as sinks or passive before activating RCEV. 0 The MVBC does not react to Event Polls. Attention: If RCEV is set from '1' to '0' while the MVBC is participates on event arbitration, then participation will be suspended until RCEV changes back to '1'. Canceling participation on event arbitration is accomplished by writing '1' to the corresponding EC-bits found in the MR.
$IL_{1..0}$	Initialization Level 0 0 Software reset of MVBC, all relevant bits are set to initial state. Data communication stops. If a frame is still transmitted, then the transmission will be completed. See Appendix G: Reset Behavior. 0 1 Configuration Mode: The MVBC does not participate at or listen to MVB traffic. The timeout mechanisms in the Telegram Analysis Unit stay disabled. This level is useful to configure MVBC registers without affecting bus traffic. 1 0 Selftest Mode: In this mode, local loop-back between the two outgoing and incoming serial bus lines is established. In this mode, the pins OC and SF remain inactive in order to prevent interfering bus traffic. 1 1 Full operation mode

Table 2.22: Status Control Register (SCR)

2.9.8 Main Control Unit (MCU)

The Main Control Unit (MCU) orchestrates all data transfers between the MVB and the Traffic Memory. The MCU handles following functions:

- Slave Functions:

  Received Master Frames are checked for their F-Code and address and a decision is made whether ports are to be processed or not. If a source port is encountered, then data is transmitted. If a sink port is encountered, then the MVBC will store the contents of the received Slave Frame in the port. All F-Codes, which are defined in the IEC TCN Standard [1], are supported.

- Master Functions:

  The MCU can transmit individually assigned Master Frames or Master Frames retrieved from Master Frame Tables. The Master Frames which are transmitted here will also be received and processed by the MVBC in the same manner as if they originated from a remote MVB participant. This feature allows one CPU to run both Bus Administrator software and application software which depend on the MVB. The MAS-bit in the SCR must be active in order to pursue Master Functions.

- Event Arbitration Mechanism

  The MCU is in charge of handling event arbitration.

- Interrupts

  The MCU can assert following interrupts:

"Master Frame Checked"	(MFC)
"All Master Frames Transmitted"	(AMFX)
"Data Transfer Interrupts"	(DTI1..DTI7)

  MFC is asserted after the MCU has retrieved a MF from the TM and passed to the Transmit Buffer in order to send it. AMFX indicates that a Master Frame Table has been processed entirely. $DTI_{1..7}$ are user-definable interrupts which occur after the data transfer and after the ports have been processed.

The MCU is closely coupled with the Status Control Register (SCR), Telegram Analysis Unit and Sink Time Supervision. Details about communication behavior are handled in chapter 3.

The MCU provides four control registers which are described in the following sections:

- Master Register (MR)
- Secondary Master Register (MR2), lower 8 bits only
- Dispatch Pointer Register (DPR)
- Secondary Dispatch Pointer Register (DPR2)

2.9.8.1 Master Registers (MR, MR2)

The Master Register (MR), along with the Dispatch Pointer Register (DPR) provides dashboard type functions to pass instructions to the MCU. It is organized in such a way so multiple independent software tasks can access this register without interfering among each other. Its principal functions include

- Signing up and canceling participation on Event Arbitration
- Dispatching Master Frames (Bus Administrator function)
- BUSY-Indicator (MCU is busy processing ports)

Master Register (MR): Address 0yFA0H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$PAR_{1..0}$		$EA_{1..0}$		$EC_{1..0}$		BUSY	CSMF	$SMF_{1..0}$		SMSM	$C_{4..0}$				
Init. Value:	00		00		00		0	0	00		0	all 0				
CPU Acc.:	R		RW1		R0W1		R	R0W1	RW1		RW1	RW				
MVBC Acc.:	rw		rw0		r		w	rw0	rw0		rw0	rw				

Symbol	Description
$PAR_{1..0}$	Indicates that MVBC is participating on Event Arbitration; Bit 1 = Related to Event type 1 Bit 0 = Related to Event type 0 0 No participation 1 Participation
$EA_{1..0}$	Event Announced by software Bit 1 = Related to Event type 1 Bit 0 = Related to Event type 0 0 No effect 1 Event announced. The MVBC participates at the next event round for selected Event Type
$EC_{1..0}$	Event Announcement Cancellation Bit 1 = Related to Event type 1 Bit 0 = Related to Event type 0 0 No effect 1 Cancels announced event and any form of participation. EA*i* and PAR*i* will be cleared. (*i* = selected Event Type) Read: Always zero
BUSY	Busy Sending Master Frame and waiting for its Slave Frame 0 Indicates that no data transfer is in progress and the MVBC is ready to send next Master Frame immediately. 1 MVBC is not yet ready to send next Master Frame. Either, the previous Master Frame has not yet been sent, or the Slave Frame has not yet arrived, or automatic dispatching is still in progress. If no Slave Frame is received, the MVBC times out and clears this bit. Exceptions: If, for any technical problems, the transmitted Master Frame does not arrive at the Decoder, then the Telegram Analysis Unit will automatically assert BTI after 1.3 ms. This signal will clear the BUSY-Bit. On the other hand, if the Master Frame is corrupted, then the EMF (Erroneous Master Frame) Interrupt is asserted and the BUSY-Bit is cleared automatically.
CSMF	Cancel Sending Master Frames 0 No action 1 Clears pending SMFE, SMFT, SMFM and SMFA requests. This mechanism cancels any scheduled MF-transmission or MF-transmission already in progress. If a telegram is transferred meanwhile, then the MVBC completes the transfer and sends no MF afterward. Read: Always zero

Continued on next page.

Continued from previous page.

Symbol	Description
$SMF_{1..0}$	Send Master Frames Automatically from MF-Tables (3 options): Write access: Actions 0 0 No action 0 1 SMFA issued (Automatic Mechanism): Start processing MF-Table pointed by the DPR immediately. The bits C4..0 indicate table size. Attention: This bit can be activated by the MVBC when the Universal Timer 1 reaches zero (given it is active) and SMFT is set to '1', or when the contents of MR2 are moved to MR when the AMFX-Interrupt occurs. End of MF-Table is indicated with the AMFX-Interrupt. The interrupt leads to a transfer from DPR2 to DPR and from MR2 to MR in order to process the next MF-Table if assigned. 1 0 SMFT issued (Timed Mechanism): Start processing MF-Table pointed by DPR at the moment the Universal Timer 1 reaches zero (precondition: Timer 1 must be active). At that moment, SMFT changes to SMFA and processing starts as if an SMFA has been issued. This command is suitable for precisely timed MF-distribution. 1 1 SMFE issued (Empty MF-Table declared): The MVBC transmits no Master Frames, but waits until the Universal Timer 1 reaches zero. At that instance, the interrupt AMFX is issued. This interrupt leads to an Internal Register transfer from MR2 to MR and DPR2 to DPR in order to process the next MF table. It is useful to handle empty time slots. Attention: Only one of the three request can be submitted. A different request will be ignored until the request has been processed or canceled. Read access: Information on type of request submitted: 0 0 No request pending 0 1 SMFA requested 1 0 SMFT requested 1 1 SMFE requested
SMFM	Send Master Frame Manually Write access: 0 No action 1 Send Master Frame manually. The Master Frame body to be sent will be retrieved from TM location 0yE00H (Master Frame Slot). The automatic Master Frame Dispatcher is still active, then the manual Master Frame will be sent after the Master Frame Table has been completed. Read access: Information on request submitted 0 No SMFM request pending 1 SMFM requested

Continued on next page.

Continued from previous page.

Symbol	Description
$C_{4..0}$	Size of Master Frame Table. Write access: 1..31 Specifies Master Frame Table of size 1..31 respectively 0 Specifies Master Frame Table of size 32 Attention: Use SMFE to specify an empty Master Frame Table (0 entries). Attention: This value will only apply if SMFA or SMFT is requested with the same write access. $C_{4..0}$ will also be updated to the value written if SMFE is requested, but in this case, the value is ignored. All other accesses, where $SMF_{1..0}$ equals '00' will keep $C_{4..0}$ unchanged. Read access: Number of Master Frames to be sent out. With every Master Frame sent, the value will be decremented by 1. If 0 is specified for 32 Master Frames, then the first decrement leads to a value of 31.

Table 2.23: Master Register (MR)

The secondary Master Register (MR2) is used along with the corresponding Dispatch Pointer Register (DPR2) to place *advance requests* to send Master Frames while MF-transmission is in progress. Details are handled in section 3.10. At the moment the AMFX Interrupt occurs, then the contents of MR2 are transferred to MR7..0 and MR2 returns to zero. The upper halfword of MR2 is not used.

Secondary Master Register (MR2): Address 0yFA4H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	-								$SMF_{1..0}$		SMSM	$C_{4..0}$				
Init. Value:	-								00		0	all 0				
CPU Acc.:	all 0								RW1		RW1	RW				
MVBC Acc.:	-								rw0		rw0	rw				

See MR (Table 2.23) for bit descriptions.

2.9.8.2 Dispatch Pointer Register (DPR, DPR2)

The DPR contains the pointer to address the MF-Table when a MF-Table is to be processed. The table size is specified in MR.

Dispatch Pointer Register (DPR): Address 0yFA8H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$DPR_{15..2}$														-	
Init. Value:	0														all 0	
CPU Acc.:	RW														-	
MVBC Acc.:	rw0														-	

The contents of the Secondary Dispatch Pointer Register (DPR2) are transferred to DPR when the AMFX interrupt occurs.

Secondary Dispatch Pointer Register (DPR2): Address 0yFACH

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$DPR2_{15..2}$														-	
Init. Value:	0														all 0	
CPU Acc.:	RW														R0	
MVBC Acc.:	rw0														-	

Symbol	Description
$DPR_{15..2}$ $DPR2_{15..2}$	Dispatch and Secondary Dispatch Pointer Value The start address to the Master Frame Tables are always aligned in a way so the lower two bits are zero. If the whole 16-bit word ($DPR_{15..0}$) is considered, then this whole word represents the pointer to the Master Frame Table, in units of 4 bytes. See also section 2.8.7 for more information on the pointer format.

2.9.8.3 Queue Management

The Queue Management is responsible for Message Queues. Queues are referenced when the QA-bit in the PCS is active. The queue is retrieved by accessing the Queue Descriptor Table. Then, a Linked List Record (LLR) is retrieved in order to access the Message Data block and the pointer to the next LLR.

Following interrupts are available to report queued data transfers and exceptions: If all records of one of the two Transmit Queues have been sent, then one of the following interrupts may occur:

"Transmit Queue 0 Complete" (TQ0C)

"Transmit Queue 1 Complete" (TQ1C, along with TQ0C)

If TQ1C is asserted, then TQ0C is asserted, too. Reason: The MCU checks Queue 0 before Queue 1. If both Transmit Queues are empty or do not exist, then the interrupt

"Transmit Queue Exception" (TQE)

is asserted. The MVBC cannot send any data. If the Receive Queue is full after receiving the message, then the interrupt

"Receive Queue Complete" (RQC)

is asserted. However, if the queue is already full before the reception, then the data gets lost and following interrupt occurs:

"Receive Queue Exception" (RQE).

2.9.9 Device Address Read & Store Unit

The Device Address is supplied via the 12-bit Device Address ($DA_{11..0}$) pins into the MVBC. This Device Address must stay constant since the MVBC references these pins throughout its operation. The Telegram Analysis Unit compares the Device Address with the address given in the Master Frame and passes the results to the MCU. The effective Device Address can be read from the Device Address Override Register (DAOR).

An override mechanism allows the user to let a software-defined value override the original Device Address. Overriding is enabled if a protection key is written to the Device Address Override Key (DAOK).

Enabling DA override:

- Write a value to DAOR. When DAOR is read, then the contents of the DA11..0 pins are still returned.
- Write 94H to DAOK. Reading DAOR now returns the new DA.

Disabling DA override:

- Write 49H to DAOK. Reading DAOR now returns the hardware DA.

2.9.9.1 Device Address Registers (DAOR, DAOK)

Device Address Override Register (DAOR): Address 0yFD8H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	-				$DA_{11..0}$											
Init. Value:	all 0				all 0											
CPU Acc.:	R0				RW											
MVBC Acc.:	-				r(w)											

Symbol	Description
$DA_{11..0}$	Device Address Write access: New device address is written into DAOR, regardless of value of DAOK. Read access: If SW-overriding is not set, then the written device address stays hidden and the effective hardware device address (supplied via pins $DA_{11..0}$) is issued. If SW-overriding is set, then the written device address is returned.

Table 2.24: Device Address Override Register (DAOR)

Device Address Override Key (DAOK): Address 0yFDCH

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	-								$K_{7..0}$							
Init. Value:	all 0								00H (disabled)							
CPU Acc.:	R0								RW							
MVBC Acc.:	-								r							

Symbol	Description
$K_{7..0}$	Override Key Write access: 94H Enables SW-overriding on Device Address: $DA_{eff} \Leftarrow$ DAOR(written) 49H Disables SW-overriding on Device Address: $DA_{eff} \Leftarrow$ DA(Input Pins) other: No effect Read access: FFH Indicates DA overriding enabled 00H Indicates DA overriding disabled

Table 2.25: Device Address Override Key (DAOK)

2.9.10 Address Logic

The Address Logic consists of two parts:

- Address Encoder
- Address Decoder

The Address Encoder uses the tables summarized in section 2.8.9, the parameters issued by the MCU and the Memory Configuration Register to generate addresses for MCU accesses to the Traffic Memory.

The Address Decoder analyzes the addresses issued by the CPU to the MVBC and Traffic Memory. If the address points to an Internal Register inside the Service Area, then the corresponding Internal Register is selected. Otherwise, the Traffic Memory Controller will be instructed to forward the access control signals to the TM. The address ranges for the Internal Registers are summarized in section 2.8.6.5.

2.9.10.1 Memory Configuration Register (MCR)

Memory Configuration Register (MCR): Address 0yF84H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$VERSION_{4..0}$					-				$MO_{1..0}$		$QO_{1..0}$		$MCM_{2..0}$		
Init. Value:	0 0 0 0 0					all 0				all 0		all 0		all 0		
CPU Acc.:	R0					R0				RW		RW		RW		
MVBC Acc.:	w0					-				r		r		r		

Symbol	Description
$VERSION_{4..0}$	MVBC Version 0H MVBC 01
$MO_{1..0}$	Master Frame Table Offset (affects MF-Tables) 0 0 Range 00000H - 3FFFFH 0 1 Range 40000H - 7FFFFH 1 0 Range 80000H - BFFFFH 1 1 Range C0000H - FFFFFH
$QO_{1..0}$	Queue Offset (a selects LLR's and Queue Data blocks) 0 0 Range 00000H - 3FFFFH 0 1 Range 40000H - 7FFFFH 1 0 Range 80000H - BFFFFH 1 1 Range C0000H - FFFFFH
$MCM_{2..0}$	Memory Configuration Mode 000 Mode 0 (16 K Bytes) 001 Mode 1 (32 K Bytes) 010 Mode 2 (64 K Bytes) 011 Mode 3 (256 K Byte) 100 Mode 4 (1 M Byte) Attention: All other modes are illegal.

Table 2.26: Memory Configuration Register (MCR)

Attention: If the user intends to move the Service Area by modifying MCR, the relocation will take place 41 ns after the write access has been completed (WR\ returning from '0' to '1'). A brief pause (1 clock cycle or 1 instruction) should be introduced in order to avoid accesses in the old MCM.

2.9.11 Arbitration Controller

The Arbitration Controller governs access rights to Traffic Memory between the CPU and the MVBC. The Arbitration Strategy is configurable with the ARB-bits found in the SCR. The Arbitration Controller assures that the last TM access will be completed properly before the switch (from CPU to MVBC or vice versa) takes place.

The Arbitration Controller operates in one state at a time:

State	Description and Signals
CPU State	CPU accesses TM or Internal Registers. This is the initial state for class 2/3/4 mode. Signals: A_EN\ = 0 (active), address bus is inactive.
MVBC State	MVBC accesses TM. This is the initial state for class 1 mode. Signals: A_EN\ = 1 (active), address bus is active.

Table 2.27: Arbitration Modes

The following Arbitration Strategies are supported:

$ARB_{1..0}$	Description and Signals
0	CPU has highest priority all time Every time the CPU requests to access the TM or Internal Registers by activating TM_REQ_CPU\, the Arbitration Controller will switch from MVBC to CPU Mode immediately. The CPU can interrupt a series of subsequent TM accesses[1] made by the MVBC (i.e. transferring 16 data words). The MVBC cannot suspend a series of subsequent accesses made by the CPU.
1	MVBC has highest priority all time Every time the CPU requests to access the TM or Internal Registers, it must wait until the MVBC has completed accessing the TM. This does even apply if the MVBC makes a series of subsequent TM accesses. If the CPU makes a series of subsequent accesses, and the MVBC requests the TM, then the accesses will be suspended until the MVBC has finished.
2	CPU locks Traffic Memory for Memory Test Using this strategy, the MVBC may not make any accesses to TM. This strategy must not be active for a too big time period if the MVBC handles data communication at that time. This strategy guarantees full data consistency on the TM, so on-line background tasks, which run RAM-test software, can make use of this strategy.
3	Mixed Priority Strategy CPU has highest priority (Strategy 0 applies) except when the MVBC transfers data between the Traffic Memory and Transmit as well as Receive Buffer (Strategy 1 applies). CPU accesses cannot break apart any such transfers. This strategy has been introduced to guarantee data consistency while data is sent or received.

[1] The term *subsequent accesses* means holding TMCS\ active while reading or writing with TMRD\ or TMWR\. The CPU must hold TM_REQ_CPU\ active during multiple accesses in order to perform subsequent accesses.

Table 2.28: Arbitration Strategies

Attention: If strategy 0 is selected, then the user must be aware not hold the TM by keeping TM_REQ_CPU\ active over long time periods. In this case, data may arrive from the MVB and cannot be stored in the TM. This may result to loss of received data.

If strategy 2 is selected during normal system operation in order to run on-line RAM tests in a background task, the lock time must be kept to a minimum (i.e. for max. 6 consecutive TM accesses).

Whenever an arbitration switch takes place, A_EN\ and the address or data pin drivers will never be active at the same time. A guard time of 1-2 clock cycles is introduced to disable A_EN\ and enable the drivers and vice versa.

2.9.11.1 Data Transfer Mode

If Arbitration Mode 3 is selected, then the MVBC calls for high priority when following accesses to the TM are made:

- Data transfer (Data Area, queued data), including accesses to PCS words which take place immediately (no pauses) before and after the data transfer
- Queue pre- and postprocessing
- One access pair for sink-time supervision: read + write TACK

2.9.11.2 Voluntary Action

If the MVBC does not operate in class 1 mode, and has completed a series of TM accesses, it will instruct the Arbitration Controller to commit a voluntary action, namely to return to CPU mode. This mechanism helps saving access time since the CPU does not need to wait until a switchover has taken place.

2.9.12 Traffic Memory Controller (TMC)

The Traffic Memory Controller TMC operates closely with the Arbitration Controller and Address Logic. The Traffic Memory Controller handles the following accesses:

- CPU accesses to Traffic Memory
- CPU accesses to Internal Registers inside MVBC
- MVBC accesses to Traffic Memory

The principal functions of the TMC are:

- Forwarding CPU memory access signals to the TM when the CPU accesses the TM
- Processing CPU signals when accessing Internal Registers
- Generating control signals to TM when the MVBC accesses the TM
- Maintaining minimum specified waitstate count for all accesses
- Proper control signal handling while the Arbitration Controller switches between CPU and MVBC mode.

Timing Diagrams:

The TMC follows the specifications given on the attached timing diagrams (see section 7.2).

2.9.12.1 CPU Signals

If the Arbitration Controller resides in CPU mode, then the memory control signals originating from the CPU are directly forwarded to the Traffic Memory, given the address does not point to the address space reserved for Internal Registers:

Always:	TM_REQ_CPU\	→	TMCS\ and D_EN\
If TM_REQ_CPU\ is active:	RD\	→	TMRD\
	WR\	→	TMWR\ and DIR
	A_EN\	→	is always active in CPU Mode
	RDY\	→	0 → 1 immediately at begin of access
			1 → 0 when access has been completed

RDY\ exhibits a "Normally Ready" behavior: The MVBC is *ready* until an access is made. The MVBC becomes *not ready* until the data is available. When the following conditions are met, then the MVBC becomes *ready* again:

- The CPU has waited until the Arbitration Controller switched to CPU mode (if this has not yet taken place)
- The minimum number of waitstates (specified by WS1..0, see SCR) has elapsed
- TMRDY\ has turned active (not applicable if Internal Registers are accessed).

TMRDY\ must conform with the "Normal Ready" behavior. If the incoming Ready-signal TMRDY\ is tied to zero, then the outgoing RDY\-signal is generated explicitly by the waitstate counter in order to abide the waitstate requirements. If TMRDY\ is connected to a memory system (i.e. DRAM controller, Dual-Port RAM), then the RDY\-signal will be forwarded to the CPU as long the minimum waitstate count has elapsed.

If TM_REQ_CPU\ is inactive, then RD\ and WR\ do not affect the MVBC nor the Traffic Memory. If Internal Registers are accessed, then TMCS\, TMRD\ and TMWR\ remain passive. The MVBC recognizes the access, performs the necessary Internal Register read or write operation, and obeys the specified waitstate requirements.

Following signaling methods are legal to start read and write accesses:

TM_REQ_CPU\	RD\	WR\	Access Started	TMCS\ [1]	TMRD\ [1]	TMWR\ [1]	D_EN	DIR\
1	X	X	No access	1	1	1	1	1
↓	1	1	Chip select	↓	1	1	↓	1
0	↓	1	Read access	0	↓	1	0	1
↓	↓	1	Read access	↓	↓	1	↓	1
↓	0	1	Read access	↓	↓	1	↓	1
0	1	↓	Write access	0	1	↓	0	↓
↓	1	↓	Write access [2]	↓	1	↓	↓	↓
↓	1	0	Write access [2]	↓	1	↓	↓	↓
0	↓	0	Illegal	0	-	-	0	-
0	0	↓	Illegal	0	-	-	0	-
0/↓	↓	↓	Illegal	0	-	-	0	-

[1] Inactive if Internal Registers are accessed
[2] Activating TMCS\ and TMWR\ simultaneously may violate setup timing specifications of available RAM chips. Consult the data sheets first.

Table 2.29: Starting TM Accesses

If the CPU initiates an access and MVBC mode is still effective, then the TMC waits until the arbitration switch has taken place. During the arbitration switch, the MVBC guarantees that TMCS\ and D_EN\ become active 1 clock cycle before TMRD\ or TMWR\ becomes active.

The access to TM (or Internal Registers) is complete when RDY\ returns from '1' back to '0'. Following signaling methods are legal to stop or cancel read and write accesses:

TM_REQ_CPU\	RD\	WR\	Access Started	TMCS\ [1]	TMRD\ [1]	TMWR\ [1]	D_EN	DIR\
↑	0/↑	1	Stops read access					
0	↑	1	Stops read access					
↑	1	0/↑	Stops write access					
0	1	↑	Stops write access					
0	↑	↓	Illegal [2]					
0	↓	↑	Illegal [2]					

[1] Inactive if Internal Registers are accessed
[2] See restrictions listed below

Table 2.30: Stopping TM Accesses

Restrictions:

- Between any two accesses made to the Traffic Memory, the CPU must stop the first access according to the table shown above and wait for at least 41 ns (1 clock cycle) before the next access can be started.
- RD\ must not be activated while deactivating WR\ or vice versa during the period TM_REQ_CPU\ is active.
- A block access consisting of multiple read or write accesses can be realized by holding TM_REQ_CPU\ active while all accesses are performed with toggling RD\ and WR\. The same block access may cover Traffic Memory and Internal Registers. It is forbidden to keep RD\ or WR\ active while changing the address every few clock cycles. This access mechanism exists in common RISC processors and must be deactivated for the TM.

2.9.12.2 Waitstate Generator

The waitstate generator is available to handle slower or variable-timed Traffic Memory RAMs (i.e. DPRAMs) or slow data paths between CPU and MVBC. The waitstates affect

- CPU accesses to Traffic Memory
- CPU accesses to Internal Registers inside MVBC
- MVBC accesses to Traffic Memory

The minimum number of waitstates can be configured with the WS-bits found in the SCR. Allowed values are 0, 1, 2 and 3 waitstates. Each waitstate means prolonging the access by one 24 MHz clock cycle. The MVBC requires 3 (or 6) clock cycles to access the TM if 0 (or 3) waitstates are configured.

For accesses to the TM, the number of waitstates can be extended to 64 cycles by holding TMRDY\ at '1'. The limit of 64 cycles is necessary since Slave Frame data must be transmitted within 4 µs after the Master Frame has been received. It also prevents deadlock situations at the CPU or MVBC. When this limit has been reached, the MVBC assumes that it accessed a bad memory location and completes the access sequence, regardless if the access turned out successfully or not.

Attention: 64 Waitstates is acceptable for data sources. For sinks however, the overall time required to make all write accesses must not exceed the duration of one 16-bit frame. Otherwise, the next Master Frame arrives while the MVBC is still busy processing the last port.

Attention: In Class 1 Mode, it is legal to tie the TMRDY\-signal to +5 V if the attached peripherals require a very long access time.

The $WS_{1..0}$ bits, in conjunction with the TMRDY\ pin shall be used as follows to configure the number of waitstate to be generated:

$WS_{1..0}$	TMRDY\	Number of waitstates inserted	TMRDY\	Number of waitstates inserted
00	0	0	1	64 or at least 0 if TMRDY\→0
01	0	1	1	64 or at least 1 if TMRDY\→0
10	0	2	1	64 or at least 2 if TMRDY\→0
11	0	3	1	64 or at least 3 if TMRDY\→0

[1] The minimum number of waitstates is always inserted regardless of the TMRDY\ level.

Table 2.31: Waitstate Settings

2.9.12.3 TM Accesses made by MVBC

While MVBC Mode is effective, the TMC will handle TM read and write accesses instructed by the MCU and sink-time supervision logic. A_EN\ and D_EN\ are held at '1' and DIR at '0'.

2.9.13 Bus Multiplexer, Data Forcing Network

The Bus Multiplexer handles internal bus traffic and routing in the MVBC. In addition, the bus multiplexer supports

- Byte swapping (needed to access CS, 8-bit Port Indexes, byte order compensation)
- Forcing Process Data

Forcing data is performed one word at a time:

- One Force Mask word is read from the Force Table
- One Force Data word is read from the Force Table
- The outgoing data is read from the Traffic Memory, forced and forwarded to the Transmit Buffer (TXB)

or

- The incoming data is retrieved from the Receive Buffer (RXB), forced and then written to the Traffic Memory.
- The remaining words are processed

The Boolean formula applied for forcing is found in section 2.8.5.

2.9.14 Class 1 Logic

The Class 1 Logic supports Class 1 Mode operation without assistance of a CPU or microcontroller. In this mode, the MVBC supports following transfers:

Ports	F-Code	Description
16 x 16-bit Process Data Ports (8 x source, 8 x sink ports; 16 x source ports if F-Code = 4)	0-4	Each port can be used to transfer 1-16 words of Process Data. Not supported: Forcing and automatic data comparison mechanism.
1 source port for Device Status Wd.	15	Device Status Report, generated by MVBC

Table 2.32: Data Transfers in Class 1 Mode

Device Address:

The unique Device Addresses, which are assigned to MVBCs operating in Class 1 Mode, must be divisible by 16 (Lower 4 bits are zero). In fact, the lower 4 bits are internally set to '0' since the corresponding input pins DA3..0 are used for different Class 1 Mode parameters.

Erroneous Data Transfers:

Class 1 Mode supports no external signaling for erroneous and missing frames. In case the MVBC receives erroneous data, no access is made to the peripheral devices in order to preserve the correct values from the previous transfers.

2.9.14.1 Process Data

The 16 Process Data ports are divided into 8 source and 8 sink ports. The MVBC is addressed if bits 11..4 of the Logical Address in the Master Frame equals to bits 11..4 of the configured Device Address. The sixteen ports are addressed with bits 3..0 of the Logical Address. Port 15 serves as a Synchro Port where the STROBE\-pin is activated at completion of every data reception.

For obvious reasons, the following accesses are not made when the MVBC operates in Class 1 Mode:

- Port Index Table (Port Index is derived from Logical Address)
- Port Control and Status Register (PCS)
  - Word 0 is substituted by internal parameters, see tables below
  - No TERR, STO, BNI, and TACK bits are written, No Valid Page (VP) bit
- Force Table, Service Area, etc.

F-Code 0 can be used to address each port individually. Using F-Code 1 (or 2, 3, 4) groups 2 (or 4, 8 or all 16) ports into one bigger port in order to access multiple ports with one telegram. This mechanism allows more efficient data transfers. The user must not use odd (or unaligned) port addresses when using nonzero F-Codes. Example: F-Code 2 with LA=004 is allowed, but not LA=005.

Table 2.33 - Process Data in Class 1 Mode, see next page.

Substitution for Port Control and Status Register:

The MVBC assumes following PCS value when accessing any Process Data ports:

Since No PCS can be obtained from the outside, the MVBC assumes following PCS image when handling transferring Process Data:

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol:	$F\text{-}Code_{3..0}$				SRC	SINK	TWCS	WA	$IE_{2..0}$			$CPE_{1..0}$		QA	NUM	FE
Value:	0000 - 0100				see [2]	see [2]	0	0	0	0	0	0	0	0	1	0

MVBC Access: Internal substitution for PCS

[1] Depending on F-Code in Master Frame

[2] The lower eight ports (Port DA...DA+7), are source ports.
The upper eight ports (Port DA+8...DA+15), are sink ports.
Exception: F-Code 4 is used.

xxYH: Address specified in Master Frame (xx = Bits 11..4 of Device Address Y = Port number)

Port Address	F-Code 0	F-Code 1	F-Code 2	F-Code 3	F-Code 4
xxFH	Sink Port F [1]	(2 words)			
xxEH	Sink Port E	Sink Port E			
xxDH	Sink Port D	(2 words)	(4 words)		
xxCH	Sink Port C	Sink Port C	Sink Port C		
xxBH	Sink Port B	(2 words)			
xxAH	Sink Port A	Sink Port A			
xx9H	Sink Port 9	(2 words)	(4 words)	(8 words)	
xx8H	Sink Port 8	Sink Port 8	Sink Port 8	Sink Port 8	
xx7H	Source Port 7	(2 words)			
xx6H	Source Port 6	Source Port 6			
xx5H	Source Port 5	(2 words)	(4 words)		
xx4H	Source Port 4	Source Port 4	Source Port 4		
xx3H	Source Port 3	(2 words)			
xx2H	Source Port 2	Source Port 2			
xx1H	Source Port 1	(2 words)	(4 words)	(8 words)	(16 words)
xx0H	Source Port 0	Source Port 0	Source Port 0	Source Port 0	Source Port 0 [2]

[1] Accessing Port 15 with F-Code activates STROBE\ for 125 ns (Synchro Port)

[2] Since F-Code 4 relates to 16 words, all 16 ports are accessed as source ports. This F-Code is suitable for multi-channel data acquisition. No sink port is available here

Table 2.33: Process Data Ports in Class 1 Mode

2.9.14.2 Device Status Report

In Class 1 Mode, the Device Status Report is generated by the MVBC. It consists of fixed parameters (10 bits), internal flags (2 bits) and user-adjustable flags (4 bits). Since No PCS can be obtained from the outside, the MVBC assumes following PCS settings when sending the Device Status Report:

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol:	$F\text{-}Code_{3..0}$				SRC	SINK	TWCS	WA	$IE_{2..0}$			$CPE_{1..0}$		QA	NUM	FE
Value:	1 1 1 1				1	0	0	0	0 0 0			0 0		0	1	0

MVBC Access: Internal substitution for PCS

Internal Contents of the Device Status Report (DSR):

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol:	Device Type				Reserved				LAA	RLD	SSD	SDD	ERD	FRC	DNR	SER
Value:	1 1 1 1				0 0 0 0				Int.	Int.	Pin [1]	Pin [1]	Pin [1]	0	Pin [1]	0

[1] Input pins: See Table 2.34.

LAA is obtained from bit 3 of the Decoder Register (DR). If a line switch took place before a Device Status Poll, then the active line after the line switch is returned. See section 2.9.4 for details.

RLD (Redundant line Disturbed, see section 2.9.4.5) is obviously read before it is cleared.

2.9.14.3 Other Substitutions

Internal Contents for Status Control Register (SCR):

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	IM	QUIET	MBC	-	$TMO_{1..0}$		$WS_{1..0}$		$ARB_{1..0}$		UTS	UTQ	MAS	RCEV	$IL_{1..0}$	
Init. Value:	0	0	0	0	Pins [1]		Pins [1]		0 1		0	0	0	0	11	

[1] Input pins: See Table 2.34

The MVBC starts up in full operational mode (IL = 3).

2.9.14.4 Class 1 Multiplexed Lines

When the MVBC operates in Class 1 Mode, different functions are assigned to some input pins and address outputs. Please note that 16 bits in the address bus are decoded as active-low chip-select (CS*i* \) signals. The remaining address bits are held at '1'. The following table summarizes these signals:

Pin		**Description**	
Class 1 Mode	**Class 2/3/4 Mode**	**Class 1 Mode**	**Class 2/3/4 Mode**
$WS_{1..0}$	$II_{1..0}$	Wait State Select bits 1..0	Interrupt Inputs 1..0
$TMO_{1..0}$	$II_{3..2}$	Timeout Coefficient 1..0	Interrupt Inputs 3..2
DNR	DA_0	Device Not Ready	Device Address bit 0
ERD	DA_1	Extended Reply Delay	Device Address bit 1
SDD	DA_2	Some Device Disturbance	Device Address bit 2
SSD	DA_3	Some System Disturbance	Device Address bit 3
($A_{19..17}$)	$A_{19..17}$	Permanently tied to 1	Address bus
CS0\	A_{16}	Chip Select bit 0	Address bit 16
CS1\ ... CS15\	$A_1..A_{15}$	Chip Select bits 1..15	Address bits 1..15

Table 2.34: Pin Reassignments for Class 1 Mode

2.9.14.5 Class 1 Peripheral Interface

In Class 1 Mode, the access mechanism from MVBC to the peripheral devices is similar to the access mechanism to the Traffic Memory in Class 2/3/4 Mode. The only difference is the decoded address bus. Very simple devices can use the CSi \-signals only to read or write ports.

Since no host CPU is present, the following signals must be kept inactive: TM_REQ_CPU\, RD\, WR\. The Interrupt Logic remains inactive since no method exists to configure the interrupt registers.

The pins DNR, ERD, SDD and SSD supply information to the Device Status Report. For DNR, the current level is used. The signals ERD, SDD and SSD are latched and will be kept until the next Device Status Report has been transmitted. After that, these bits will be cleared.

2.9.15 Interrupt Logic

The Interrupt Logic can handle up to 32 interrupts which originate from functional units inside the MVBC and from external interrupt pins. The Interrupt Logic has been designed in order to allow efficient cooperation with software. The programmer can choose whether to use fixed-priority interrupt vectors, check the bits in the Interrupt Status Register, or use a combination of both.

The Interrupt Logic consists of two parts which operate independently. Each part is equipped to handle 16 interrupts, provides their own output pins INT0\ and INT1\, and one set of four registers described below:

- Interrupt Mask Registers (IMR0, IMR1)
- Interrupt Pending Registers (IPR0, IPR1)
- Interrupt Status Registers (ISR0, ISR1)
- Interrupt Vector Registers (IVR0, IVR1)

The Interrupt Pending Register (IPR*i*) captures the interrupt request signals from the outside (II3..0 pins) and MVBC functional units. The Interrupt Mask Register (IMR*i*) is used to keep interrupt signals from penetrating from the (IPR*i*) to the Interrupt Status Register (ISR*i*) and to the Interrupt Vector Register (IVR*i*). Any active bit inside the ISR*i* will trigger the outgoing interrupt signal INT*i* \.

Attention: Each part of the Interrupt Logic also contains a hidden state variable: *frozen* or *not frozen* (released). Details about this mechanism is described in section 2.9.15.1. Freezing Interrupt Logic 0 has no effect on Interrupt Logic 1 or vice versa.

2.9.15.1 Interrupt Handling Mechanism

Following operation occurs on the falling edge of an incoming interrupt signal:

- The interrupt request *k* is registered in bit *k* in IPR*i* (notation: IPR*i* [*k*] = '1').

- If (ISR*i* is *not frozen*) then
  if (IMR*i* [*k*] = '1' (*not masked*)) then
  the interrupt is forwarded to the ISR*i* [*k*]. At the same time, IPR*i* [*k*] is reset.
  else if (IMR*i* [*k*] = '0' (*masked*)), then
  IPR*i* [*k*] is reset and the Interrupt Logic forgets the interrupt request.
  else, (if ISRi is *frozen*), then
  the value is held in the IPR*i* [*k*] until the ISRi is no longer *frozen*. If the same interrupt occurs for the second time and ISR*i* is still *frozen*, then the second interrupt is *swallowed*. Avoiding this case is a software responsibility.

- If ISR*i* [*k*] becomes active, then the outgoing interrupt signal INT*i* \ changes to '0'. At the same time, the interrupt vector is computed from the ISR*i* and made visible in IVR*i*

- The software can check the interrupts using two methods:
  a. by reading the interrupt vector from IVR*i*, or
  b. by examining the ISR*i* directly.

  Read accesses to the IVR*i* as well as to the ISR*i* will automatically *freeze* the ISR*i* and deactivate the outgoing INT*i* \-pin.

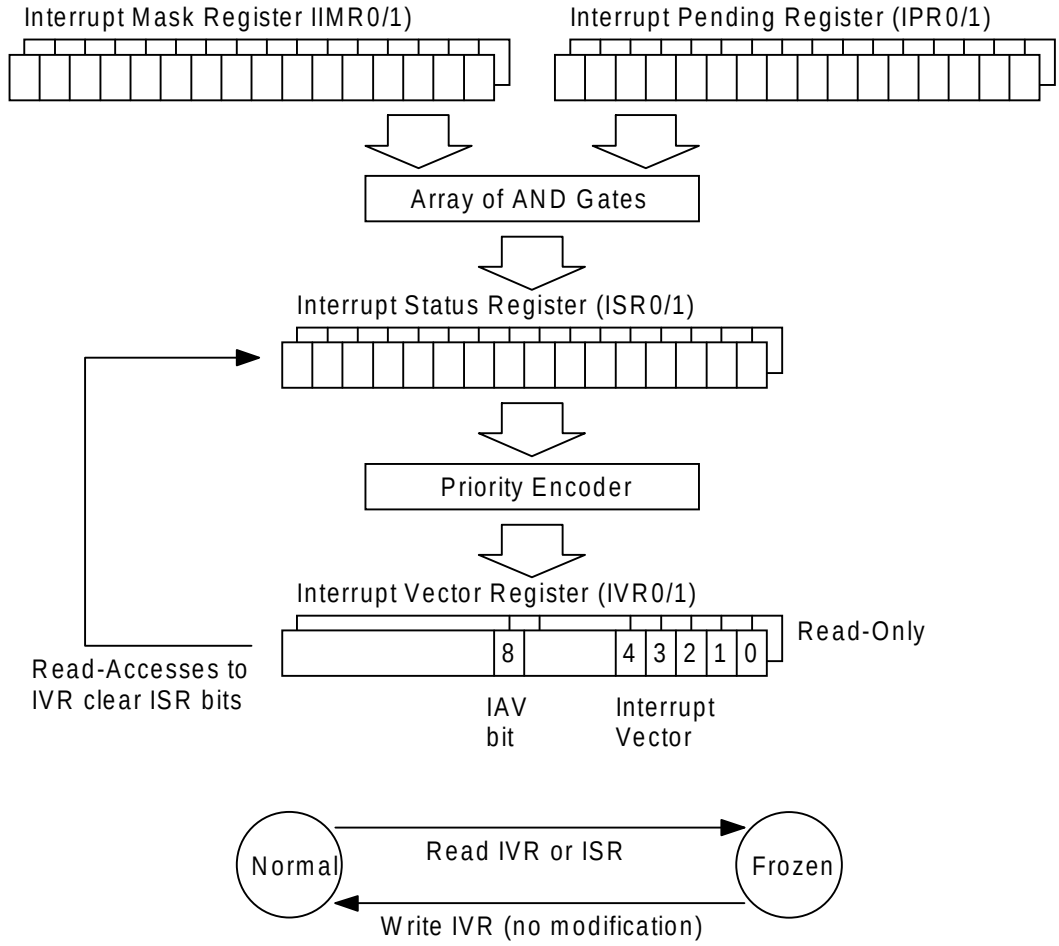
  In case (a.), the ISR*i* -bit pointed by the vector is automatically cleared when reading the interrupt vector. After this, the IVR*i* will automatically provide the vector to the next lower priority interrupt found in the ISR*i* or indicate that no more interrupts are to be serviced.

  In case (b.), the user shall write a bit pattern to ISR*i* in order to clear one or more interrupts simultaneously. This write access will not touch the operational state of the ISR*i* (*frozen* or *not frozen*).

- The ISR*i* can be released by writing any value to IVR*i* (regardless which value is written, the value is ignored). This release operation shall be undertaken when the CPU has serviced all interrupts which appeared in the ISR*i*. After releasing, the ISR*i* is no longer frozen and the new interrupts, which accumulated in the IPR*i*, propagate into the ISR*i*. If interrupts are present again, the output signal INT*i* \ becomes active again.

Though this mechanism uses a fixed priority scheme, the priority rules the order how the interrupts are to be serviced. None of the lower priority interrupts can be overrun by higher priority interrupts.

Interrupt Signal path:



Two independent "Frozen" flags exist for both controller 0 and 1

Figure 2.20: Interrupt Control Data Flow Structure

(Space below has been left blank intentionally)

2.9.15.2 Interrupt Sources

Abbre-viation	Name	Brief Description, (MVBC Functional Unit)
TI1	Timer Interrupt 1	Indicates that the Timer 1 has reached zero. (Universal Timer)
TI2	Timer Interrupt 2	Indicates that the Timer 2 has reached zero. (Universal Timer)
MFC	Master Frame Checked	Indicates Master Frame being retrieved from TM (MF Slot or MF Table) and clearance is given to transmit it. This interrupt may occur before actual transmission starts. Usage: Information to Bus Administrator Software that next MF can be sent. (Main Control Unit)
SFC	Slave Frame Checked	Indicates completed transfer of a Slave Frame, regardless if an error has occurred or not. This interrupt does not occur if a SF Reply Timeout occurs. RTI would occur instead. (Telegram Analysis Unit)
EMF	Erroneous Master Frame	Indicates that an erroneous Master Frame has been received. This interrupt may occur at any device, even at the device which has sent the Master Frame. (Telegram Analysis Unit)
ESF	Erroneous Slave Frame	Indicates that an erroneous Slave Frame has been received. This interrupt may occur at any device, even at the device which has sent the Slave Frame. (Telegram Analysis Unit)
DMF	Duplicate Master Frame	Indicates that two duplicate Master Frames have been received within Slave Frame reply time (default value: 42.7 µs). (Telegram Analysis Unit)
DSF	Duplicate Slave Frame	Indicates that two duplicate Slave Frames have been received within the time a Slave Frame has been expected within Slave Frame reply time (default value: 42.7 µs). This is one possible outcome of a collision. (Telegram Analysis Unit)
BTI	Bus Timeout Inter-rupt	Indicates that no Master Frame has been received within 1.3 ms. Usage: Informs Bus Administrator Software to attain Master privileges in order to transmit a Master Frame itself. (Telegram Analysis Unit)
RTI	Reply Timeout Interrupt	Indicates that, after reception of a Master Frame, a Reply Timeout has occurred, meaning no Slave Frame has been received within the specified Reply Timeout period (default value: 42.7 µs). (Telegram Analysis Unit)
FEV	Frames Evaluated Interrupt	Indicates that 65,536 frames have been checked and that the total number of erroneous telegrams can be found in the Error Counter Register (EC). (Telegram Analysis Unit)
DTI*i* ***(i=0..7)***	Data Transfer In-terrupt *i*	Interrupt is asserted upon successful data transfer. The interrupts are specified in the PCS of the affected ports. (Main Control Unit)
XQE	Transmit Queue Exception	Indicates that both Transmit Queues are either empty or nonexistent. No message is sent. (MCU Queue Man-agement)
RQE	Receive Queue Exception	Failed attempt to receive a message into a full or non-existent Receive Queue. The received message is lost. (MCU Queue Management)
XQ0C	Transmit Queue 0 Complete	Indicates that the end of the queue 0 has been reached and all the contents have been successfully sent to the MVBC for transmission. It occurs after message transfer. (MCU Queue Management)
XQ1C	Transmit Queue 1 Complete	Indicates that the end of the queue 1 has been reached and all the contents have been successfully sent to the MVBC for transmission. It occurs after message transfer. (MCU Queue Management)
RQC	Receive Queue Complete	Indicates that the Receive Queue is now full. No data has been lost yet. (MCU Queue Management)
XI*i* **(i=0..3)**	External Interrupts	Four external interrupt inputs (II3..0 pins). Incoming interrupts are recognized by the falling edge. (Interrupt Logic)
AMFX	All Master Frames	Indicates that the last Master Frame has been retrieved from the Traffic Memory is now subject to be transmitted. At the moment Interrupt MFC occurs, the last Master Frame has been sent into the bus. (Main Control Unit)

Table 2.35: Interrupt Sources

2.9.15.3 Interrupt Pending Register (IPR0, IPR1)

All internal and external interrupts are collected in one of the Interrupt Pending Registers (IPR*i*). One bit is assigned to each interrupt. If the ISR*i* is frozen, then the interrupts accumulate in the IPR*i* until the ISR*i* is released. Writing '1's to IPR*i* will cause interrupts intentionally in order to test the interrupt controller.

Interrupt Pending Register (IPR0): Address 0yFB0H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	EMF	ESF	DMF	DSF	AMFX	MFC	SFC	RTI	BTI	DTI7	DTI7	DTI5	DTI4	DTI3	DTI2	DTI1
Init. Value:	all 0															
CPU Acc.:	RW															
MVBC Acc.:	rw															

Interrupt Pending Register 1 (IPR1): Address 0yFB4H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	TI2	XI3	XI2	XQE	RQE	XQ1C	XQ0C	RQC	FEV	Not used				TI1	XI1	XI0
Init. Value:	all 0									0				all 0		
CPU Acc.:	RW									R0				RW		
MVBC Acc.:	rw									-				rw		

Symbol	Description
All Interrupts	Read Access: 0 Affected interrupt is not pending 1 Affected interrupt is pending Write Access: 0 Remove pending interrupts (used to test interrupt mechanism) 1 Apply pending interrupts (used to test interrupt mechanism)

Table 2.36: Interrupt Vector Registers (IPR*i*)

2.9.15.4 Interrupt Mask Registers (IMR0, IMR1)

The Interrupt Mask Registers (IPR*i*) allow masking any interrupt source. If a bit in the IMR*i* is '1', then the corresponding interrupt will propagate from the IPR*i* to the ISR*i*, given that ISR*i* is not frozen. Otherwise, if the bit equals to '0', then the interrupt will be filtered away. The Interrupt Mask Registers allow read and write accesses, but are not altered by the MVBC.

Interrupt Mask Register (IMR0): Address 0yFB8H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	EMF	ESF	DMF	DSF	AMFX	MFC	SFC	RTI	BTI	DTI7	DTI7	DTI5	DTI4	DTI3	DTI2	DTI1
Init. Value:	all 0															
CPU Acc.:	RW															
MVBC Acc.:	r															

Interrupt Mask Register 1 (IMR1): Address 0yFBCH

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	TI2	XI3	XI2	XQE	RQE	XQ1C	XQ0C	RQC	FEV	Not used				TI1	XI1	XI0
Init. Value:	all 0									all 0				all 0		
CPU Acc.:	RW									R0				RW		
MVBC Acc.:	r									-				r		

Symbol	Description
All Interrupts	Read Access: 0 Interrupt is masked away (disabled) 1 Interrupt is not masked (enabled) Write Access: 0 Apply masking (Affected interrupt(s) will no longer penetrate into ISR*i*.) 1 Remove masking (Affected interrupt(s) will no longer penetrate into ISR*i*.)

Table 2.37: Interrupt Vector Registers (IMR*i*)

2.9.15.5 Interrupt Status Registers (ISR0, ISR1)

The Interrupt Status Registers provide information about the sources which have generated an outgoing interrupt. A read access to an ISR*i* will automatically freeze the affected ISR*i*. Writing '0's to the affected bits will clear the bit. Writing '1's will cause no action.

Interrupt Status Register (ISR0): Address 0yFC0H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	EMF	ESF	DMF	DSF	AMFX	MFC	SFC	RTI	BTI	DTI7	DTI7	DTI5	DTI4	DTI3	DTI2	DTI1
Init. Value:	all 0															
CPU Acc.:	RW0															
MVBC Acc.:	rw															

Interrupt Status Register 1 (ISR1): Address 0yFC4H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	TI2	XI3	XI2	XQE	RQE	XQ1C	XQ0C	RQC	FEV	Not used				TI1	XI1	XI0
Init. Value:	all 0									0				all 0		
CPU Acc.:	RW0									R0				RW0		
MVBC Acc.:	rw									-				rw		

Symbol	Description
All Interrupts	Read Access: 0 Interrupt is set. 1 Interrupt is not set. Attention: Read accesses will freeze the corresponding interrupt controller, regardless of the value read. Write Access: 0 Clear interrupt. 1 No effect on contents. Attention: Any modification in the ISR*i* has immediate effect on the interrupt vector found in the Interrupt Vector Register IVR*i*.

Table 2.38: Interrupt Vector Registers (ISR*i*)

2.9.15.6 Interrupt Vector Register (IVR0, IVR1)

The Interrupt Vector Registers return the interrupt vector and provide a mechanism to freeze and release the ISR*i*. The IVR*i* is connected directly via a priority encoder (combinational logic) to the corresponding ISR*i*.

Interrupt Vector Register (IVR0): Address 0yFC8H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	Not used							IAV	Not used				$VEC_{3..0}$			
Init. Value:	all 0							0	all 0				all 0			
CPU Acc.:	-							R	-				R			
MVBC Acc.:	-							w	-				w			

Interrupt Vector Register 1 (IVR1): Address 0yFCCH

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	Not used							IAV	Not used				$VEC_{3..0}$			
Init. Value:	all 0							0	all 0				all 0			
CPU Acc.:	-							R	-				R			
MVBC Acc.:	-							w	-				w			

Symbol	Description
IAV	Interrupt Available Indicates that at least one bit is set in the corresponding ISR*i* and that $VEC_{3..0}$ is an applicable interrupt vector.
$VEC_{3..0}$	Interrupt Vector. Read Access: If IAV=0, then $VEC_{3..0}$ = '0000'. Otherwise: Returns interrupt vector and automatically clears bit 2^($VEC_{3..0}$) in the ISR*i*. Consequence: The IVR*i* will be updated thereafter. Examples of data read: 0000 ISRi [15..1] = '0', ISRi [0] = '1' 0001 ISRi [15..2] = '0', ISRi [1] = '1', ISRi [0] = Don't Care : : : : 0010 ISRi [15..3] = '0', ISRi [2] = '1', ISRi [1..0] = Don't Care 1110 ISRi [15] = '0', ISRi [14] = '1', ISRi [13..0] = Don't Care 1111 ISRi [15] = '1', ISRi [14..0] = Don't Care Attention: Read accesses will freeze the corresponding interrupt controller, regardless of the value read. Write Access: Releases frozen state of affected interrupt controller (if frozen). The written value is ignored.

Table 2.39: Interrupt Vector Registers (IPR*i*)

2.9.15.7 Designers' Responsibilities

The HW designer must be aware that this Interrupt Logic uses quasi edge-driven inputs. The incoming interrupt signals are latched at the rising clock edge and double-sampled to detect a '1' to '0' transition. Therefore, these two levels shall be stable for at least 41.7 n (1 clock cycle).

Interrupts resulting from data transfer exceptions (EMF, ESF, DMF, DSF, RTI, BTI) will generate a copy of the Master Frame, too, but is stored in a separate register.

2.9.16 Universal Timers

The MVBC provides two general-purpose Universal Timers. Timer 1 can be used to start sending Master Frames (see Master Register, SMFT bit, section 2.9.8.1) at precise time intervals. Timer 2 is not directly used by the MVBC except that both timers can pass interrupts to the Interrupt Logic. The two timers have the following characteristics:

Timer 1:	Minimum Time:	10 µs	10 µs = 240 clock cycles
	Maximum Time:	ca 655 ms	10 µs x(216)
	Resolution:	10 µs	
	Interrupt:	TMR1	Timer 1 Interrupt
Timer 2:	Minimum Time:	250 ns	250 ns = 6 clock cycles
	Maximum Time:	ca 8.3 ms	125 ns x(216)
	Resolution:	125 ns	125 ns = 3 clock cycles
	Interrupt:	TMR2	Timer 2 Interrupt

Each timer consists of a 16-bit down-counter with automatic reload and manual reset mechanism. When a counter is active and its value reaches zero, then the output signal (TMR*i* \, *i*={1,2}) becomes active for three clock cycle (125 ns) and an interrupt is generated (TI*i*).

Each timer provides a Timer Counter Register (TC*i*) to hold the actual count value, a Timer Reload Register (TR*i*) to specify the period, and two control bits found in the Timer Control Register (TCR) to pause or reset the counter.

2.9.16.1 Timer Control Register (TCR)

Timer Control Register (TCR): Address 0yFE0H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	Not used										RS2	TA2	n. used	XSYN	RS1	TA1
Init. Value:	all 0										1	0	0	0	1	0
CPU Acc.:	-										RW0	RW	-	RW	RW0	RW
MVBC Acc.:	-										r	r	-	r	r	r

Symbol	Description
RS1, RS2	Reset Timer 1, 2 0 The selected timer is set to zero. If the corresponding TA*i* -bit is active, then the output signal TMR*i* \ and the interrupt T*i* I\ is asserted and the counter reloads automatically with the value stored in the Timer Reload Register TR*i* and down-counting will continue. Otherwise, if the corresponding TA*i* -bit is zero, then the output signal and the interrupt are not asserted and the counter stays at zero. The signal and the interrupt will not be asserted if TA*i* is set to '1' afterward in order to restart the counter. 1 No action Read Access: 1 Always returned.
TA1, TA2	Timer Active 1, 2 0 Timer is halted. Output signal TMR*i* \ and generation of interrupts are suppressed. 1 Timer is running.
XSYN	External Synchronization 0 No synchronization 1 Timer 1 is synchronized (if timer is active) Timer 1 issues a pulse (TMR1\) and interrupt immediately (if timer is inactive)

Table 2.40: Timer Control Register (TCR)

The Timer Reload Registers contain the automatic reload values for the respective Timer Counter Registers.

Timer Reload Register 1 (TR1): Address 0yFF0H
Timer Reload Register 2 (TR2): Address 0yFF4H

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	TR1, TR2															
Init. Value:	all 0															
CPU Acc.:	RW															
MVBC Acc.:	r															

These registers can be altered even while the counter is active. The new values become effective at the next reload.

TRi Reload Value	Period on Timer 1	Period on Timer 2
0	10 us	125 ns (250 ns) [1]
1	20 us	250 ns
2	30 us	375 ns
(k)	(k+1)*10 us	(k+1)*125 ns
65535	655.35 ms	3.292 ms

[1] Interrupt occurs every 125 ns, Timer signal TMRi \ every 250 ns. Reason: Pulse width = 125 ns.

Table 2.41: Timer Periods

The Timer Counter Registers contain the current count values. The values can be read and modified by the software *on-the-fly*. These registers can be altered while the respective timers are active or not. Writing a zero to one of these registers results to the same behavior as resetting them with writing '0' to the RS*i* -bit in the TCR.

Timer Counter Register 1 (TC1): Address 0yFF8H
Timer Counter Register 2 (TC2): Address 0yFFCH

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	TR1, TR2															
Init. Value:	all 0															
CPU Acc.:	RW															
MVBC Acc.:	rw															

2.9.16.2 External Synchronization

An external synchronization mechanism is supported for Timer 1. This mechanism can be enabled by setting the XSYN-bit (inside TCR) to 1. If enabled, the incoming interrupt signal II3\ will serve as the synchronization input. In order to prevent II3\ from generating interrupts, the Interrupt Mask Register must be set accordingly (See section 2.9.15.4). Synchronization takes place at the falling transition from '1' to '0', similarly done as for external interrupts. The synchronization is completed after two clock cycles.

The synchronization pulse invokes following actions:

- If the timer 1 is not active (TA1=0):

  A timer pulse is generated which leads to a Timer Interrupt, TMR1\ active for three cycles and MF dispatching being started if configured to do so.

- If the timer 1 is active (TA1=1):

  Timer Counter 1 (TC1) is set to zero. At the next 10 µs pulse, the reload takes place. Attention: The timer pulse occurs within 10 µs.

Applications:

Synchronized operation of multiple MVBCs which dispatch Master Frames to multiple MVBs at precisely synchronized time instances. Two variations are possible:

- Timer 1 of one MVBC is active and distributes its timer pulse to all other on-board MVBCs. Timer 1 of the other MVBCs is kept inactive
- Timer 1 in all MVBCs are active. After power-up, a synchronization pulse is passed into all MVBCs. This synchronization pulse may originate from an external unit, or from following outputs of one MVBC:

  - STROBE\ (for global synchronization)
  - TMR1\, TMR2\

  The synchronization pulse may be reissued sporadically. As long all MVBCs have been powered up or reset at the same time, the 10 µs pulses appear simultaneously and all MVBCs will operate perfectly synchronously.

2.9.17 Sink-Time Supervision Logic

The Sink-Time Supervision Logic helps to check whether data has been sent or received within a specified time interval. If the Sink-Time Supervision Logic is activated, then the TACK bits of a specified dock range are decremented by 1 at fixed time intervals. If a TACK value is already zero, then it will stay zero. The Sink-Time Supervision runs like a *background-task* in the MVBC, taking advantage of the idle periods of the MCU waiting for incoming Master or Slave Frames.

For every dock, the following steps are undertaken:

- Read TACK-Value (PCS, Word 3)
- If TACK is nonzero, then it is decremented by 1
- Write TACK-Value back to TM (only if the read TACK was not zero)

The Sink-Time Supervision Logic does not inform the CPU if one or more TACK bits have reached zero. The software is responsible to compare the TACK bits with a threshold value when retrieving (or depositing) data from (or to) the Traffic Memory.

2.9.17.1 Sink-Time Supervision Register (STSR)

The Sink-Time Supervision Register specifies the range of docks and supervision time interval. The user is responsible to avoid TM access overloading due to supervising too many docks within a too small time interval. See Figure 2.22 for limiting factors.

Sink-Time Supervision Register (STSR): Address 0yF8CH

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$SI_{3..0}$				$R_{11..0}$											
Init. Value:	all 0				all 0											
CPU Acc.:	RW				RW											
MVBC Acc.:	r				r											

Symbol	Description
$SI_{3..0}$	Supervision Interval 0 = Inactive 2 = 2 ms 4 = 8 ms 6 = 32 ms 8 = 128 ms 1 = 1 ms 3 = 4 ms 5 = 16 ms 7 = 64 ms 9-15 = 256 ms
$R_{11..0}$	Range of docks to be supervised This 12-bit value specifies that all docks from zero to R11..0 are subject to sink-time supervision. The user must assure that this value does not exceed the number of available docks in the LA Data Area assigned by the selected Memory Configuration Mode.

Table 2.42: Sink-Time Supervision Register (STSR)

2.9.17.2 Traffic Memory Loading

The sink-time supervision takes place in bursts. When the time interval counter has elapsed, then the TM accesses start. In order to avoid bus traffic overflows, the following formula can be applied to find the minimum possible time interval for a given number of ports.

$$t_{acc} = 2\ t_{cp}\ (W + 3) \qquad t_{bc} = N\ t_{acc} \qquad t_{wc} = N\ (t_{acc} + 2\ A\ t_{cp})$$

t_{cp} Clock Period, 41.67 ns
W Number of waitstates
t_{acc} Access time required to read and rewrite one TACK word (ns)
A Average time required to perform one arbitration switch, here: 2 clock cycles
N Number of docks to be supervised
t_{bc} Best-case timing (ns; no arbitration switch required)
t_{wc} Worst-case timing (ns; two arbitration switches required for every access)

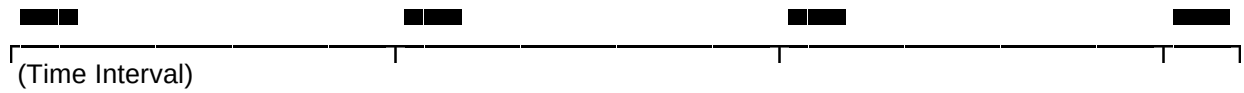
t_{bc} and t_{wc} represent fictitious boundaries. In field applications, some value inbetween will apply. Software engineers are urged to base their computations on the worst-case parameters:

$$L = \frac{t_{wc}}{10^{6}\ I} \times 100$$

I Time Interval (ms)
L Loading ratio (percent)

The loading ratio L represents the worst-case time fraction the Sink-Time Supervision accesses the TM. The two charts shown on the next page provide a better understanding of these formulae. The following diagrams illustrate how the Sink-Time Supervision Logic accesses the TM and how these accesses interact with those originating from the CPU and MCU.

Sink-Time Supervision without any communication:



TM sharing between CPU, MCU and Sink-Time Supervision:

Legend: Data CPU ↔ TM Data MCU ↔ TM Sink-Time Supervision

Figure 2.21: Sink Time Supervision Events

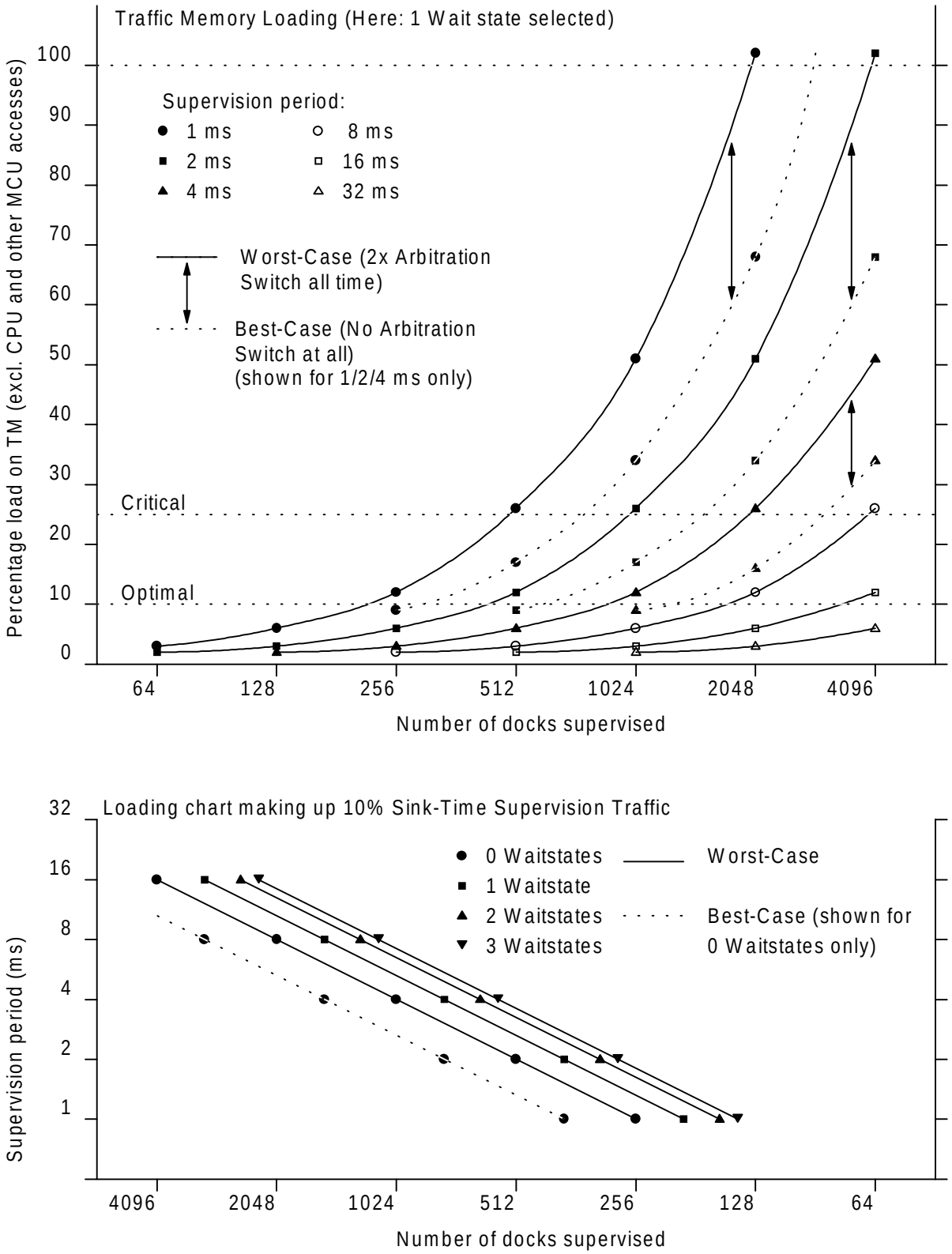


Figure 2.22: Traffic Memory Loading Charts

2.10 Clock Generator

The Clock Generator generates all clock and timing interval pulses for the Decoder, Universal Timers and MCU.

2.11 Testing Facilities

The MVBC exhibits a high degree of testability and observability. The different test mechanisms are described in the following sections.

2.11.1 Ad-Hoc Test Facilities

The internal MVB loopback mechanism allows elaborate functional tests such as

- Transfers using all F-Codes
- Full scale event arbitration
- Device Status Polls
- Simulated telegram errors (i.e. CRC), timeouts and jammed lines

In addition, the high degree of obsersvability (read-access to all Internal Registers) permits tests such as

- Interrupt Logic with simulated interrupts
- Universal Timers
- Device Address Override Mechanism
- Register Test

2.11.2 JTAG: Boundary Scan

The MVBC supports following Boundary Scan commands according to IEEE 1149.1 JTAG:

Command	**Description**
IDCODE	Returns ID Code for MVBC: **1ABC D02B** (Hexadecimal)
SAMPLE	Boundary Scan: Captures values of input, I/O pads
EXTEST	Boundary Scan: Sets values on output, I/O pads
BYPASS	Bypass Mode
SCANMODE	Reserved for Internal Scan (see section 2.11.3). Attention: This feature does not comply with IEEE 1149.1

Table 2.43: Supported JTAG Functions

Besides the four mandatory JTAG signals (TCK, TMS\, TDI, TDO), the MVBC provides a Test-Reset-Signal (TRST\). At power-up, this signal must be active while RESET\ is active. After that, TRST\ will reset the JTAG logic only.

The MVBC contains 80 Boundary Scan cells. These cells cover 77 pads (except power and JTAG pins) and three tristate enable signals (Address Bus, Data Bus, Z-Mode control for remaining pads)

Scan cell order:

Scan Cell Order (follows pins clockwise around the chip) i.e. output of cell 1 is connected to input of cell 2

1	STROBE\	2	A_1 ...	20	... A_{19}	21	A_EN\
22	INT0\	23	INT1\	24	RDY\	25	RD\
26	WR\	27	TM_REQ_CPU\	28	TMRDY\	29	DA_0 ...
40	... DA_{11}	41	RESET\	42	CLASS_MODE	43	Z_MODE\
44	TMR1\	45	TMR2\	46	D_EN\	47	DIR

Continued on next page.

Continued from previous page.

48	TMRD\	49	TMWR\	50	TMCS\	51	D_{15} ...
66	... D_0	67	II_0\ ...	70	... II_3\	71	SF
72	OCA	73	DISCHARGE	74	FCL	75	CLK
76	ICA	77	ICB				

78	Enable(D15..0)	79	Enable(A19..1)	80	Enable all other output signals; See section 2.5.2.

2.11.3 Internal Scan

The MVBC provides 15 linear internal scan paths which are used to check for internal faults. Each path covers not more than 70 flip flops. The two RAM arrays for TXB and RXB, transparent latches (used to delay signals by half clock cycle) and one SR-latch (found in the Arbitration Controller) are not included in the path.

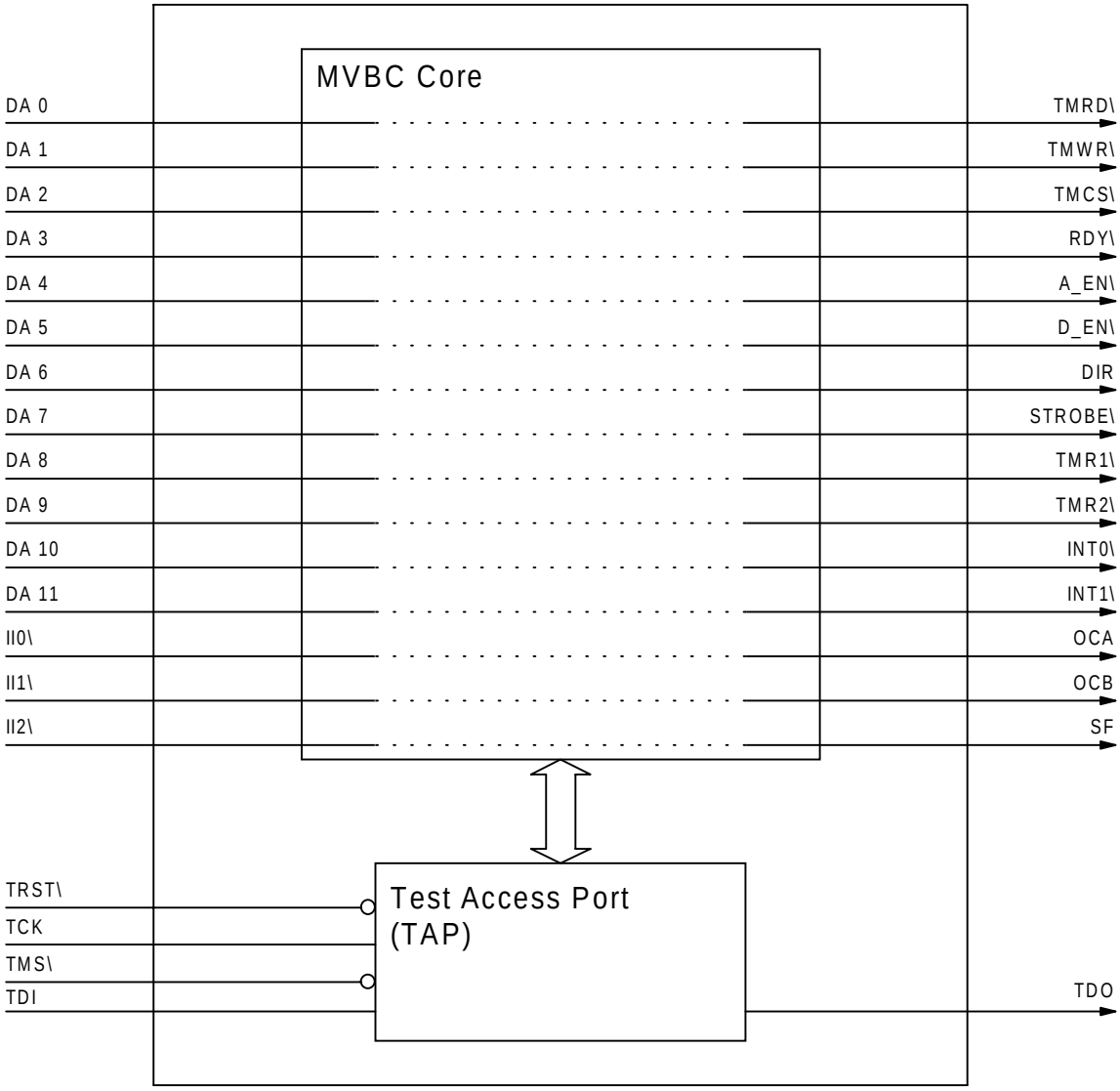


Figure 2.23: Internal Scan Mechanism

Instructions to enter Internal Scan Mode: Load "**110**" into the JTAG Instruction Register (IR), then pass through following states: "Exit IR", "Update IR", "Run-Test/Idle".

2.11.4 Isolation of internal RAM cells

The following bit combination isolates the RAM arrays serving as Transmit and Receive Buffers (TXB, RXB) from the remaining circuitry inside the MVBC in order to permit easy testing:

RES\ = '0' and RD\ = '0' and WR\ = '0'

The inputs and outputs of the internal RAMs will be connected directly to the external pins. The following table shows the signal substitution for the external pins:

Pin	Substitution	Description
CLK	CLK	Clock signal for RAM arrays
CLASS_CMODE	Tst_WE\	Write Enable Signal
DA_4	Tst_MemSel\	Selects Transmit Buffer (0) or Receive Buffer (1)
$DA_{3..0}$	Tst_$A_{3..0}$	Address to RAM arrays
$A_{16..1}$	Tst_$DI_{15..0}$	Input data to RAM arrays
$D_{15..0}$	Tst_$DO_{15..0}$	Output data to RAM arrays

Table 2.44: Multiplexed Pins for Multiplex-Isolated RAM Test

The memory test cannot be performed while the MVBC is in regular operation (transferring data, processing interrupts, etc.). This restriction is obvious since the RESET\-signal is active. The following figure illustrates the effective interconnection while the RAMs are isolated:

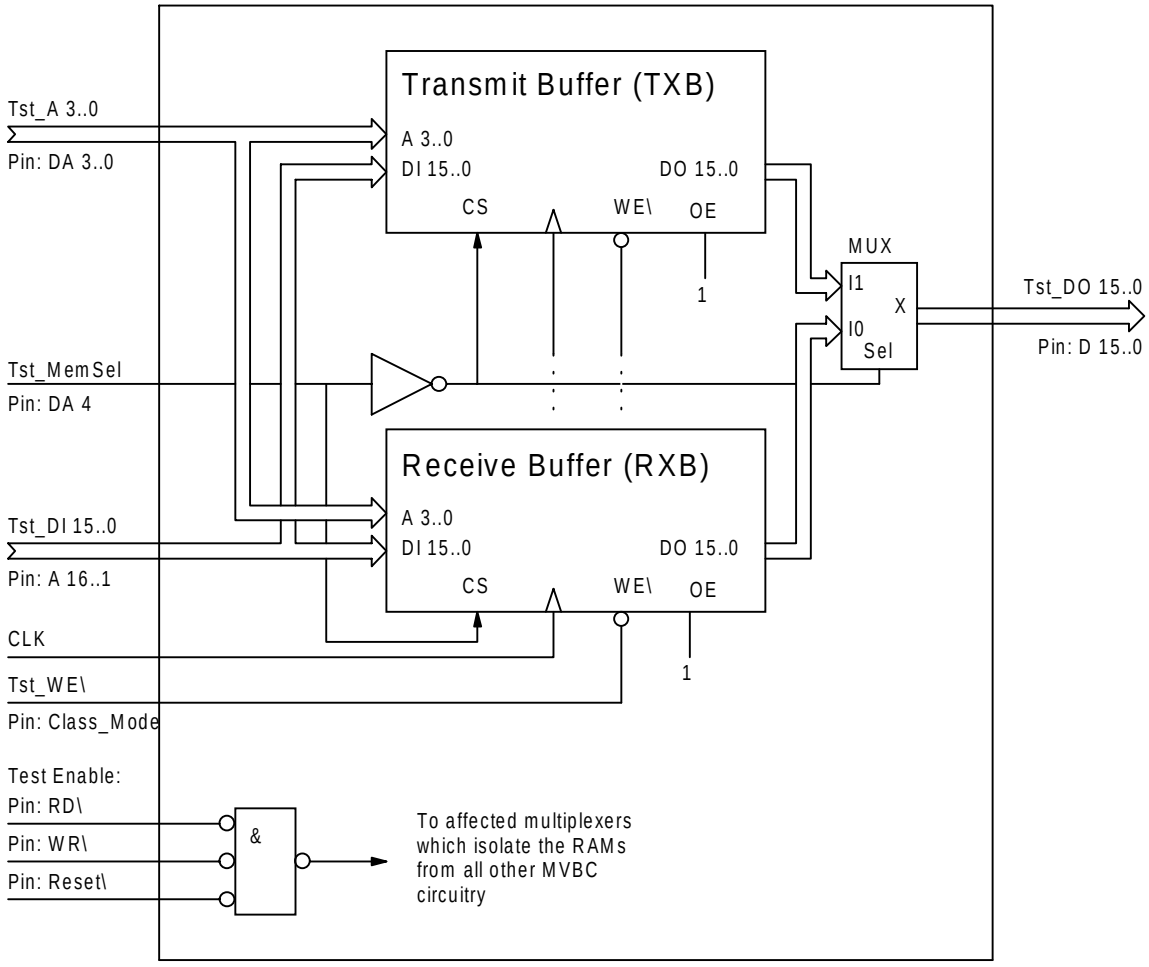


Figure 2.24: Isolated RAM Arrays

The timing diagrams showing the signals to perform memory tests are summarized in section 7.3. Please note that consecutive write accesses cannot be made over both TXB and RXB blocks at the same time, meaning Tst_MemSel kept constant while Tst_WE\ is active. Tst_MemSel must go inactive before the other RAM is tested. This restriction does not apply to read accesses. After the memory test has been completed, the MVBC must be properly reset (RESET\ and TRST\ active) before regular operation can resume.

2.11.5 High-Impedance Mode

All I/O and output pads enter High-Impedance Mode immediately when the input signal Z_MODE\ is active. As long the MVBC is not operating in initialization level 2 or 3, and the input signals remain unchanged, the internal state will be preserved. In the application environment, Z_MODE\ must be connected to '1'.

2.11.6 RAM Tests on Traffic Memory

As long the MVBC operates with Initialization Level 1, any regular RAM-test algorithm can be applied to test the Traffic Memory. If the MVBC operates with the level 2 or 3, then testing capabilities are restricted in order to maintain data consistency.

If an on-line RAM test is compulsory, then Arbitration Mode 2 (ARB=2, see SCR) can be applied. This mode holds back any accesses made by the MVBC in order to assure data consistency. However, this mode must not be active for a too long time in order to avoid overruns. Therefore, the on-line RAM test should be divided into a series of small indivisible access sequences. Each locked interval should not exceed 2-4 µs, and the time fraction the TM is locked should not exceed 1/5. Two examples are shown below:

Example 1: RAM Test without checking for coupling faults:

Set Arbitration Mode to 2	$ARB_{1..0}$ <= "10"
Read primary cell	X1 <= TM[A1]
Write pattern to primary cell	TM[A1] <= P1
Read + Check primary cell	P1 =?= TM[A1]
Restore primary cell	TM[A1] <= X1
Restore old arbitration Mode	$ARB_{1..0}$ <= *Old Value*

Pause! Do not continue immediately!

Example 2: RAM Test with checking for coupling faults:

Set Arbitration Mode to 2	$ARB_{1..0}$ <= "10"
Read primary cell	X1 <= TM[A1]
Read secondary cell	X2 <= TM[A2]
Write pattern to primary cell	TM[A1] <= P1
Read + Check primary cell	P1 =?= TM[A1]
Read + Check secondary cell	X2 =?= TM[A2]
Restore primary cell	TM[A1] <= X1
Restore old arbitration Mode	$ARB_{1..0}$ <= *Old Value*

Pause! Do not continue immediately!

3 BEHAVIORAL OVERVIEW

This chapter summarizes the behavior the MVBC will follow during data transfers. The data transfers can be arranged into following categories:

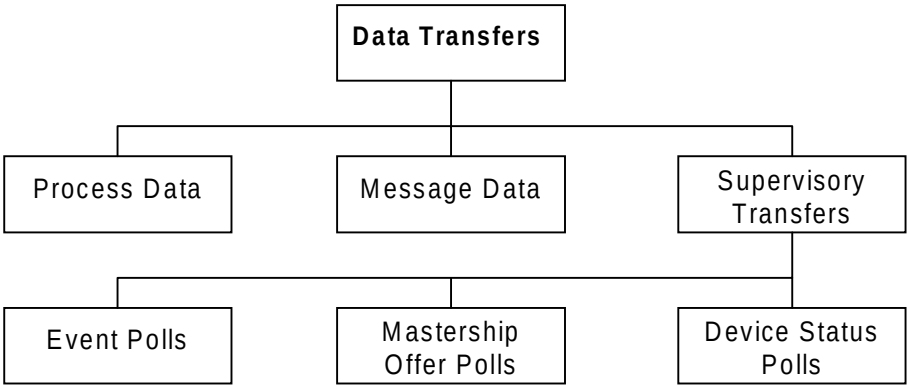


Figure 3.1: Types of Data Transferred

Section 3.1 indicates the preconditions the MVBC must fulfill in order to participate on MVB communication. Section 3.2 summarizes the general procedure the MVBC will undertake while transferring data over the MVB. The different data transfers (Process Data, Message Data, etc.), which may alter or supplement the general procedure, are summarized in the following sections.

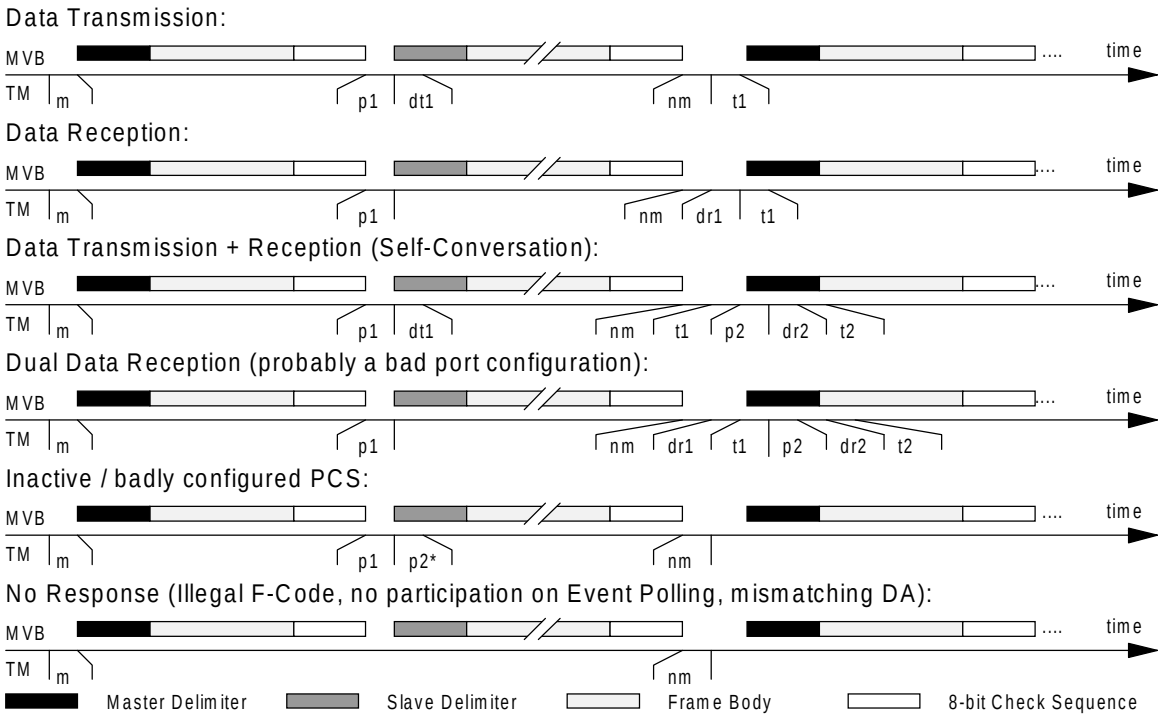
3.1 Preconditions

Following conditions must be met in order to let the MVBC participate on the MVB traffic:

- Initialization Level IL1..0 in the SCR must be set to '11'. If '10' is selected, then the full functionality is restricted to the internal loop-back lines (no outgoing signals). Values '00' and '01' do not permit data communications.
- A valid timeout coefficient TMO1..0 must be specified. The default value is 42.7 µs.
- A valid Device Address must be configured (either over DA11..0 pins or DAOR, see section 2.9.9.1)
- Event polling: In order to participate on event polling rounds, the RCEV-Bit in the SCR must be active.
- Masters only: The MAS-bit in the SCR must be active if the MVBC shall send Master Frames.

3.2 General Procedure

The MVBC makes following TM accesses while one telegram is transferred:



* If a 2nd port must be processed and is active, then it will be processed as if it would be the 1st port

m Master Frame Retrieval — Applies to active Masters only

nm MF Retrieval for next Telegram
- Manual Transmission: Retrieve MF from Master Frame Slot (MFS)
- Automatic Transmission: Retrieve MF from MF-Table pointed by DPR

Takes place before port processing in order to avoid delays

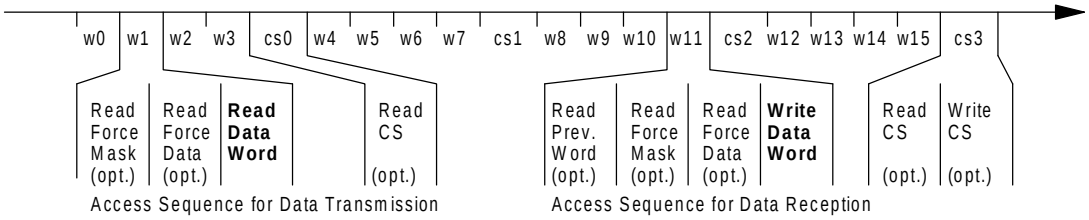
p1 Port Preprocessing
- (2 - 8 accesses)

Read LA-PIT or DA-PIT. No PIT-access for Physical Ports (PP)
Read LA / DA / PP - PCS Word 0
Read LA / DA / PP - PCS Word 1
Queue Preprocessing (optional, max. 5 accesses)

- Processing discontinues if port is inactive, F-Code in PCS mismatches, or port is temporarily disabled (PTD)

dt1 Data Transmission: 1-52 accesses (1-16 words)
dr1 Data Reception: 0-72 accesses
dr2 Data Reception for 2nd port (in case of self-conv.)

(Slave delimiter is already underway while data is retrieved from the TM)



t1 Port Postprocessing
- (2 - 8 accesses)

Write 0xFFFF to TACK bits (applies if data transfer took place without errors)
Read LA / DA / PP - PCS Word 1
Write LA / DA / PP - PCS Word 1
Queue Postprocessing (optional, max. 5 accesses, applies if no errors occurred)

t2 Port Postprocessing for 2nd port (in case of self-conversation)

p2 Port Preprocessing for 2nd port (in case of self-conversation)

Figure 3.2: TM Access Sequences

See also: "Appendix A: Function Code Summary", "Appendix B: Port Processing Overview" and "Appendix C: Required PCS Settings for all Ports" which are helpful to configure ports.

3.3 Process Data Transfers (F-Codes 0-4)

The MVBC supports Process Data Transfers for all frame sizes specified in [1]. The MVBC handles Process Data in the Logical Address space.

Master Frame:

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$F\text{-}Code_{3..0} = \{0..4\}$				Logical Address $LA_{11..0}$											

Slave Frame:

F-Code:	0	1	2	3	4	5-7
Frame Size:	1	2	4	8	16	not supported

Applicable Ports:

Port	Condition to process port
Port in LA Space	LA Port Index Table is configured PCS of affected port is configured (active, not temp. disabled)

Table 3.1: Process Data Transfers

Test case: If UTQ (SCR bit) is active, then the Test Source Port (TSRC) is checked before the LA port. If UTS is active, then the Test Sink Port (TSNK) is checked after the LA port. If both UTQ and UTS are active, then TSRC and TSNK are checked, but no LA port.

Allowed Options:

- Forcing Data (see section 3.9.1)
- Supporting non-real-time systems (see section 3.9.2)
- Data Transfer Interrupts (see section 3.9.3)
- Automatic Data Comparison Mechanism (see section 3.9.4)
- Synchro Port (see section 3.9.5)
- Transfer with Check Sequence (see section 3.9.6)
- Transferring non-numeric Data (see section 3.9.7)
- Write Always-Option to recover erroneous data (see section 3.9.8)

Discouraged Options:

- Attaching Message Queues
- Attempting to request Data Comparison Mechanism on source port

Attention: If no port shall be assigned to a specific Logical Address, then the Port Index shall point to a common inactive port (i.e. Port 0) where the PCS defines a passive port (SRC=0, SINK=0).

Attention: Assigning two different Logical Addresses to one common port index is allowed, but not strongly recommended.

During normal operation, the MVBC cannot send Process Data to itself since the same port cannot be declared as a bidirectional port. However, self-conversation can be invoked for self-test purposes by enabling UTS and/or UTQ so the MVBC will always check the Test Port.

3.4 Mastership Offer Poll (F-Code 8)

The MVBC handles Mastership Offer Polls as simple device-addressed polls. The addressed bus participant replies with a 1-word Slave Frame if a Master Frame with F-Code 8 is received. The software, which handles TCN upper layer protocols, is responsible to use this F-Code to handle mastership transfers from one Bus Administrator to another.

Master Frame:

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$F\text{-}Code_{3..0} = \{8\}$				Device Address $DA_{11..0}$											

Slave Frame: 1 Word

Applicable Ports:

Port	Condition to process port
Mastership Offer Source Port (FC8)	Device Address in MF matches with own Device Address PCS is configured (active source, not temp. disabled)
Mastership Offer Sink Port (MOS)	PCS is configured (active sink, not temp. disabled)

Table 3.2: Mastership Offer Polls

Allowed Options:

- Data Transfer Interrupts (see section 3.9.3)

3.5 Device Status Polls (F-Code 15)

The MVBC supports Device Status Polls for both Class 1 and 2/3/4 Mode. If the MVBC operates in Class 1 Mode, then the Device Status Report is generated inside the MVBC. See section 2.9.14.2 for details. Otherwise, the Device Status Report is retrieved from port FC15.

Master Frame:

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$F\text{-}Code_{3..0} = \{15\}$				Device Address $DA_{11..0}$											

Slave Frame: 1 Word

Applicable Ports:

Port	Condition to process port
Device Status Port (FC15)	Device Address in MF matches with own Device Address PCS is configured (active source, not temp. disabled)
Port in DA Space	DA Port Index Table is configured PCS is configured (active sink, not temp. disabled) Nonzero MCM (if MCM=0, then no DA port space is available)

Table 3.3: Device Status Polls

Allowed Options:

- Data Transfer Interrupts (see section 3.9.3)

3.6 Event Arbitration

The MVBC supports the lower-level protocol for Event Arbitration. Event arbitration is necessary in order to let one MVBC transfer sporadic data (i.e. a message) to another MVBC. Details on the Event Arbitration protocol are discussed in [1]. Three different Event Polls are supported and described next:

- Start Event Polls (F-Code 9)
- Group Event Polls (F-Code 13)
- Individual Event Polls (F-Code 14)

3.6.1 Start Event Polls (F-Code 9)

All Event Polling Rounds must start with a Start Event Poll in order to let MVBCs, which have announced events, to participate.

Master Frame:

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	F-Code$_{3..0}$ = {9}				Event Mode EM$_{3..0}$				Event Type ET$_{3..0}$				Reserved			

Slave Frame: 1 Word, known as Event Frame; Collisions or no replies may occur.

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	F-Code$_{3..0}$				Logical Address LA$_{11..0}$ or Device Address DA$_{11..0}$											

The Event Frame must contain this format since the Bus Administrator returns it as a Master Frame.

Applicable Ports:

Port	Condition to process port
Event Frame Source Ports for both Event Types ET={0,1} (EF0, EF1)	Depends on Event Mode (EM) and Event Type (ET) PCS is configured (active source, not temp. disabled) RCEV bit is active
Event Frame Sink Port (EFS)	PCS is configured (active sink, not temp. disabled) RCEV bit is active

Table 3.4: Start Event Polls

If RCEV is 0, then participating on Event Polling is suspended. The EFS must be used by the Bus Administrator. All other slave devices can keep this port inactive or active for passive monitoring purposes.

Allowed Options:

- Data Transfer Interrupts (see section 3.9.3)

Event Types (ET):

ET	Description
0	Start/continue Event Polling Round events announced for high priority. If events are announced for low-priority polling (ET=1) only, then the MVBC will not participate until the next Start Event Poll (MF: F-Code 9, ET=0). Affected port: EF0. Affected flags: EA0 and PAR0 in the Master Register (MR).
1	Start/continue Event Polling Round events announced for low priority. If events are announced for high-priority polling (ET=0) only, then the MVBC will not participate until the next Start Event Poll (MF: F-Code 9, ET=0). Affected port: EF1. Affected flags: EA1 and PAR1 in the Master Register (MR).
2-15	Illegal. The MVBC will not respond to these values.

Table 3.5: Event Types (ET)

The MVBC *memorizes* the Event Type for all upcoming Group and Individual polls (F-Codes 13 and 14) until the next Start Event Poll (F-Code 9) is made. If ET=0 (or 1), then Port EF0 (or EF1) are used to retrieve the Event Frame. Announced and participating events for the two Event Types behave independently from each other, meaning a poll with ET=0 does not affect any MVBC flags related to ET=1 (i.e. PAR1, EA1, CPE1).

Event Modes (EM):

EM	Replying Allowed	Polling Round	Description (See flow-chart on Figure 3.3)
0	Yes	Continued	All MVBCs, which are currently participating at the current Event Polling Round (PARET='1'), will reply with an Event Frame.
1	Yes	Started	All MVBCs with announced events (EAET='1') will participate (PARET:='1') at this Event Polling Round and reply with an Event Frame
2	No [1]	Continued	All participating MVBCs (PARET='1') will continue to do so. This Event Mode can be used to change Event Type.
3	No [1]	Started	All MVBCs with announced events (EAET=1), will participate (PARET:='1') at this Event Polling Round without replies.
4-15	No	Illegal	The MVBC will not respond to these EM.

[1] For EM=2 and 3, no bus participant may reply and all MVBCs will wait until a Reply Timeout occurs.

Table 3.6: Event Modes (EM)

3.6.2 Group Event Polls (F-Code 13)

The Group Event Polls provide a mechanism to poll 2,4,8,...all 4096 MVB devices. If the Start Event Poll resulted in a collision, then this poll can be used to split the devices into two equal-sized groups (i.e. 2x2048) which are polled accordingly. If collisions continue to occur, then the groups are split again and polled. This group poll can be used until a small enough group of devices has been isolated where only one device replies with an Event Frame. The smallest group covers 2 devices.

Master Frame:

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$F\text{-}Code_{3..0} = \{13\}$				Device Group Address $DGA_{11..0}$											

Slave Frame: 1 Word, known as Event Frame; Collisions or no replies may occur. See also section 3.6.1.

Applicable Ports:

Port	Condition to process port
Event Frame Source Ports for both Event Types ET={0,1} (EF0, EF1)	Device Group Address (DGA) matches PAR*i* is active, where *i*=ET of last Start Event Poll PCS is configured (active source, not temp. disabled) RCEV bit is active
Event Frame Sink Port (EFS)	PCS is configured (active sink, not temp. disabled) RCEV bit is active

Table 3.7: Group Event Polls

Allowed Options:

- Data Transfer Interrupts (see section 3.9.3)

Device Group Address:

The DGA is a 12-bit pattern which can be used to select a bi-sectional group of 2,4,8,... all 4096 Devices.

$DGA_{11..0}$	**Description**
1111 1111 1110	All devices are addressed (global poll)
1111 1111 110A	Half of them: DAs with matching DA_0 = A
1111 1111 10AA	One quarter of of them: DAs with matching $DA_{1..0}$ = 'AA'
:	:
10AA AAAA AAAA	Four of them are addressed: DA's with matching $DA_{13..0}$
0AAA AAAA AAAA	Two of them are addressed: DA's with matching $DA_{14..0}$

Table 3.8: Device Group Addresses (DGA)

3.6.3 Individual Event Polls (F-Code 14)

The Individual Event Polls address single devices to obtain Event Frames, as long this devices participate on the event polling round for the Event Type defined in the last Start Event Poll.

Master Frame:

Bit Number	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Symbol	$F\text{-}Code_{3..0}$ = {15}				Device Address $DA_{11..0}$											

Slave Frame: 1 Word, known as Event Frame; Collisions or no replies may occur. See also section 3.6.1.

Applicable Ports:

Port	**Condition to process port**
Event Frame Source Ports for both Event Types ET={0,1} (EF0, EF1)	Device Address (DA) matches PAR*i* is active, where *i*=ET of last Start Event Poll PCS is configured (active source, not temp. disabled) RCEV bit is active
Event Frame Sink Port (EFS)	PCS is configured (active sink, not temp. disabled) RCEV bit is active

Table 3.9: Individual Event Polls

Allowed Options:

- Data Transfer Interrupts (see section 3.9.3)

3.6.4 Software Responsibility

Following aspects of Event Arbitration is must be covered by software:

All Class 2/3/4 Devices, operating as slaves:

- Following steps must be undertaken by the Link Layer Software to initiate a sporadic data transfer:
  - Choose Event Type ET, 0 or 1.
  - The Event Frame (Master Frame contents which would initiate the transfer) must be copied into the Data Area of the EF*i* (*i*=ET). Example for Message Data transfers: Event Frame contains F-Code 12 and own Device Address.
  - The PCS of EF*i* must be configured as active source. Both CPEi-bits must be inactive.
  - The source port, involved in the sporadic data transfer, must contain valid data and its PCS must be configured accordingly. The CPE*i*-bit (Clear Pending Event) must be active.
  - EA*i* is set to '1' to announce the event.
- Participation on event polling will be cleared when the sporadic data transfer (i.e. Message Data) has taken place. This is accomplished with the active CPE*i* bit.
- Clearing participation: An exception applies for queued messages. See section 3.6.1.
- If required, the software can cancel announced and participating events by writing '1' to EC0 or EC1.

Attention: If CPE0 or CPE1 is not set, then the event will never be signed off and the MVBC continues participating at the next event polling round immediately.

Attention: The user shall not enable any of the CPE*i* bits in the PCS of those ports which are not intended for event-driven data transfers. Otherwise, event arbitration does not function properly.

All Bus Administrators (BA), operating as masters:

- Bisectional algorithm to perform a complete Event Polling Round. This algorithm is part of the BA Software.
- The Bus Administrator Software must copy the Event Frame from the Event Frame Sink Port (EFS) to the Master Frame Slot (MFS) in order to return the Event Frame as a Master Frame.

Third-Party Events:

Technically, device A may announce an event to instruct device B to send data to device C. In this case, device A must still provide a sink port for the transferred data in order to clear the announced event (active CPEi bit!).

Overruns:

A successful transmission of a port with activated CPE*i* will clear the announced event no matter what caused the transmission, i.e. cyclic Process Data transfer. Three cases may occur:

1. The master transferred the affected data before starting the next Event Polling round. The sporadic data transfer has been taken care and the MVBC will not participate on the upcoming Event Polling round.
2. The master received the Event Frame and retransmits it. The affected data transfer takes place twice.
3. The master received the Event Frame and the BA software recognizes that the Event Frame is identical with the previous Master Frame transmitted. Therefore, one transfer can be cut.

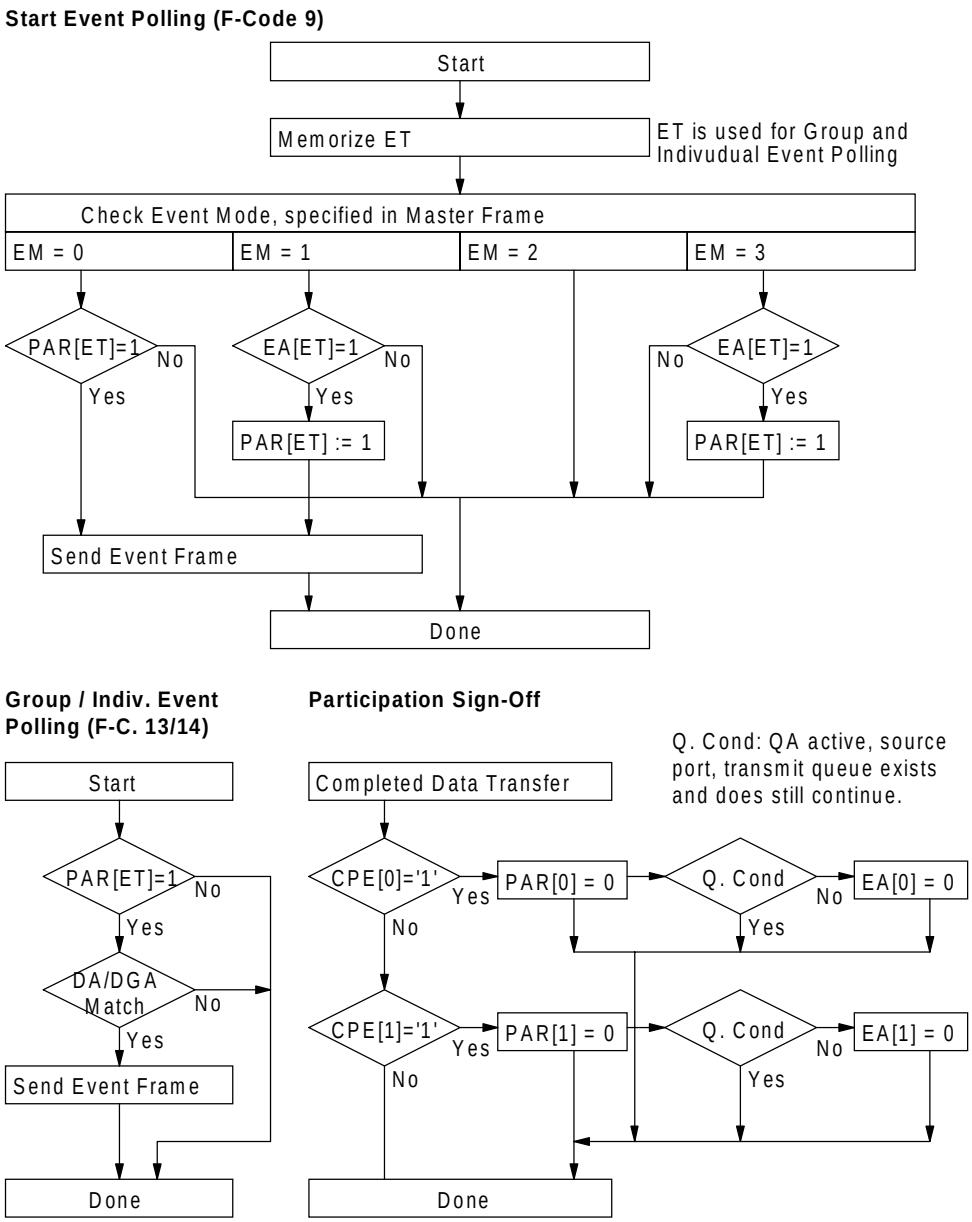


Figure 3.3: Event Polling Flowcharts

3.7 Message Data Transfers (F-Code 12)

The MVBC supports queued and nonqueued Message Data transfers. Message transfers are either sent to individual devices or broadcast to all devices.

Master Frame:

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$F\text{-}Code_{3..0} = \{12\}$				Device Address $DA_{11..0}$											

Slave Frame: 16 Words, Word 0 must contain following information:

Address *i* (Low Address Byte):

Bit Number	7	6	5	4	3	2	1	0
Symbol	Comm. Mode $CM_{3..0}$				$DDA_{11..8}$			

Address *i+1* (High Address Byte):

Bit Number	7	6	5	4	3	2	1	0
Symbol	$DDA_{7..0}$							

DDA = Destination Device Address

The format in Word 0 is mandatory since it is inspected by the MVBC. The entire message is considered as a non-numeric byte stream. The MVBC will take necessary actions to transfer the low address byte before the high address byte.

Applicable Ports:

Port	Condition to process port
Message Source Port (MSRC)	Device Address (DA) matches with own device PCS is configured (active source, not temp. disabled, NUM-bit = 0)
Message Sink Port (MSNK)	At least one of following 3 conditions must apply: Device Address matches given CM=1 or CM=15 (Broadcast) or MBC (SCR bit) is active and CM is valid (Broadcall) PCS is configured (active sink, not temp. disabled, NUM-bit = 0)

Table 3.10: Message Data Transfers

Allowed Options:

- Data Transfer Interrupts (see section 3.9.3)
- Transferring non-numeric Data - required! (see section 3.9.7)

Communication Modes (CM):

ET	Description
1	Point-to-Point Message: The receiver may receive the message if Destination Device Address matches with own DA. Exception: Message Broadcalling (MBC, see SCR) is enabled.
15	Broadcast message: The receiver may receive the message.
0, 2-14	Illegal. The MVBC will not receive any messages with these CM.

Table 3.11: Communication Modes (CM)

If no queues are used, then Message Data is stored to and retrieved from the Data Area, using the page pointer (VP) to address the page, as it is done with all other ports. If the queuing mechanism is enabled, then all messages are stored to and retrieved from Message Queues.

3.7.1 Message Queues

Message queueing is enabled if the QA-bits of the message ports (MSRC and/or MSNK) are set to '1'.

Attention: Queuing can only function properly in conjunction with event arbitration using one Event Type. The CPE bit for the chosen Event Type must be active in the Message Source Port.

Message Queues consist of the Linked List Structure (LLR) and the Queue Data area. See section 2.8.7 for details. Whenever a message is sent or received, the MVBC will perform two additional operations:

- Queue Preprocessing
- Queue Postprocessing

Queue Preprocessing takes place before outgoing data is read from one of the Transmit Queues (XQ0, XQ1), or before incoming data is stored to the Receive Queue (RQ). For outgoing messages, this operation reads the LLR pointer from QDT[XQ0] first. If this pointer is zero, then it assumes that XQ0 is *nonexistent* and will check XQ1. Otherwise, it retrieves the Data Pointer (DP). If DP is zero, then it assumes that XQ0 is *empty*, so XQ1 will be checked. XQ1 will be checked in the same manner as XQ0. If XQ1 is *nonexistent* or *empty*, too, then the interrupt "Transmit Queue Exception" (XQE) is asserted and message transfer will be cancelled. Otherwise, the Message Data ready to be retrieved and sent. For incoming messages, this operations checks the RQ in a similar manner. If the RQ is *nonexistent* or *full*, then the "Receive Queue Exception" (RQE) is asserted. There is no alternative Receive Queue to check.

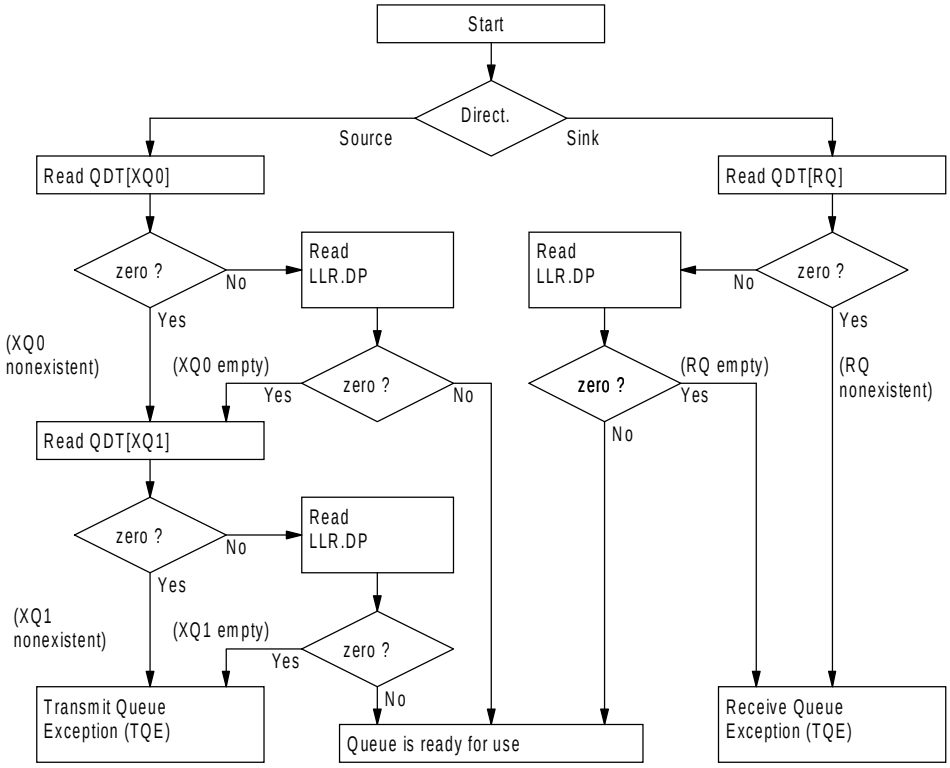


Figure 3.4: Queue Preprocessing

Queue Postprocessing takes place after outgoing data is read from one of the Transmit Queues or stored in the Receive Queue. This operation is needed to update the Queue Descriptor Table so the QDT of the affected queue points to the next entry in the LLR. This algorithm also checks whether the end of a particular queue has been reached or not. In detail, the Next Pointer of the current LLR is copied into QDT. If this pointer is zero, or the Data Pointer in the next LLR is zero, the MVBC assumes that the end has been reached. Concerning Transmit Queues: XQ0 is always checked before XQ1. If the end has been reached, then the interrupts "Transmit Queue 0/1 Complete" (XQ0C, XQ1C) or "Receive Queue Complete" will be asserted.

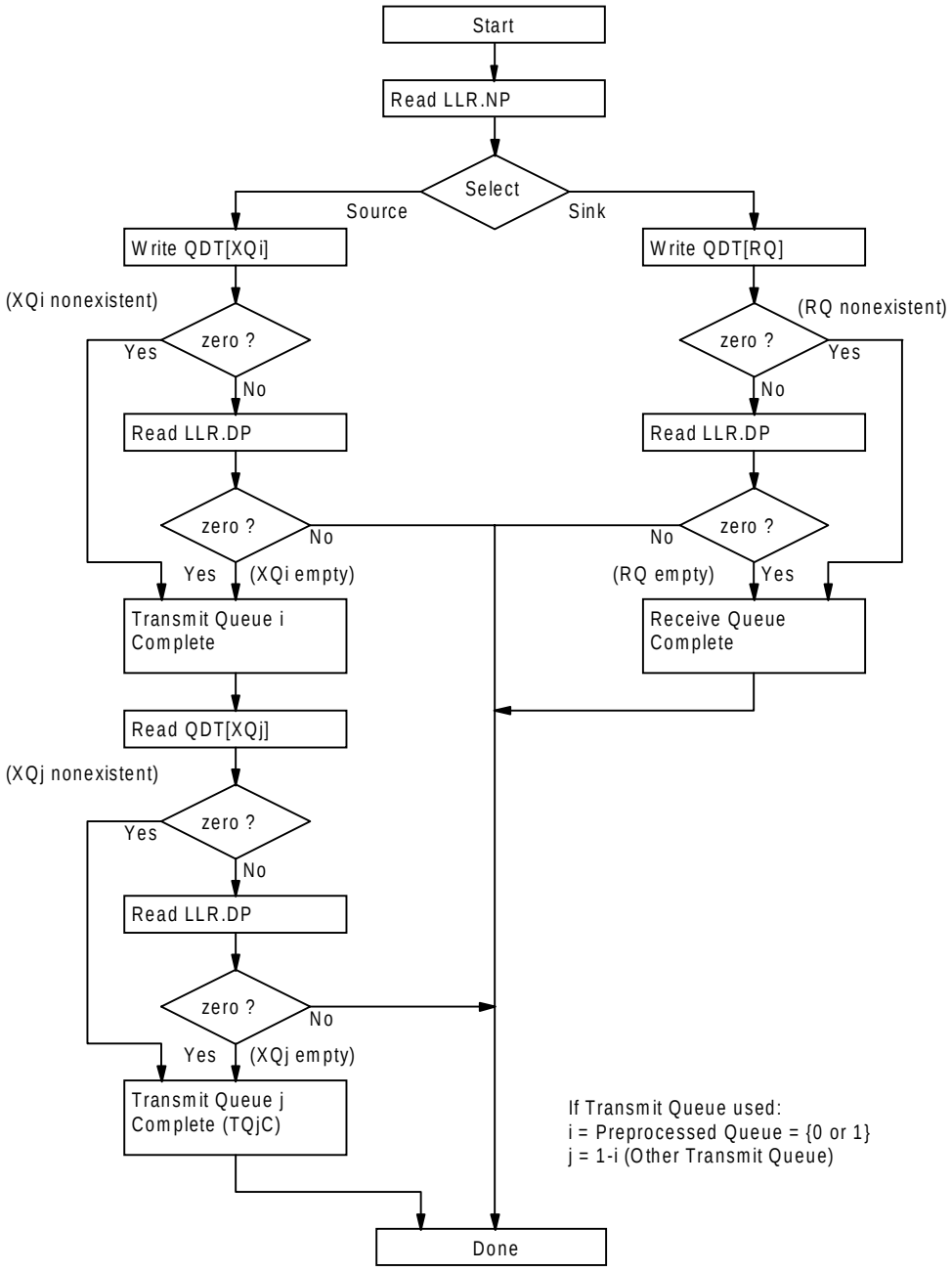


Figure 3.5: Queue Postprocessing

Notice on Event Polling with Queued Messages:

Normally, the CPE*i*-bit (*i*=ET) clears both EA*i* (announced event) and PAR*i* (participation), when the corresponding source port has been processed. If message queues are attached, two cases may occur:

- If at least one of the two Transmit Queues contain one or more outgoing messages, then only PARi is cleared. The event maintains announced for the next Event polling round. However, the MVBC will no longer participate in the current Event Polling round.
- If both Transmit Queues contain no more messages after the transfer, the both PARi and EAi are cleared.

3.8 Yet unsupported Master Frames (F-Codes 5-7, 10-11)

The MVBC will receive these Master Frames properly, but will take no additional actions afterward. All ports remain untouched.

3.9 Special Features

3.9.1 Forcing Process Data

A bitwise forcing mechanism is supported in order to override transmitted or received data with fixed values. Forcing can be enabled portwise by setting the FE-bit in the PCS to '1'. If forcing is enabled, then every access to the TM Data Area is extended by two additional accesses to the Force Table (see sections 2.8.5 and 2.9.13):

- Read Force Mask Word *i*
- Read Force Data Word *i*
- Read/Write Data Word *i*

Attention: Forcing is restricted to Process Data only. The user is required to keep the FE-bit inactive for all other forms of transfers. Otherwise, undefined behavior may occur.

Attention: Increased traffic between MVBC to TM due to accesses to the Force Table.

3.9.2 Disabling Ports Temporarily

The MVBC provides a reservation mechanism to assure consistent data reception to systems which do not comply with any real-time requirements, even though multi-tasking is supported. The mechanism is realized by temporarily disabling the sink-port while one or more tasks are retrieving data from the corresponding Data Area inside the TM. However, depending on the time required to read the data, one or more successive Process Data frames, which arrive at a later time, may be discarded entirely.

This mechanism uses a counter approach which has been derived from classical SW semaphores used for interprocess communication in multi-tasking operating systems. This feature allows multiple concurrent tasks to reserve and access the same port.

The PCS Word 1 provides a byte for the counter (Disable/Enable Counter, $DEC_{7..0}$) and a PTD-bit (Port Temporarily Disabled). The initial values are zero. The MVBC is responsible for the following work: If DEC is nonzero, then the MVBC will receive one more frame and stores it in *Not*(VP). At the moment PCS Word 1 is written back, VP is inverted as usual and PTD is now active. If both $DEC_{7..0}$ and PTD are nonzero, then no data will be received into the port processed. However, if DEC is zero again, data will be received again and PTD is cleared.

The software must do the following in order to receive consistent data:

1. Increment $DEC_{7..0}$ by 1
   - Read $DEC_{7..0}$ (3 indivisible operations)
   - Increment value
   - Write $DEC_{7..0}$
2. Retrieve data from TM to local host memory
3. Decrement $DEC_{7..0}$ by 1
   - Read $DEC_{7..0}$ (3 indivisible operations)
   - Decrement value
   - Write $DEC_{7..0}$

In/Decrementing DEC must not be interrupted by any other task. However, step 2 may be interrupted or temporarily suspended.

Attention: Not more than 255 tasks may read data from the same port.

3.9.3 Data Transfer Interrupts

Data Transfer Interrupts ($DTI_{7..1}$) can be configured in each port by setting the $IE_{2..0}$ to a nonzero value. A value of 7 will enable the Automatic Comparison Mechanism (see section 3.9.4). Interrupts will be issued after processing the port (data transfer plus TACK bits plus final read and write access to PCS word 1) has been completed.

3.9.4 Automatic Comparison Mechanism

The Automatic Comparison Mechanism is activated if Data Transfer Interrupt 7 (PCS: IE2..0='111') is enabled and the port is declared as an active sink. In this case, the MVBC issues interrupts only if the received data differs from the previously received data. This powerful feature can be used to avoid time-wasting reevaluations if the received data did not change. While data is transferred from the Receive Buffer (RXB), following steps are performed for each word transferred:

1. Read previous word from Data Area, pointed by VP
2. Check whether they differ
3. Write received data to Data Area, pointed by Inv(VP)
4. Remaining words: Proceed with step 1 if no difference is yet found
   Proceed with step 3 if difference have been detected

Since no page mechanism exists for queued data (VP), Comparison is made on the same data block. Check Sequences will not be compared, even if TWCS is active. If Forcing is enabled, then forcing will be performed before the Comparison will be made. The MVBC assumes that the previous data has also been forced. Following TM access pattern is used:

- Read Force Mask Word *i* (if forcing is enabled)
- Read Force Data Word *i* (")
- Read Data Word *i*, pointed by VP (as long data is still similar)
- Write Data Word *i*, pointed by *Not*(VP)

Attention: Increased traffic: Two TM accesses are made for each transferred data word. Four accesses are made if Forcing is activated, too.

3.9.5 Synchro Port

One port in the LA space serves as the Synchro Port if it is declared as a sink port. Any data transfer to a Synchro Port causes the strobe signal "STROBE\" to be activated for three clock cycles (125 ns). This port is useful to trigger external devices (i.e. an A/D-converter) or to synchronize bus participants. Since the Synchro Port is assigned to the highest Port Index, the size of the port is limited to one dock (max. 4 words).

MVBC Mode	MCM	Access to Synchro Port	Max. port size
Class 2/3/4 Mode	0, 1	Port Index 255	4 words (1 dock)
Class 2/3/4 Mode	2	Port Index 1023	4 words (1 dock)
Class 2/3/4 Mode	3, 4	Port Index 4095	4 words (1 dock)
Class 1 Mode	(not applicable)	Port 15 [1]	1 word

[1] No strobe signal is generated when two or more consecutive ports are accessed where the last port coincides with port 15. Example: F-Code on port 14 to access both ports 14 and 15 does not activate the strobe signal.

Table 3.12: Synchro Ports

3.9.6 Transfers with User-Supplied Check Sequences

During normal operation, the Check Sequence is generated inside the Encoder using the polynomial described in section 2.9.2. The user may configure selected ports where the MVBC shall obtain user-supplied Check Sequence from the TM instead. However, the Decoder will always use the same generator polynomial to check received data. Any deviating Check Sequence results to a CRC error with all its consequences.

Applications: Intentional transmission of erroneous data, i.e. for CRC mechanism tests and Bridges

In the TM, both CS octets (for both corresponding pages in the Data Area) are stored in PCS Word 3. Write accesses use the read-modify-write approach to update one such octet because the MVBC supports 16-bit TM accesses only.

If frames containing 8 or 16 data words are transferred, then the CS is accessed after 4 every words. If the frame contains 4 or less words, then the CS is accessed after all words have been transferred.

Attention: When TWCS is enabled, then the number of TM accesses increases by $max(1,\lfloor n/4 \rfloor)*a$. n = number of words transferred; a = 1 for transmission, 2 for reception.

3.9.7 Transferring Non-Numeric Data

Non-Numeric Data is understood as a character or byte string ordered in ascending addresss order. In the PCS, Word 0, all data must be declared with the NUM-bit. This will ensure data transfers in the correct byte and word order.

Attention: Messages (see section 3.7) are always non-numeric and must be declared as non-numeric data.

3.9.8 Write-Always Option

In some applications, i.e. gateways, transferring erroneous data intentionally may make sense. If the Write Always (WA)-bit (PCS, Word 0) is active, then the MVBC will process all erroneous data where the overall frame size has not been changed by noise or MVB failure. If TWCS is active, too, then the received (possibly mismatching) CRC will be stored along with the data.

3.9.9 Self Conversation

Self-Conversations occur when the MVBC processes two ports in the period one telegram is transferred. Typically, the first port is a source and the second port a sink. Only one port can be a source. However, an incorrect PCS configuration would represent a scenario where both ports act as sink ports. In case data is transferred, then both ports receive the data.

Common forms of self-conversations include

- Event Arbitration: An application, which runs on the same system where the Bus Administrator Software is running, announces an event.
- Self-addressed Message Data, broadcasting, broadcalling (F-Code 12)
- Self-addressed Device Status Poll (F-Code 15)
- Self-addressed Mastership Offer Poll (possible, but makes no sense; F-Code 8)
- Using of Test Ports for internal (IL=2) and external loopback test (IL=3, using short-circuited lines)

3.9.10 Test Ports

Under regular operating conditions, no self-conversation with Process Data (F-codes 0-4) can take place since only one port is available and must either be configured as a source or sink port. The test ports allow self-tests with transferring Process Data using internal or external loop-back. The maximum size is 16 words and the is located in the same memory locations where the Message ports (MSRC, MSNK) are located. The Test Source Port (TSRC) and Test Sink Port (TSNK) can be activated by setting the respective SCR-bits UTQ and UTS to '1'.

Appendix B indicates the conditions for every F-Code where Test Ports are referenced. The user is discouraged to use test ports for regular data transfers over the MVB.

Attention: In the current MVBC implementation, the Message and Test Ports are equivalent:
MSRC ⇔ TSRC; MSNK ⇔ TSNK

3.10 Master Frame Dispatcher

The MVBC provides an intelligent mechanism to dispatch individual Master Frames or a finite series of Master Frames from Master Frame Tables. The automatic dispatcher is suitable to send cyclic Process Data at precise time intervals; the manual mechanism is suitable to handle individual frames and sporadic data transfers such as messages.

Prerequisite: MAS-bit (SCR) is active, IL = 3 (full mode) or 2 (internal loopback test)

If the prerequisite is not met, then the MVBC will wait until MAS and IL have been set to the values shown above. The MAS-bit can be used to suspend (not cancel) sending Master Frames. Canceling all MF transmission can be achieved by writing a '1' to the CSMF-bit (Cancel Sending Master Frames, inside MR). Four different dispatching mechanisms are supported:

- SMFM Send Master Frames Manually
- SMFA Send Master Frames Automatically immediately
- SMFT Send Master Frames automatically at next Time Slot
- SMFE Empty Master Frame Table encountered, no MF's sent

For SMFA, SMFT and SMFE, the interrupt AMFX is asserted in order to inform the CPU that the addressed MF-Table has been processed.

3.10.1 Manual Mechanism (SMFM)

The manual mechanism transmits individual Master Frames. This mechanism has a lower priority than the automatic approach, so the BAS can append one Master Frame in advance while the automatic dispatcher is still busy working off a MF-Table. When this table has been finished, then the appended Master Frame will be transmitted, except if:

- No additional MF-table is signed up using advance request (see section 3.10.5) with SMFA (no waiting for next timer pulse)
- The BUSY-bit is still active (data transfer still in progress).
- Minimum spacing requirements between previous Slave Frame and current Master Frame are met (4 µs).

If any of these cases applies, then sending the manually requested MF will be postponed until both conditions are met. The BAS must undertake two steps to send a Master Frame: The MF body must be written into the Master Frame Slot (MFS, inside Service Area) and the transmission must be initiated by activating SMFM in the MR. When the three conditions noted above are met, then the MVBC reads the MF from the MFS and transmits it on the MVB. After the read-access, the interrupt

"Master Frame Checked" (MFC)

is asserted to inform that the BAS can write the next MF into the MFS and announce the next MF transmission with SMFM. The BUSY-bit in the MR will become active immediately and stay active until the entire telegram (Master + Slave Frame) has been transferred or a timeout has occurred. This bit is coupled with the "Slave Frame Checked" interrupt (SFC), however this interrupt will not be triggered if a Reply Timeout occurs due to a missing Slave Frame. Figure 3.6, part 1 illustrates few examples of SMFM.

3.10.2 Automatic Mechanism (SMFA)

The automatic mechanism is able to process MF-Tables containing up to 32 Master Frames. Automatic dispatching has a higher priority over manual dispatching, meaning that manual MF-requests submitted thereafter are postponed until automatic dispatching has been completed.

Two steps must be undertaken in order to transmit all Master Frames from one MF Table: First, the 16-bit pointer (Lower two bits are always zero) to the MF Table is written to the DPR. The pointer format is described in section 2.8.8. Next, the transmission is initiated by simultaneously specifying SMFA (Send Master Frames Automatically)

and 5-bit table size in the MR. Allowed sizes are 1-32. A size of zero means 32! See section 3.10.4 to send *empty* MF-Tables instead.

MF-transmission starts at the moment all of the following conditions are met:

- The BUSY-bit (see MR) is zero.
- The minimum frame spacing on the MVB must be reached.

The MFC-Interrupt is asserted for every obtained MF body. When the entire MF-Table has been finished, the interrupt

"All Master Frames Transmitted" (AMFX)

will be asserted. The AMFX-Interrupt is asserted when the last entry from the MF-Table has been retrieved. The BUSY-bit remains active until the last telegram has been processed entirely. Figure 3.6, part 2 illustrates an example of SMFA.

3.10.3 Timed Mechanism (SMFT)

The timed mechanism operates in a similar manner as the automatic mechanism. It allows sending Master Frames at precise time intervals. The MVBC waits until the Universal Timer 1 reaches zero before the MF-Table is processed. The Universal Timer 1 must be configured accordingly (active counter, valid value inside the Timer Reload Register 1).

The procedure to send timed MF's is similar as described in section 3.10.2. However, SMFT must be used instead of SMFA. The BUSY-bit will be activated at the moment the first MF is transmitted.Figure 3.6, part 3 illustrates an example of SMFA.

3.10.4 Empty MF-Tables (SMFE)

SMFE is used to specify empty MF-Tables. The MVBC waits until the Timer reaches zero and issues the AMFX-Interrupt immediately afterward. The AMFX-interrupt is needed to handle advance requests described in the following section. Figure 3.6, part 4 illustrates the SMFE-feature.

(Space below has been left blank intentionally)

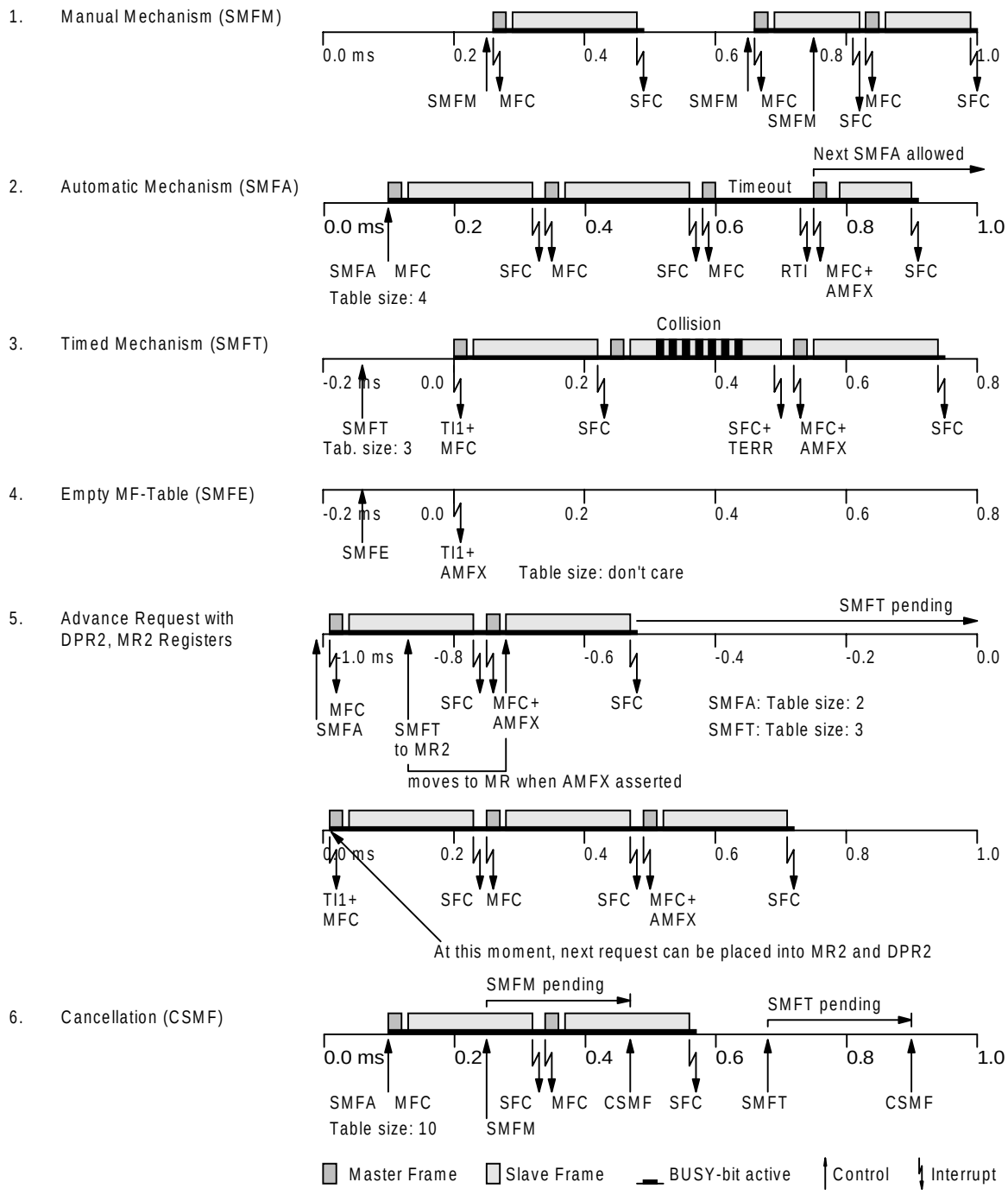


Figure 3.6: Behavior of Master Frame Dispatcher

3.10.5 Advance Requests

The host CPU can submit advance requests to the MVBC in order to transmit MFs on time without struggling with CPU interrupt service latency problems. This mechanism is necessary if the interrupt latency to service the AMFX-interrupt and to initiate the next MF is too big.

Advance request are made into the secondary Master and Dispatch Pointer Registers MR2 and DPR2. The contents of these registers are automatically transferred into MR and DPR when the AMFX-interrupt, originating from the current request, occurs.

Attention: If AMFX is masked, then AMFX will not pass an interrupt to the CPU but still perform the register transaction. The transfer takes place regardless if the Interrupt Logic is frozen or not.

DPR2 contains the pointer to the next MF-Table, and MR2 specifies length of that MF-Table and its transmission mode (SMFT, SMFA, SMFE) as it should be specified for MR. For obvious reasons, MR2 does not include the upper halfword containing control bits for Event Polling, CSMF and BUSY. MR2 and DPR2 permit full read and write access. The values can be modified if necessary.

At the moment the AMFX-interrupt occurs, the contents of DPR2 and MR2 are moved to DPR and MR respectively. MR2 is cleared afterward in order to avoid repeating the same request. The transfer from MR2 to MR is cumulative, meaning that pending requests inside MR (SMFA, SMFE, SMFM and SMFT) will not be cleared.

Figure 3.6, part 5 illustrates a few examples. First, SMFA is issued to MR to transfer two MFs. While the first telegram is transferred, SMFT is passed to MR2 to transfer three MFs. After the timer reaches zero, the next advance request may be issued.

3.10.6 Cancellations

The CSMF-bit (Cancel Sending Master Frames) cancels any current dispatching request. Any manual dispatching request (SMFM) can be cancelled before the MVBC has started accessing the Master Frame Slot (MFS) to transmit the MF. If an automatic dispatching request (SMFA) is cancelled, then the current telegram is processed entirely and no additional MF will be dispatched. Timed requests (SMFT) can be completely cancelled any time unless the timer reaches zero meanwhile. SMFE requests can be cancelled as long the timer has not yet reached zero.

The MVBC provides no means to reinstate cancelled requests. However, MF dispatching can be suspended by setting MAS to '0' (inside SCR) temporarily.

3.10.7 External Synchronization

Timed MF Transmission (with SMFT, SMFE) can also be initiated by the external signal II3\ so multiple MVBCs can orchestrate their MVBs synchronously. See section 2.9.16.2.

(Space below has been left blank intentionally)

4 HARDWARE APPLICATION SUGGESTIONS

4.1 MVBC with 16-Bit Traffic Memory

The following figure shows a typical schematic where the MVBC is running in Class 2/3/4 Mode. The Class_Mode-Pin must be connected to +5 V.

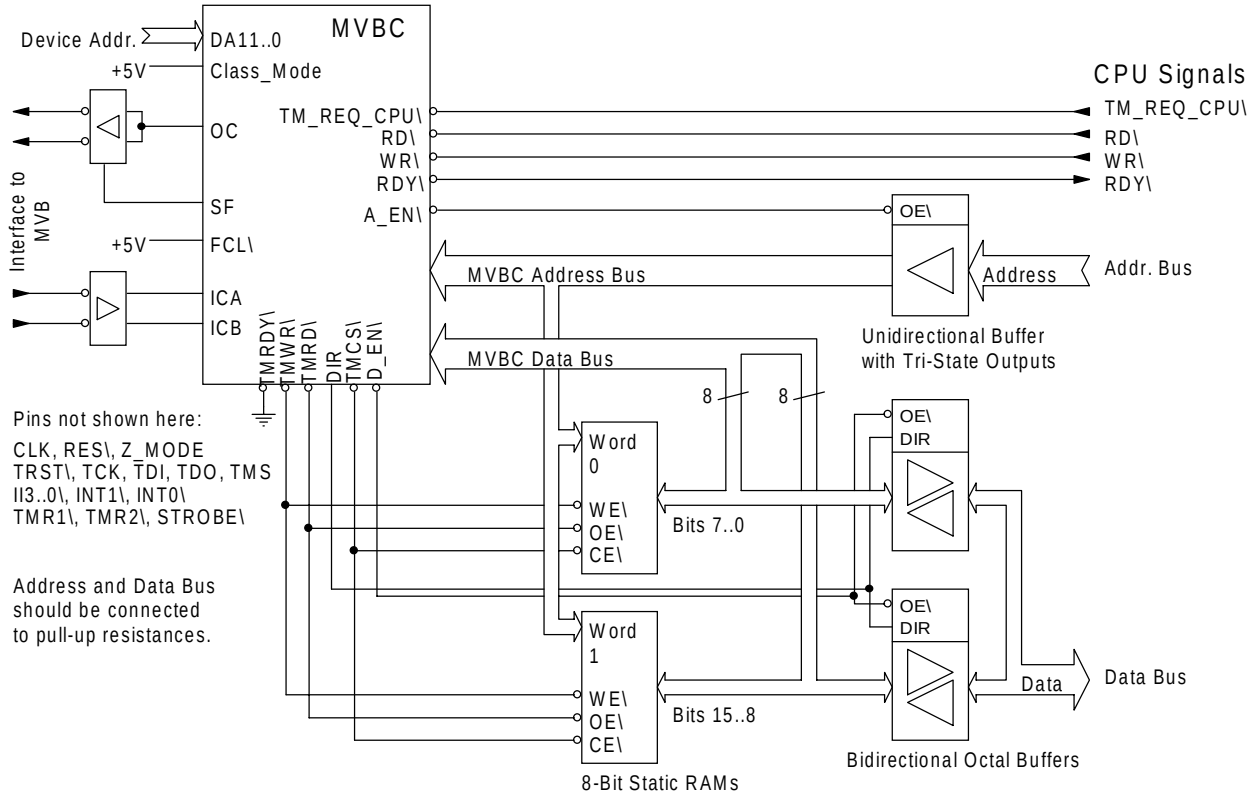


Figure 4.1: MVBC with 16-bit Traffic Memory

4.2 MVBC operating in Class 1 Mode

A diagram showing the MVBC operating in Class 1 Mode is shown on the next page. Class 1 Mode operation is enabled when the pin CLASS_MODE is connected to ground. In simple applications, the MVBC operates without assistance of a host CPU or microcontroller. The following default connections must be considered:

- RD\, WR\, TM_REQ_CPU\ are tied to +5 V
- $II_{3..0}$\ are tied individually to +5V or ground in order to configure reply timeout and TM access wait states.
- INT1\, INT0\, RDY\ are open
- Only $DA_{11..4}$ are used to define the Device Address. $DA_{3..0}$ are used to supply information for the Device Status Report.

A microcontroller may be desirable when following features of the MVBC are to be utilized:

- Visibility of internal interrupts, i.e. due to telegram errors, timeouts, etc.
- Using the Universal Timers
- Overriding Device Address
- Accessing the ports directly over the same data bus

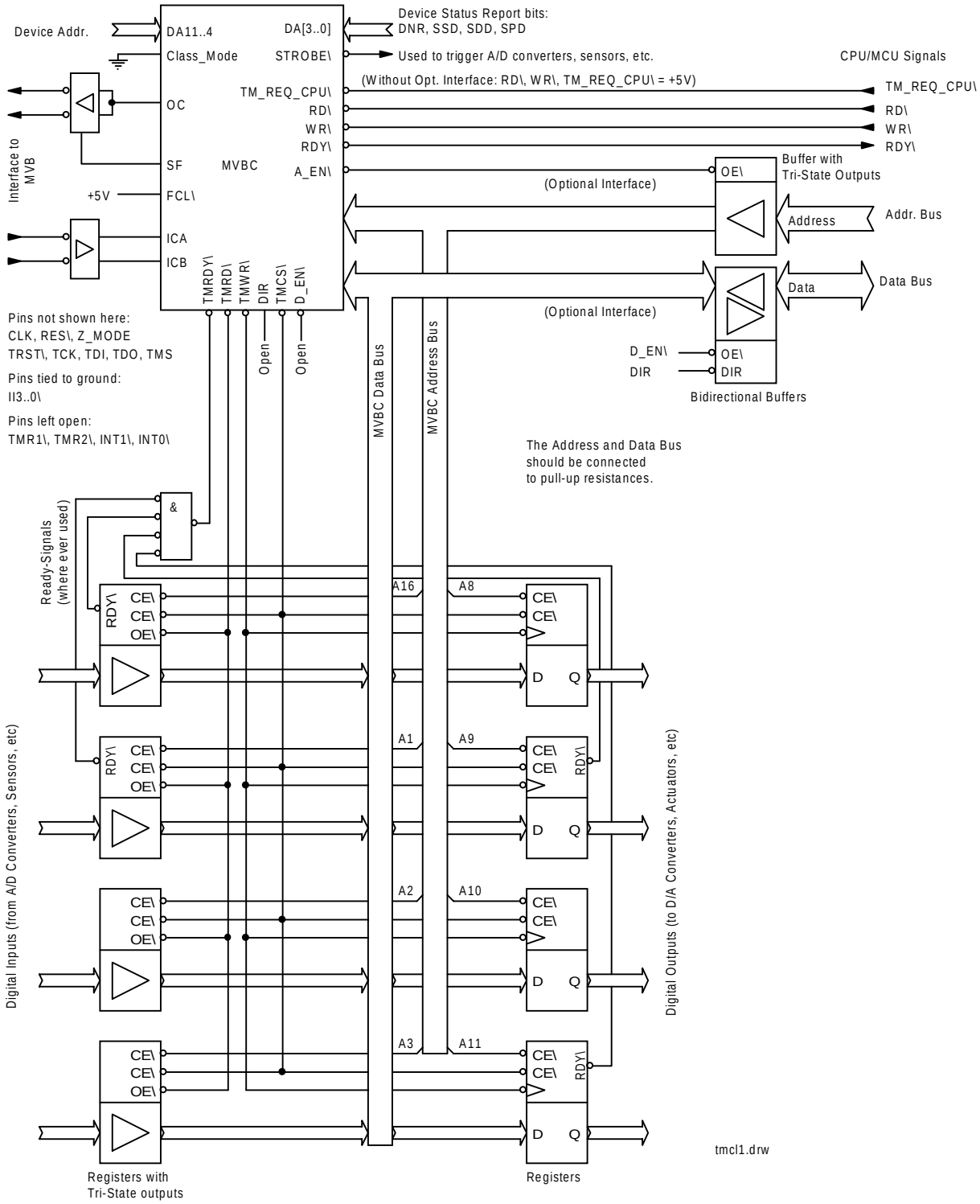


Figure 4.2: MVBC Operating in Class 1 Mode

MCM=0 is assumed when accesses to Internal Registers in Class 1 mode are made. In order to avoid access confusions, every port should check the address bit in combination with TMCS\. When the ports are accessed by the host over the same data bus, then the user must be aware not to choose addresses which collide with the Service Area where the Internal Registers are located. The danger is eliminated when at least one of the upper three address bits A19..17 are nonzero.

4.2.1 Class 1 Mode Operation with "Traffic Memory"

The following figure shows a simple application of the MVBC using a 2 KB Dual-Port RAM and a primitive interrupt mechanism. This example requires two sink ports and one source port for the data transfer, and port 15 to supply the STROBE\-signal. Two data transfers are required to perform a read or write access. The first transfer is used to supply the 8-bit address and, if a write access is intended, valid 8-bit data. The second transfer is used to read the data along with the previously supplied address. Using multiple ports allow applications with larger DPRAMs.

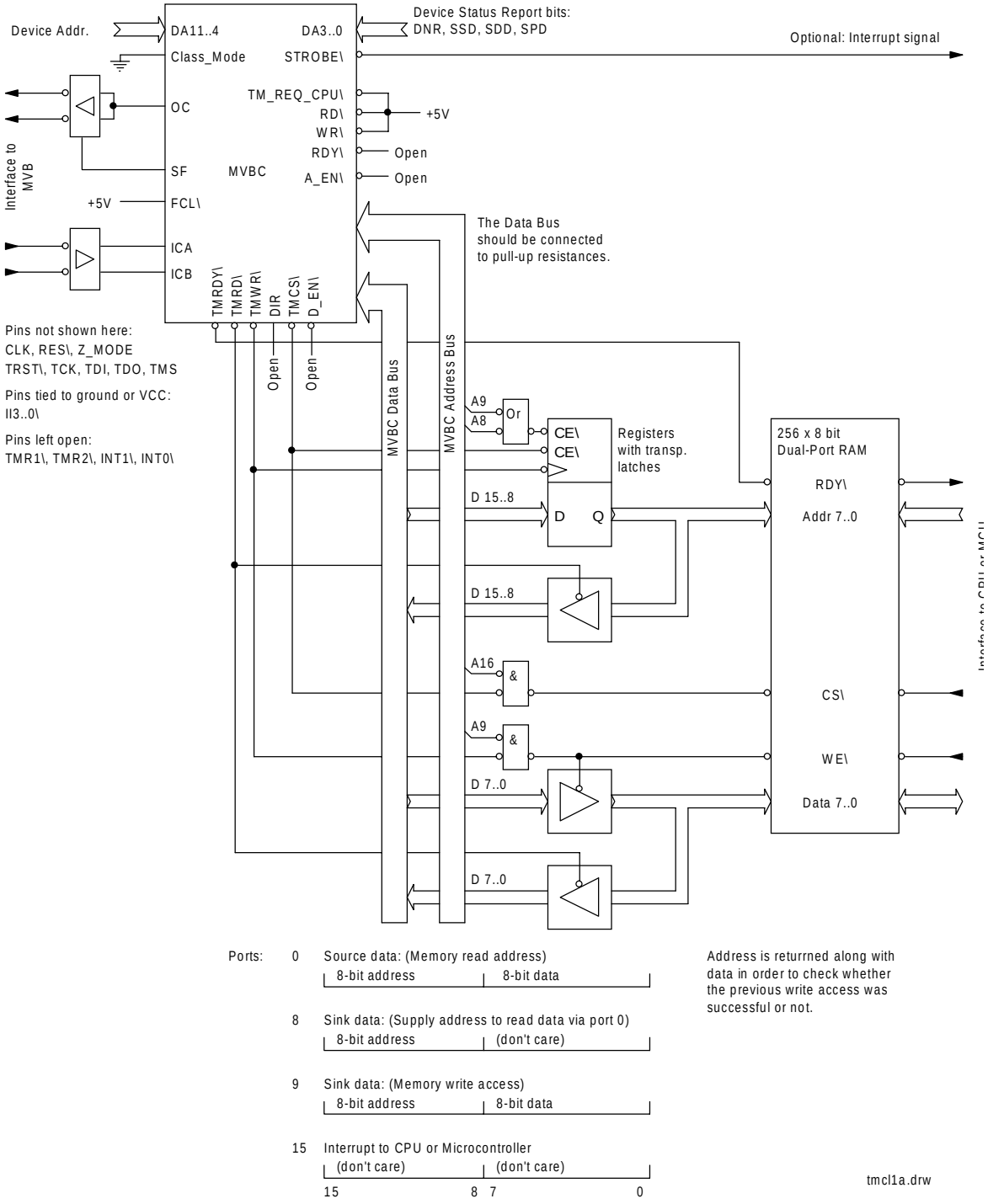


Figure 4.3: MVBC in Class 1 Mode and Dual-Port RAM

4.3 Interrupt, Timer and Strobe Signals

The MVBC provides two outgoing interrupt lines which should be connected to separate interrupt inputs at the host CPU. If only one input is available, then the two lines can be combined with an AND-gate. In this case, the interrupt handler software must check both ISR0/IVR0 and ISR1/IVR1 when an interrupt has occurred. On the other hand, the external interrupt inputs (II3..0) are available for any application.

If the host CPU provides more than two interrupt inputs, and must run a real-time system which is triggered by one Universal Timer, then the timer signal (TMR1\ or TMR2\) should be connected directly to an interrupt input in order to achieve minimum latency.

The STROBE\-Pin is a general-purpose active-low signal which may be used to interrupt the host CPU, synchronize an Universal Timer or trigger peripheral devices such as A/D converters. An easy way to let the STROBE\-signal cause an interrupt is to connect STROBE\ to one of $II_{3..0}$\.

4.4 Other Pins

Device Address $DA_{11..0}$:
The MVBC does not latch the DA at power-up. Therefore, a valid DA must be supplied while the MVBC is in service. DIP-Switches or hexadecimal rotary switches are strongly recommended. In Class 1 Mode, $DA_{3..0}$ supply information about Device Status (DNR, ERD, SDD, SSD) which are not sampled. Valid values must be supplied at all time.

Z-Mode\: During normal operation, this pin is connected to +5 V.

JTAG pins: If the Boundary Scan is not used, then all JTAG inputs except TRST should be connected to +5 V. TRST is connected along with RESET\ to the asynchronous reset signal.

(Space below has been left blank intentionally)

5 PROGRAMMING GUIDELINES

This chapter gives a brief summary to initialize and operate the MVBC.

5.1 MVBC Initialization

The MVBC including JTAG Boundary Scan hardware must be reset with the asynchronous reset (RES\, TRST\). After that, the MVBC will be operating with Initialization Level IL=0 (Reset Mode) and MCM=0.

In this mode, the Traffic Memory is accessible, but all registers except the IL-bits (Initiation Level) of the Status Control Registers will be held at their initial values.

Step 1: Set IL to 1 to allow MVBC configuration

If a different Memory Configuration Mode should be used, then the configuration should be done now. If possible, the offset for Message Queues and MF-Tables should be defined.

Step 2: Write MCR to define MCM, QO and MO

If MCM is nonzero, then the Service Area has moved to its final position and shall no longer be moved around! If the TM works faster, then the number of waitstates should be reduced. Additional SCR parameters, such as Timeout Coefficient, Arbitration Strategy, should be defined. The MAS bit must be active for Bus Administrators. If Event Polling is supported, then RCEV must be active.

Step 3: Initial definition of SCR (but keep IL at 1)

Step 4: Configure and activate Universal Timers

Step 4 may be necessary if the host system requires timer pulses or interrupts. Example:

Timer 1: Basic Time Slot (i.e. 1 ms)

Timer 2: Timer Pulse for Real-Time Operating System

Step 5: RAM test on Traffic Memory (if required by user)

Before Process Data ports should be defined in the logical address space, the Port Index Table should be formatted in order to keep all unused ports inactive. The easiest solution is to clear the entire LA-PIT to zero so dock zero will be addressed by default. The PCS Word 0 and Word 1 of dock 0 shall also be zero in order to keep the port inactive.

Step 6: Format LA-PIT

Step 7: Format DA-PIT (same manner as with LA-PIT, if MCM>0)

Step 8: Clear PCS Word 0 and 1 of all Physical Ports

The last step assures that all Physical Ports are cleared. This is a safety measure in order to avoid unexpected MVBC behavior if not all Physical Ports are configured properly.

Step 9: Read original Device Address, or modify it

If the MVBC is connected to one MVB line only, the Decoder must be informed so.

Step 10: Switch lines if necessary, then set SLM to '1'

A read access to the Decoder Register shall be made afterward to assure SLM is set and the correct line is selected.

Step 11: Self-Test, if required

A few ports can be configured in order to transfer one or more telegrams over the internal loopback lines. At the end of the test, the initial PCS settings should be restored. Most MVBC functions, i.e. transfers with all supported F-Codes, can be tested, but this may most likely limited by the time available to run the test before the system can go into full operation.

Step 12: Interrupt Controller

The interrupt controller should be activated for all interrupts which are used by the software. This step can take place while application-specific interrupt service routines are attached to the interrupt handler software. For all interrupts, which are not serviced, the corresponding bits in the Interrupt Mask Register shall be kept at zero.

5.2 TM and Port Initialization

Step 13: Initialization of all Process Data ports

All affected ports are located in the LA space. For every port, a Port Index must be defined and stored in the Port Index Table. Using the Port Index, the PCS must be configured and a defined value be written into the Data Area. If Forcing is active, then the Force Table must be configured accordingly.

Step 14: Initialization of Message Data Support

The affected ports are MSRC and MSNK. The PCS must be configured accordingly. Section 5.3 gives details on installing correct data structures for message queues.

Step 15: Initialization of ports for supervisory transfers

The Slave devices require proper configuration of EF0, EF1 and FC15. EF0 and EF1 are necessary to permit participation on Event Polling. FC15 must contain a correct initial Device Status Report.

Bus Administrators also require EFS, FC8 and MOS to be configured. Systems, which are in charge of collecting Device Status Reports, must initialize the DA space. This is done similarly as with the LA space.

Step 16: Bus Administrator: Master Frame Tables

This applies to Bus Administrators. If the MVBC shall transmit Master Frames automatically, then Master Frame Tables should be prepared. See section 5.4 for details.

Step 17: Sink-Time Supervision

Here, the maximum range of used LA ports must be known. The Sink-Time Supervision stays inactive until loopback test (IL=2) or full operation (IL=3) is enabled. The user must be careful not to choose a small time interval when the number of ports to be supervised is high. See Figure 2.22 in section 2.9.17.2. The TM traffic should not exceed 10%.

Step 18: Full Operation: Set IL to 3. Good Luck!

Step 19: Bus Administrator: Check for foreign active BA

The Bus Administrator must check if bus traffic is already present. If not, then a MF shall be sent. If a collision occurs, then it shall wait for a unique time period in order to watch the MVB and check if other masters are already active. If the time has elapsed and no Master Frame has been received meanwhile, then the Bus Administrator can start sending Master Frames. Otherwise, it must wait until it has been polled for Mastership Offer.

5.3 Message Queue Data Structures

If Message Queues are used, then the user must supply an adequate data structure the MVBC can work with. The MVBC considers following rules:

The MVBC treats nonexistent queues similarly as empty queues. The user may signify a nonexisting or empty Transmit Queue with one following parameter settings:

- Both QDT[XQ0] and QDT[XQ1] are zero
- First Data Pointers (DP) of both queues are zero
- Combinations of above

The empty Receive Queue can be specified in a similar manner:

- QDT[RQ] is zero
- First data Pointer (DP) is zero

The last record in a Transmit Queue can be specified with one of following parameter settings:

- Next Pointer (NP) in last LLR pointing to last valid data block is zero, in both Transmit Queues
- Data Pointer (DP) in subsequent LLR is zero, in both queues
- Combinations of above

The end of a Receive Queue can be specified in a similar manner:

- Next Pointer (NP) in last LLR is zero
- Data Pointer (DP) in subsequent LLR is zero

The user is free to choose linear linked lists or ring buffer structures. At least, the chosen data structure must be established before Message Data transfers start.

If the linear list solution is preferred, then the user should establish two linked lists for each queue where one list is attached to the MVBC while the other is attached to the software interface. 2*n* blocks are necessary to realize an effective buffer with capacity of *n* blocks. On the other hand, the ring buffer solution uses the memory more efficiently and requires only *n*+1 blocks.

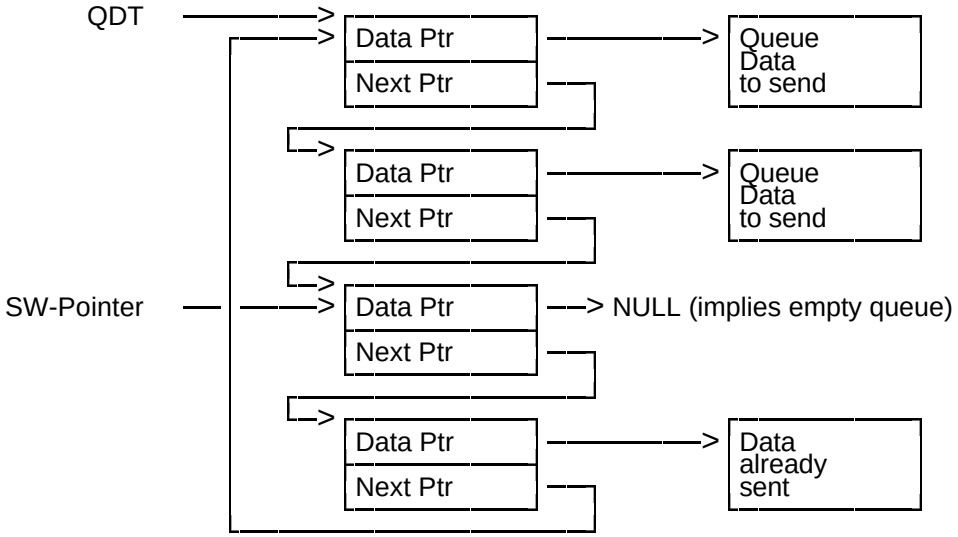


Figure 5.1: Ring-Buffer Solution for Message Queues

5.4 Master Frame Tables

Master Frame Tables allow the MVBC to send small series of up to 32 Master Frames automatically. These tables are primarily intended to handle cyclic Process Data. If the slot time is 1 ms, then up to 15-20 telegrams transferred within each time slot. One cycle contains a defined number of time slots. This depends on the biggest time interval for Process Data. If the interval is 1024 ms (210), the cycle contains 1024 time slots. Each time slot is associated with one MF-Table. However, one MF-Table can be shared among multiple time slots if the same Process Data variables must be transferred.

1. Assume ca 16 Master Frames per interval are exchanged. The size of each table ends up to be 16 words (some may require 8, 24 or 32 word blocks) -> 32 byte average.

2. 1024 x 32 bytes = 32 KB

3. A table containing frame sizes is required. This table may either be located in the Traffic Memory or in CPU local RAM workspace. The size is 1024 x 8 = 8 KB.

4. A MF Pointer Table is necessary to store the start addresses to every MF Table. The MF Pointer Table may be located in the TM or in the CPU local RAM workspace. The size is 1024 x 16 = 16 KB.

Result: Storing all Master Frame Tables requires roughly 56 KB. If several tables are identical (say 50% of them), then the duplicate MF-Tables can be discarded so the MF Table size reduces from 32 KB down to 16 KB. The overall memory requirement reduces from 56 KB down to 40 KB. MF-Tables to be distributed over all unused TM fragments. For the 1 MByte TM, all MF-Tables must be located within one aligned 256 KB block. The following figure summarizes the data recommended data structure:

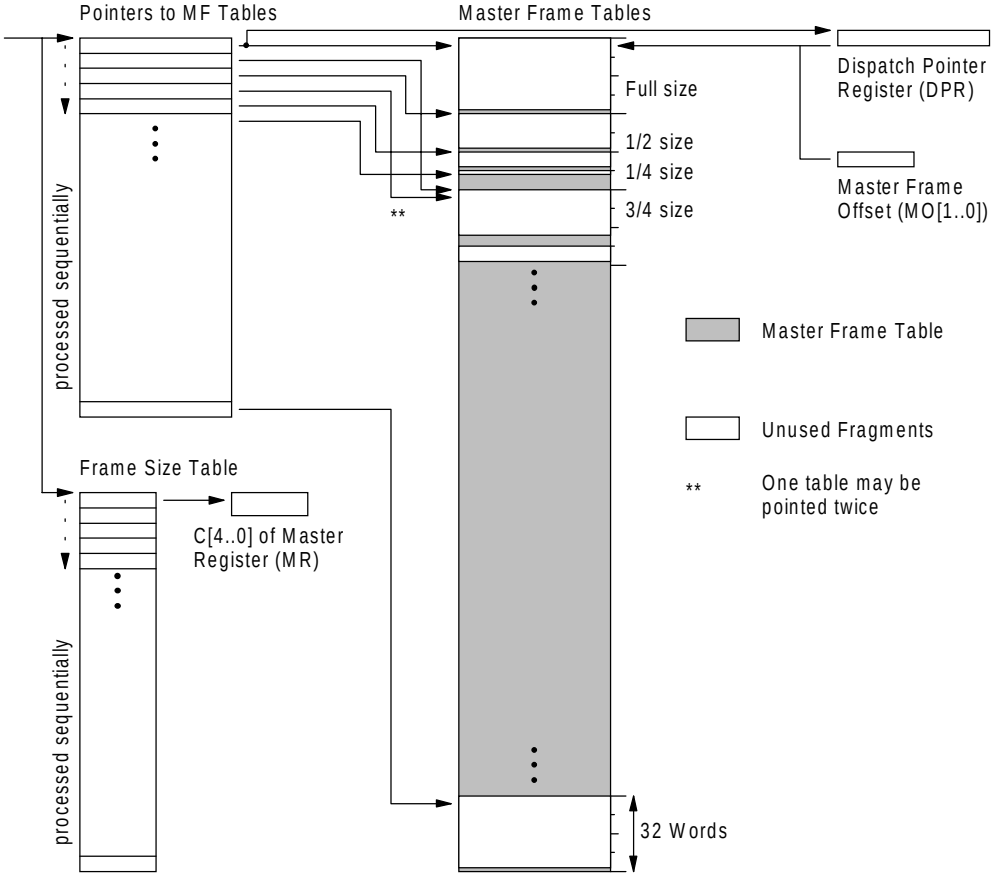


Figure 5.2: Recommended Data Structure for Managing MF-Tables

Insufficient TM Space for MF-Tables:

If the MF-Tables do not fit into the TM, then one would suggest to implement some sort of intelligent *caching mechanism* in order to keep additional TM traffic at a minimum. First, the programmer should identify the MF-Tables which are used frequently. These tables should reside in the TM permanently. Less frequently used MF-Tables should be copied from CPU local RAM workspace to the TM upon request.

6 TECHNICAL DATA

6.1 Environment

Operation beyond following temperature limits may impair the lifetime of the device:

Operating Temp.: -40 °C ... + 85 °C (Industrial grade range)

Storage Temp.: -40 °C ... + 125 °C

6.2 Mechanical Data

100 pin QFPP quad flat pack plastic with the following dimensions:

Symbol	Unit	Min	Typ	Max
A	mm			3.05
A1	mm	0.05	0.10	0.35
A2	mm	2.55	2.70	3.05
b	mm	0.20	0.30	0.40
C	mm	0.13	0.15	0.20
D	mm	19.85	20.00	20.15
E	mm	13.85	14.00	14.15
e	mm	--	0.65	--
HD	mm	22.8	23.6	24.4
HE	mm	16.8	17.6	18.4
L	mm	0.3	0.8	1.2
L1	mm	1.4	1.8	2.2
y	mm	--	--	0.15
ZD	mm	--	0.58	--
ZE	mm	--	0.83	--
Theta	° (angular)	0	5	10
n	mm	--	100	--
x	mm	--	--	0.13

Table 6.1: Mechanical Data

Notes:

- Chips are delivered in plastic packages.
- All dimensions are given in mm, which is controlling dimension
- The value of theta is measured in degrees

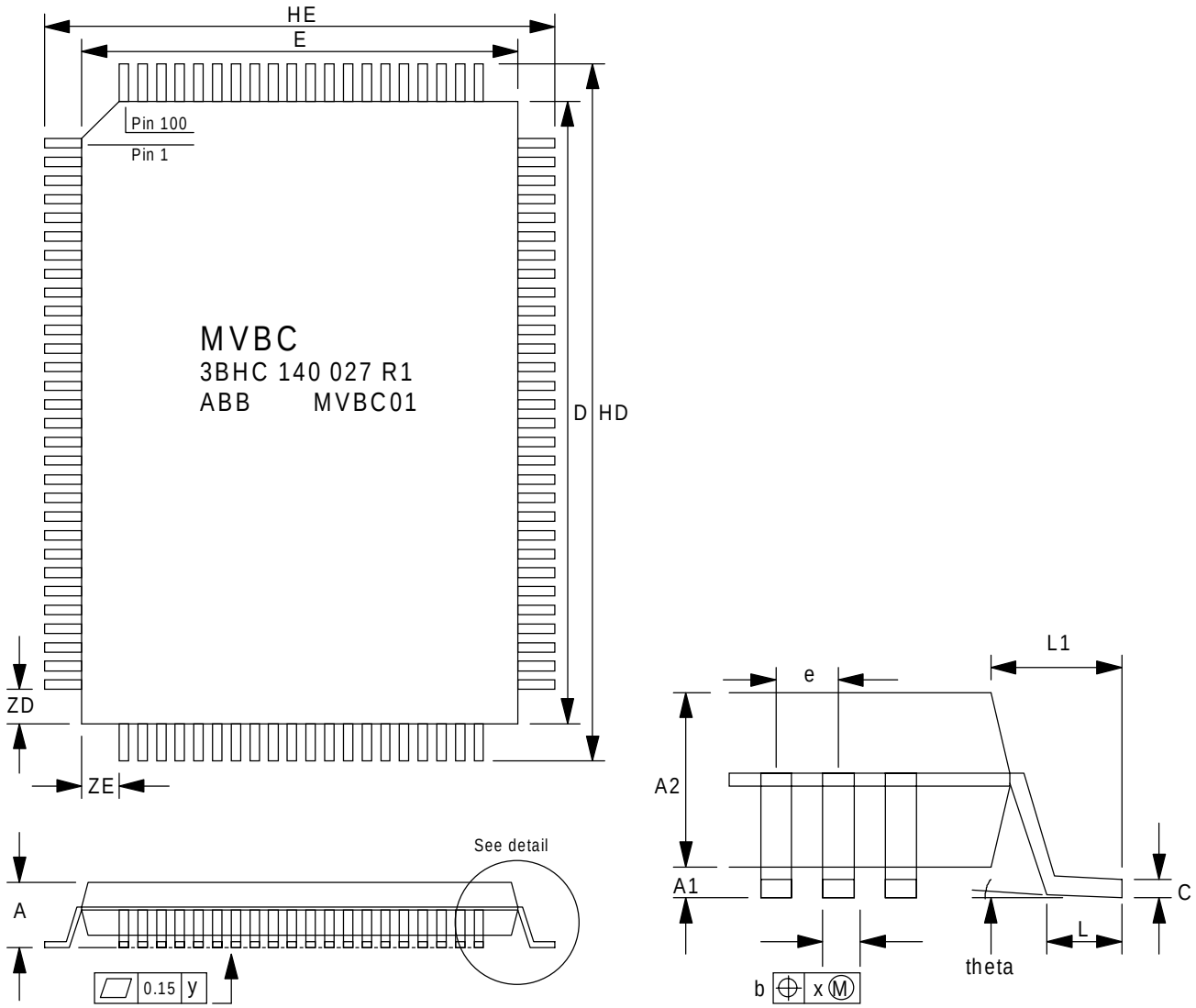


Figure 6.1: 100-Pin QFPP

6.3 Absolute Maximum Ratings

Warning: Exceeding absolute ratings may lead to permanent damage of the MVBC.

Power Supply Voltage: -0.5 V ... +7 V

Supply Voltage relative to Vss to any pin except Vcc:
-0.5 V to Vcc+0.5V

Maximum output current: 16 mA (short circuit load)

Soldering temperature: 235 °C
See also table on section 9.1.

Operating temperature range: -40 ... 85 °C
Storage temperature: -55 ... 150 °C

Total power dissipation: $P_t = P_c + P_p = 672$ mW
P_c = Power dissipation in core
P_p = Power dissipation in pads

These are calculated maximum power dissipation figures over the entire industrial temperature range between -40...+85°C and entire supply voltage range between 4.5 and 5.5 V.

Total current draw: $I_t = P_t / V_{DD} =$ 134 mA

6.4 Recommended Operating Conditions

Description		Test Condition	Min	Typ	Max	Unit
Supply Voltage Vcc	V_{CC}	-	4.50	5.00	5.50	V
Supply Voltage Vss [Ref.]	V_{SS}	-	0.00	0.00	0.00	V
Input Voltage, High	V_{IH}	-	2.00	-	5.50	V
Input Voltage, Low	V_{IL}	-	-0.50	-	0.80	V

Table 6.2: Recommended Operating Conditions

6.5 DC Electrical Characteristics

Default conditions: V_{DD} = +5 V, T = 25 °C, V_{SS} = 0 V (Ground)

Description		Test Condition	Min	Typ	Max	Unit
Output Voltage, High	V_{OH}	I_{OH} := -4 mA	2.40	-	-	V
Output Voltage, Low	V_{OL}	I_{OL} = +4 mA	-	-	0.40	V
Input Current, High	I_{IH}	$V_{in} = V_{SS}$	-	-	-10.0	µA
Input Current, Low	I_{IL}	$V_{in} = V_{DD}$	-	-	10.0	µA
Output Current, High	I_{OH}	$V_{out} = V_{OH}$	4.00	-	-	mA
Output Current, Low	I_{OL}	$V_{out} = V_{OL}$	4.00	-	-	mA
Tristate output leakage current	I_{OZ}	$V_{out} = V_{DD}$	-	-	10	µA
Fan-Out (74LS inputs)			10	-	-	inputs
Input Capacitance	C_{in}	including package	-	-	10	pF
Output Capacitance	C_{out}	including package	-	-	10	pF
Transceiver Cap.	$C_{i/o}$	excluding package	-	-	10	pF

Table 6.3: DC Electrical Characteristics

6.6 AC Electrical Characteristics

Description		Test Condition	Min	Typ	Max	Unit
Input rise time	t_R	0.8 - 2.0 V levels	-	-	100	ns
Input fall time	t_F	0.8 - 2.0 V levels	-	-	100	ns
Output pad propagation delay	t_{TLH}	C_L=50 pF	-	-	11.60	ns
Output pad propagation delay	t_{THL}	C_L=50 pF	-	-	11.78	ns

Table 6.4: AC Electrical Characteristics

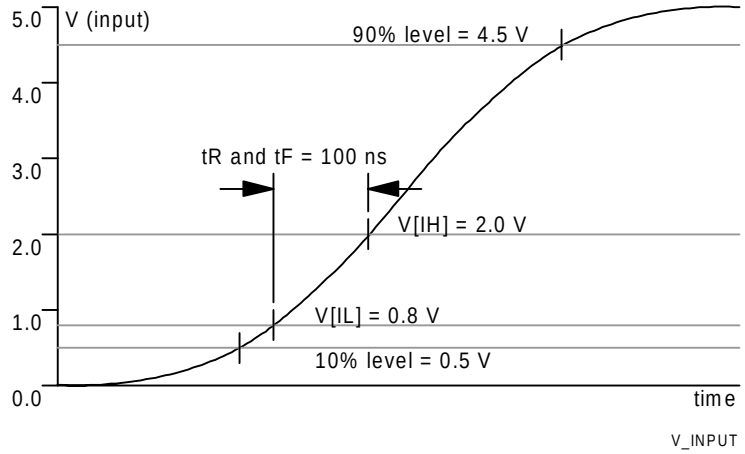


Figure 6.2: DC Levels

Further AC characteristics: see chapter 7: Timings.

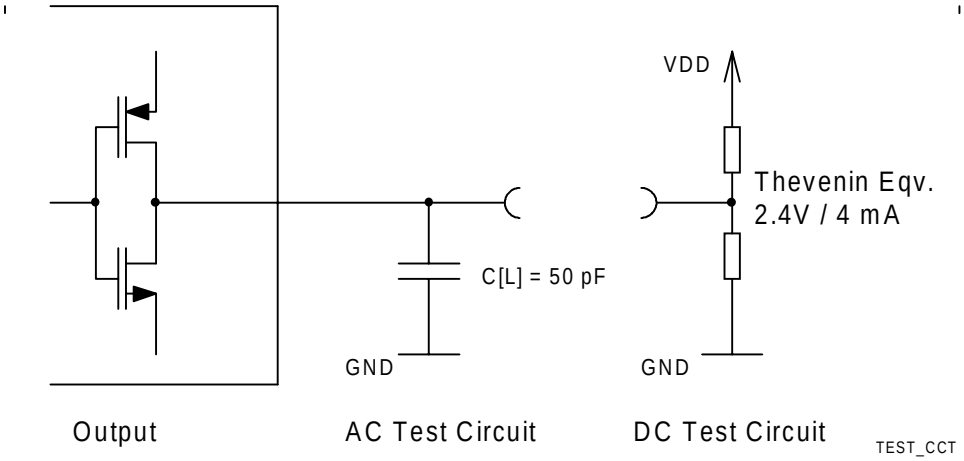


Figure 6.3: AC and DC Test Circuits

7 TIMINGS

Attention: All timings have been derived from timing analysis and simulation tools at worst-case conditions (50 pF load, +85 °C, minimum allowed supply voltage, expected worst-case fabrication lot, process corners etc.). If the capacitive load is less than 50 pF, then the propagation delays through output pads reduce by a factor of 0.103 * (50 pF - C_L). If the capacitive load is higher than 50 pF, then the propagation delays through output pads increase by a factor of 0.146 * (C_L - 50 pF).

7.1 General Timings

Clock Signal:

The MVBC can operate both synchronously and asynchronously with the host CPU. Asynchronous operation is understood where the MVBC and host CPU are attached to separate clock signal oscillators. For asynchronous operation, greater care must be given on the timing of the TM and register access control signals (TM_REQ_CPU\, RD\, WR\, etc.), address and data bus. Mechanism have been implemented to avoid internal metastability situations when control signals change at inconvenient periods with respect to the rising clock edge.

Synchronous operation allows a more time-efficient communication between the MVBC, TM and host CPU. In this case, the clock frequency of the host CPU is not limited to 24 MHz only. A fraction or a multiple of this frequency can be used instead (i.e. 8, 12, 16, 24, 36, 48 MHz).

Symbol	Parameter	min	typ	max	Unit
tCLCK	Clock Signal Low Time	16.7	20.8	25	ns
tCHCK	Clock Signal High Time	16.7	20.8	25	ns
tCKHL	Clock Signal Fall Time (level V_{IH}, V_{IL})	-	-	5	ns
tCKLH	Clock Signal Rise Time	-	-	5	ns
tCKIN	Clock Input Period [1] (at level V_{IT}) [2]	-	41.7	-	ns
Duty Cycle	Clock signal level ratio	40	50	60	%

[1] The clock period will affect data transfer rate and all other timing parameters such as Reply Timeout, Bus Timeout, etc.

[2] VIT applies to middle voltage level, VIL < VIT < VIH. See also figure 7.1.

Table 7.1: Timing Symbols for Clock Signal

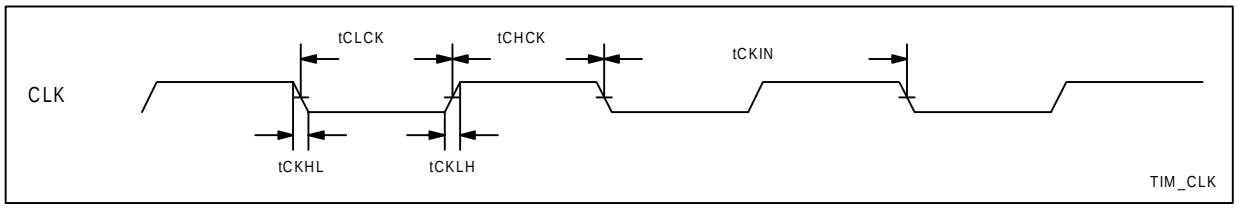


Figure 7.1: Clock Signal

VIT applies to middle voltage level, VIL < VIT < VIH and is used as middle voltage reference points for all further timing diagrams.

(Space below has been left blank intentionally)

General Input and Output Signal Delays:

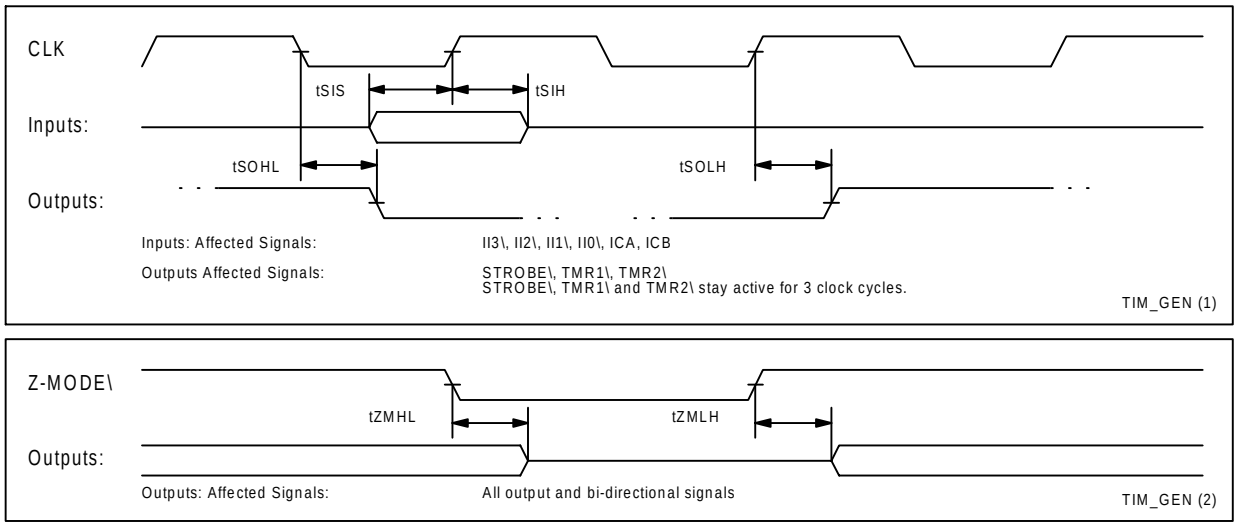


Figure 7.2: Synchronous Inputs, Synchronous Outputs, High-Impedance Mode

Symbol	Parameter	min	typ	max	Unit
tSIS	Setup Time for synchronous input signals	3	-	-	ns
tSIH	Hold Time for synchronous input signals [1]	1	-	-	ns
tSOHL	High-to-Low propagation delay for synchronous output signals	-	-	23	ns
tSOLH	Low-to-High propagation delay for synchronous output signals	-	-	25	ns
tZMHL	Z-Mode becoming active until all outputs changed to high impedance	-	-	12	ns
tZMLH	Z-Mode becoming inactive until all outputs have been re-enabled	-	-	22	ns

1 The hold time applies only if JTAG boundary scan test is made. Otherwise, it is zero.

Table 7.2: Timing Symbols for Synchronous Signals, High-Z Pin

Asynchronous Reset:

Proper operation is guaranteed, if both MVBC logic and the JTAG test facility are reset (TRST\). If application environment supports no JTAG signals, then TRST\ must also be connected to RESET\.

Symbol	Parameter	min	typ	max	Unit
tRESL	Reset signal duration	125	-	-	ns
tRESS	Reset signal setup time [1]	12	-	-	ns
tRESH	Reset signal hold time [1]	0	-	-	ns
tRESP	Reset signal to output signals settling to initial values	-	-	34	ns
tTAED	(see read access timing)				
tCASA	(see read access timing)				

1 Even though RESET\ is an asynchronous reset, it must not be deactivated inside the forbidden zone where the clock signal changes from '0' to '1'. Violations may cause metastability and inadequate operation.

Table 7.3: Timing Symbols for Asynchronous Reset

(Space below has been left blank intentionally)

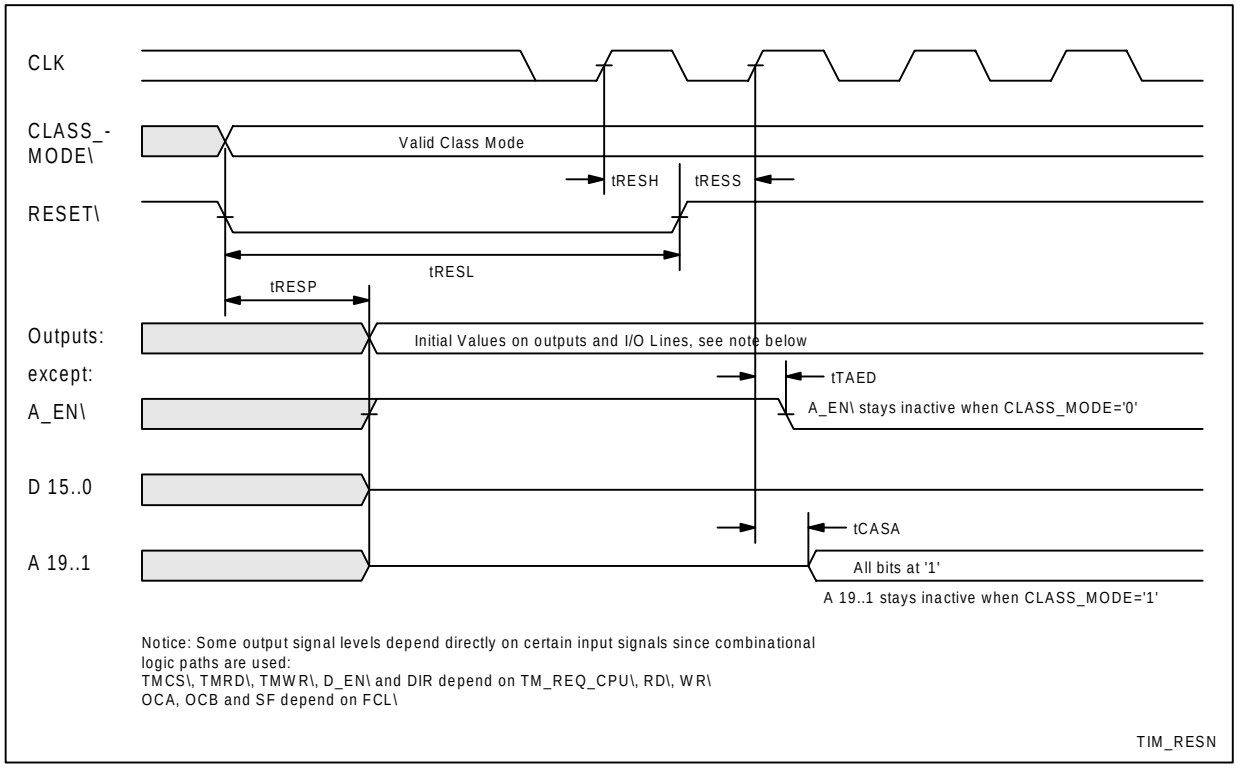


Figure 7.3: Asynchronous Reset

Miscellaneous Signals:

Symbol	Parameter	min	typ	max	Unit
tINHL	Clock to falling INT0\, INT1\	-	-	33	ns
tINLH	Clock to rising INT0\, INT1\	-	-	32	ns
tXIP	Penetration time for external interrupts to INT1\ [1]	-	-	160	ns
tOCHL	Clock to falling OC	-	-	26	ns
tOCLH	Clock to rising OC	-	-	26	ns
tSFHL	Clock to SF falling	-	-	25	ns
tSFLH	Clock to SF rising	-	-	27	ns
tSF	SF overhead at beginning and end of transmitted frames	-	125	-	ns
tFCHL	falling FCL\ to OC becoming active	-	-	20	ns
tFCLH	rising FCL\ to OC becoming inactive	-	-	20	ns

[1] Sum of tSIS, 3 clock cycles, tINLH

Table 7.4: Timing Symbols for Miscellaneous Signals

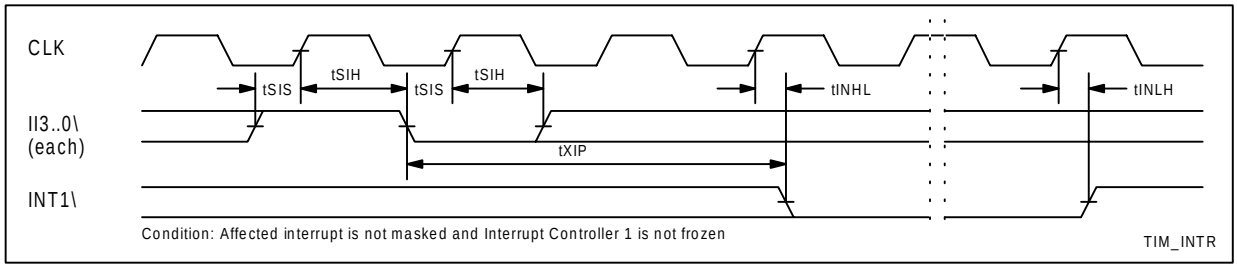


Figure 7.4: Propagation of External Interrupts

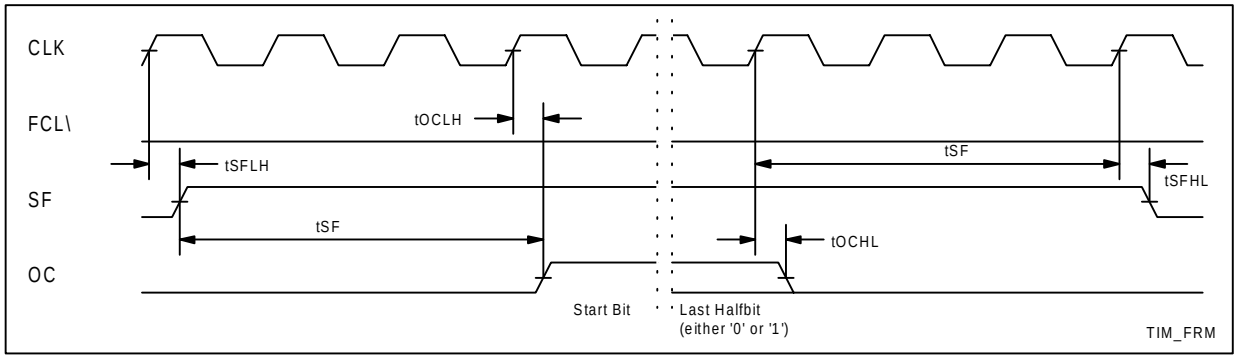


Figure 7.5: SF at Beginning and End of Transmitted Frames

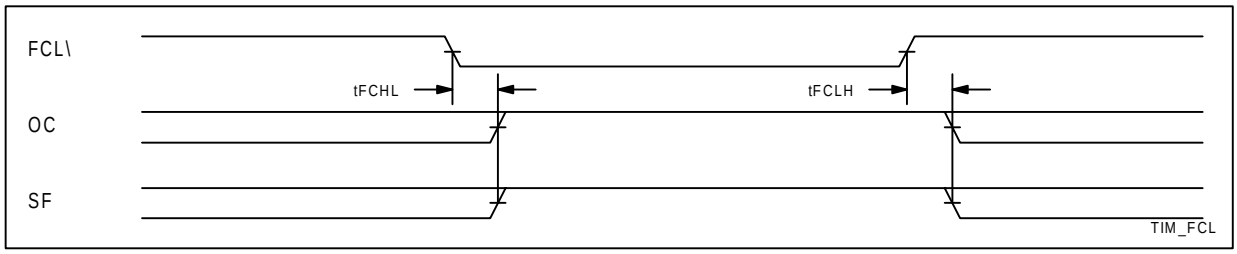


Figure 7.6: Combinational logic path for FCL

7.2 Timings for MVBC accessing Traffic Memory

Symbol	Parameter	min	typ	max	Unit
tTCSF	Rising clock edge to falling TMCS\	-	-	36	ns
tTCSR	Rising clock edge to rising TMCS\	-	-	37	ns
tTCSL	TMCS\ active duration [1]	125	3tCKIN	-	ns
tTCSH	TMCS\ passive duration	40	tCKIN [5]	-	ns
tTRDF	Rising clock edge to falling TMRD\	-	-	33	ns
tTRDR	Rising clock edge to rising TMRD\	-	-	35	ns
tTRDL	TMRD\ active duration [2]	83	2tCKIN [3]	86 [3]	ns
tTRDH	TMRD\ passive duration	39	tCKIN [3]	-	ns
tTWRF	Rising/Falling clock edge to TMWR\	-	-	34	ns
tTWRR	Rising/Falling clock edge to TMWR\	-	-	35	ns
tTWRL	TMWR\ active duration [2]	63	1.5tCKIN [3]	-	ns
tTWRH	TMWR\ passive duration	61	1.5tCKIN [3]	-	ns
tAENF	Rising clock edge to falling A_EN\	-	-	24	ns
tAENR	Rising clock edge to rising A_EN\	-	-	25	ns
tAENH	A_EN\ passive duration	208	5tCKIN	-	ns
tDENF	Rising clock edge to falling D_EN\	-	-	28	ns
tDENR	Rising clock edge to rising D_EN\	-	-	27	ns
tDENH	D_EN\ passive duration	208	-	-	ns
tDIRF	Rising clock edge to falling DIR	-	-	27	ns
tDIRR	Rising clock edge to rising DIR	-	-	28	ns
tDIRL	DIR active duration [2]	83	2tCKIN [3]	86 [3]	ns
tDIRH	DIR passive duration	40	tCKIN [3]	-	ns
tRDYS	TMRDY\ setup time (for recognition at next clock cycle)	2	-	-	ns
tRDYH	TMRDY\ hold time (applies to JTAG boundary scan only)	2	-	-	ns
tRDYI	TMRDY\ inactive time (JTAG active)	4	tCKIN	-	ns
"	TMRDY\ inactive time (no JTAG active = normal operation)	2	tCKIN	-	ns

Continued on next page.

Continued from previous page.

Symbol	Parameter	min	typ	max	Unit
tADDD	Rising clock edge to address bus valid/invalid/changing	-	-	40	ns
tADDA	Rising clock edge to address bus Hi-Z to active	-	-	31	ns
tADDZ	Rising clock edge to address bus active to Hi-Z	-	-	20	ns
tADDV	Address bus valid time during write accesses [2]	-	3tCKIN	-	ns
tDATD	Rising clock edge to data bus valid	-	-	35	ns
tDATA	Rising clock edge to data bus Hi-Z to active	-	-	31	ns
tDATZ	Rising clock edge to data bus active to Hi-Z	-	-	20	ns
tDATV	Data bus valid time during write accesses [2]	48	2tCKIN	-	ns
tDATS	Data bus setup time during read accesses	13	-	-	ns
tDATH	Data bus hold time during read accesses (JTAG Bd. scan only)	1	-	-	ns
tDATE	Data bus RAM chip output enable time after TMRD\ or DIR falling [4]	0	-	-	ns
tDATW	Data bus RAM chip output disable time after TMRD\ or DIR rising [4]	0	-	35	ns
tMACC	MVBC access time cycle [2]	-	3tCKIN	-	ns

[1] 0 waitstates assumed and TMRDY\ permanently active, and only one access is made with TM_REQ_CPU\ active, i.e. reading MF-Slot. Add tCKIN ns for additional wait states

[2] 0 waitstates assumed and TMRDY\ permanently active. Add tCKIN ns for additional wait states

[3] Typical value applies to two consecutive accesses only

[4] tDATE and tDATW represent delays from TMRD\ to output driver response of affected memory chips

[5] Applies only if another access is made immediately thereafter

5Table7.5: Timing Symbols for MVBC Accesses to TM (continued)

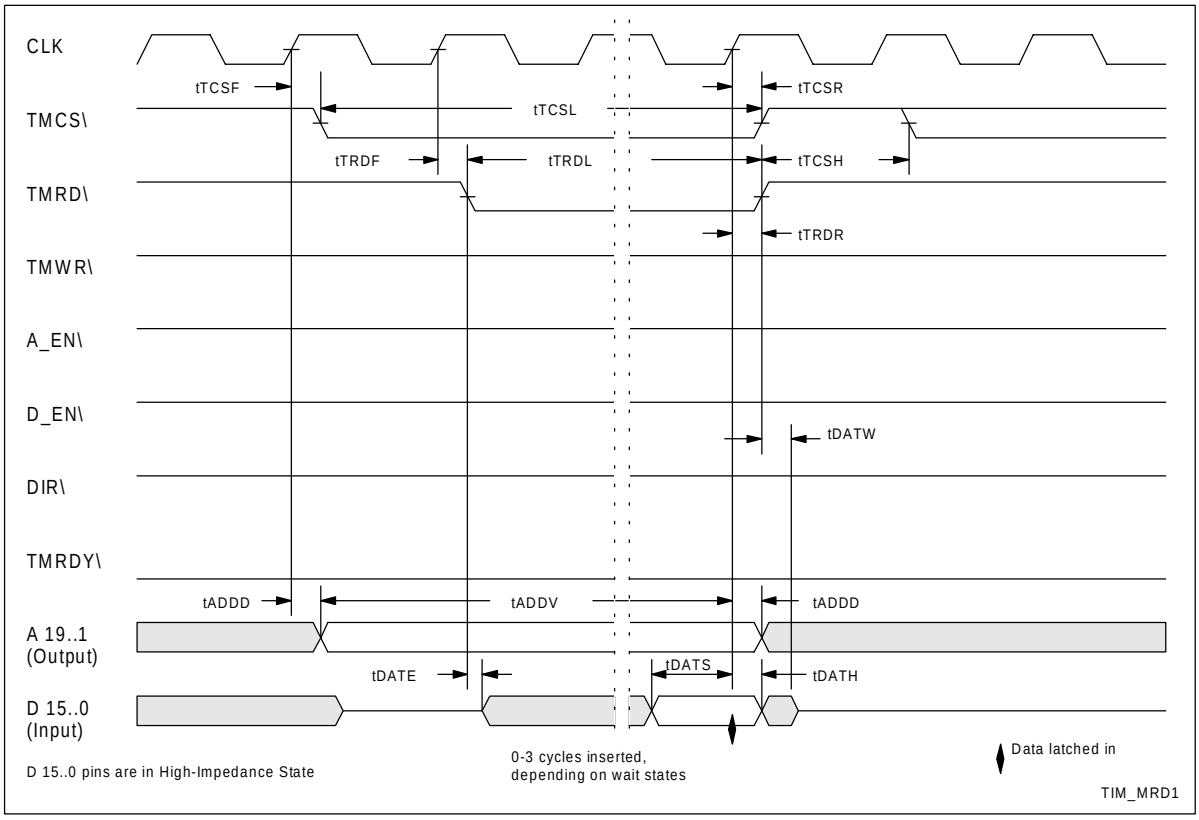


Figure 7.7: Simple Read Access by MVBC to TM

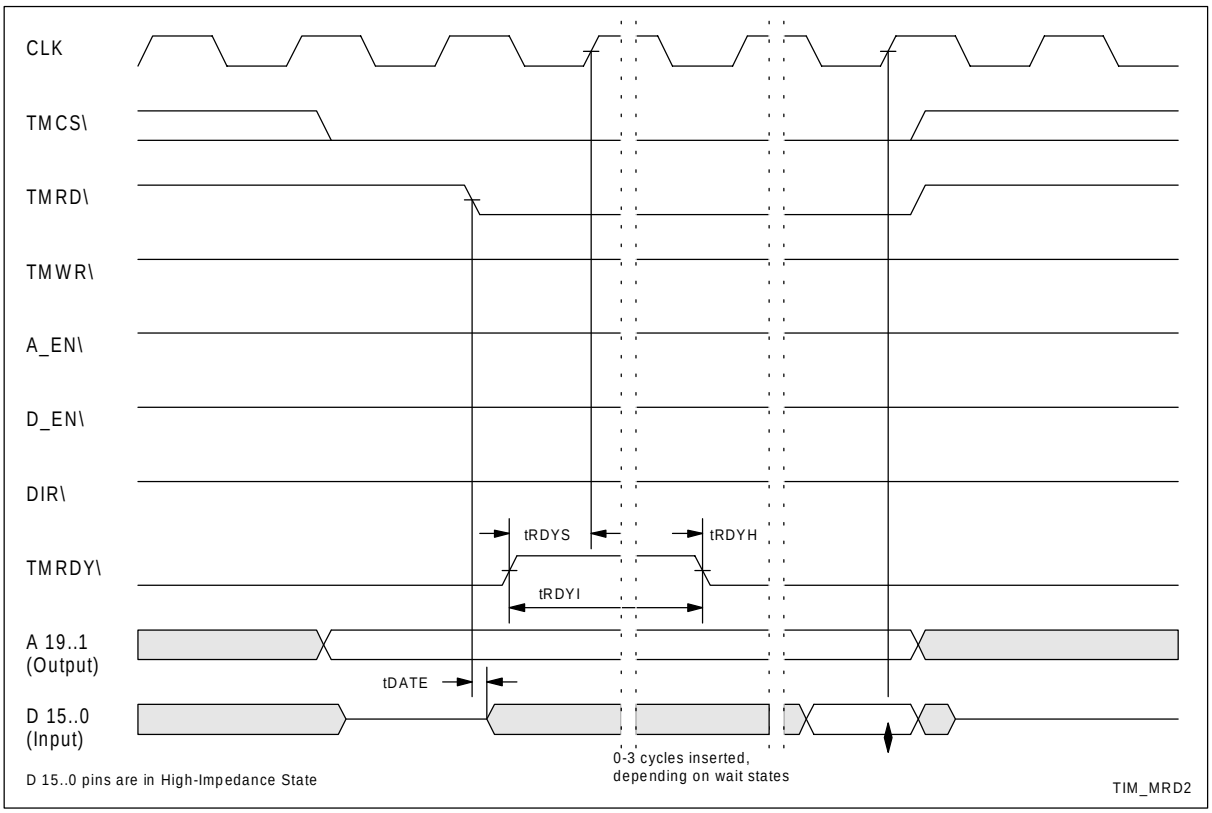


Figure 7.8: Read Accesss by MVBC to TM Delayed by TMRDY

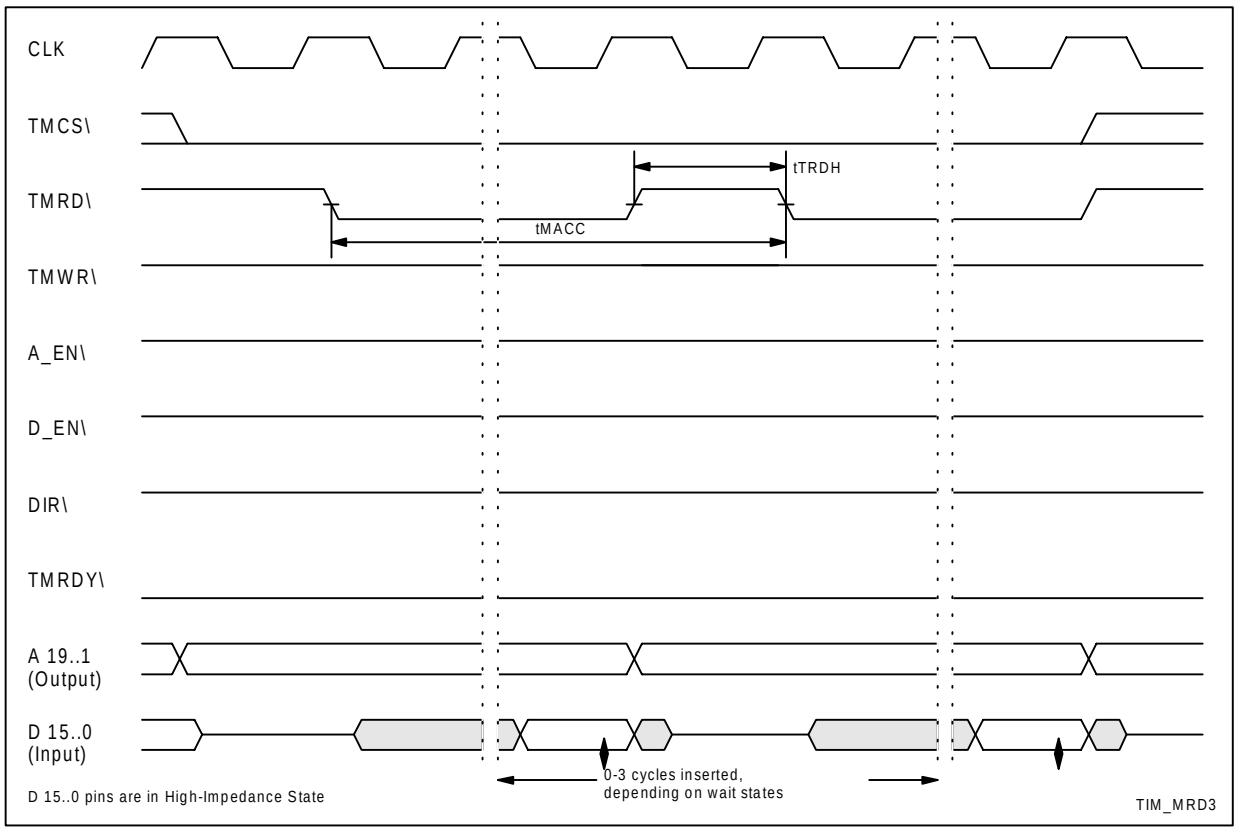


Figure 7.9: Two Consecutive Read Accesses by MVBC to TM

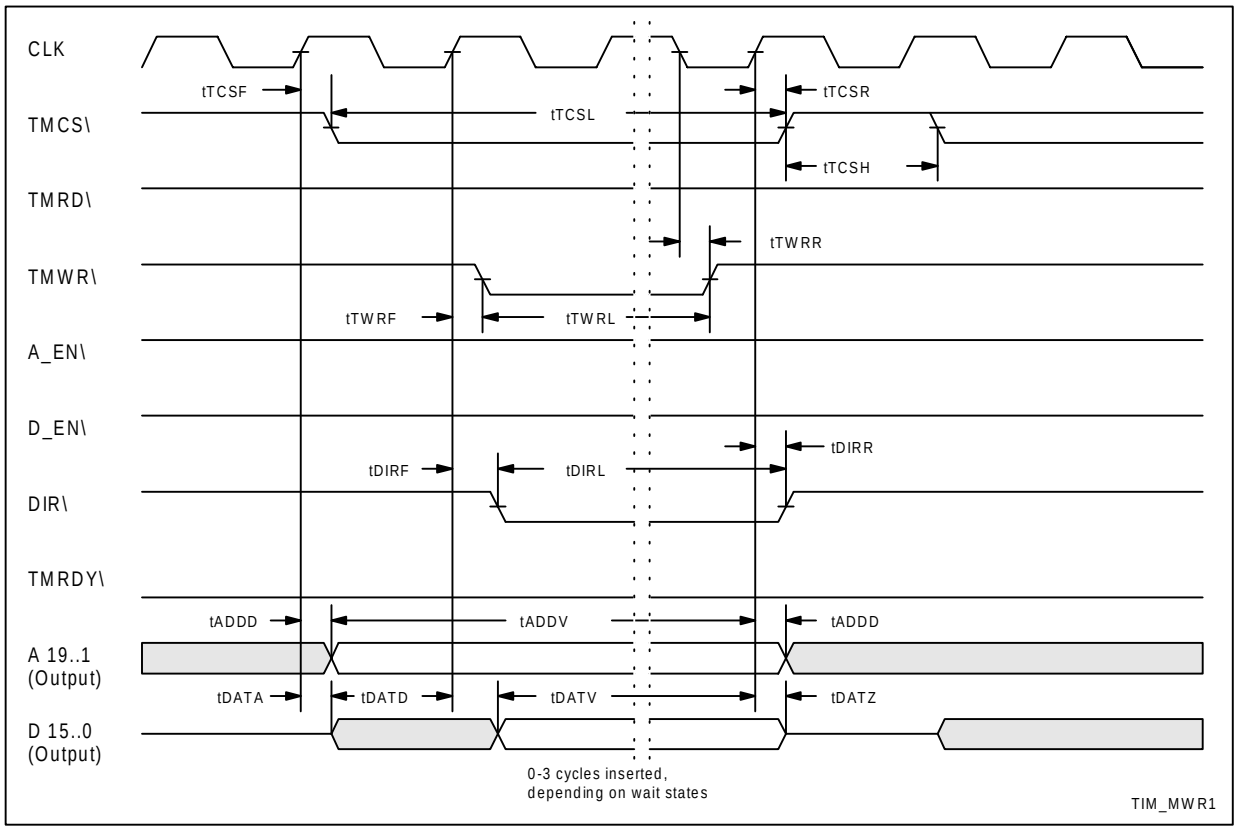


Figure 7.10: Simple Write Access by MVBC to TM

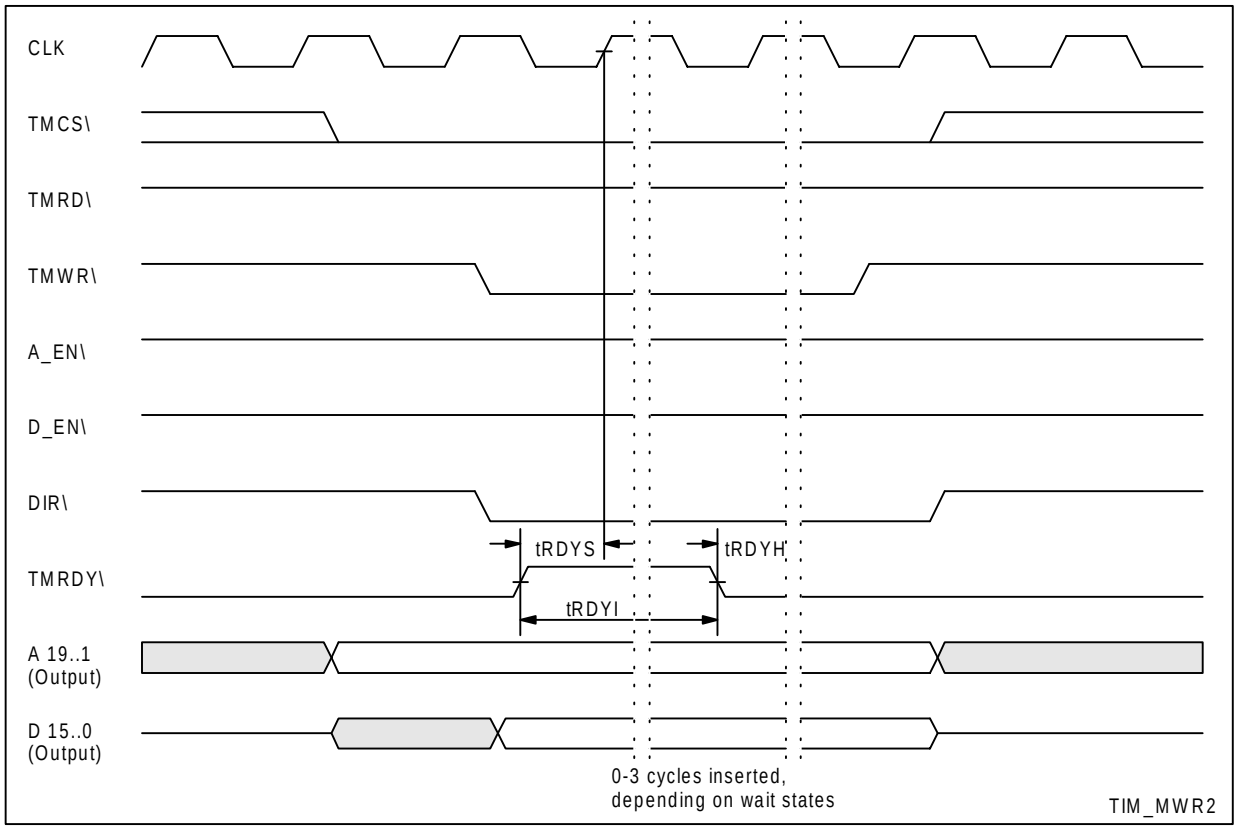


Figure 7.11: Write Accesss by MVBC to TM Delayed by TMRDY

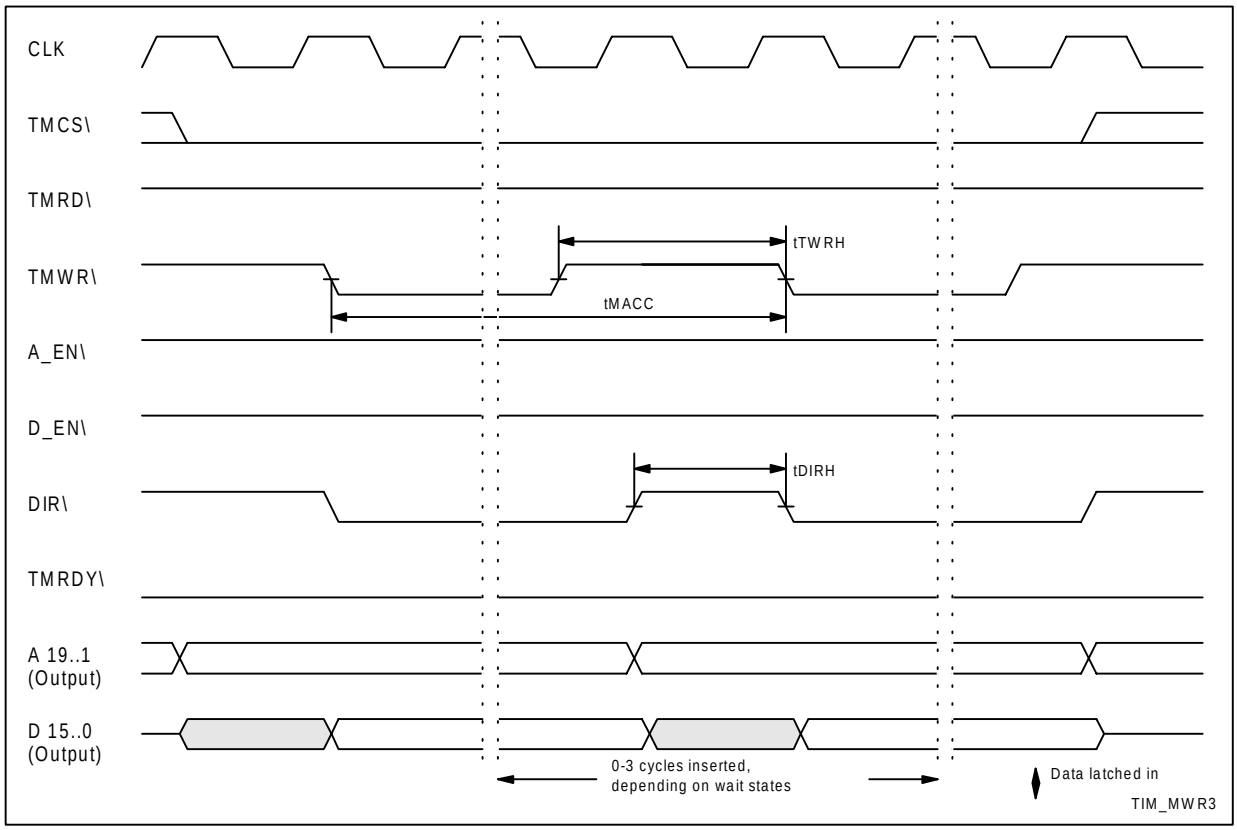


Figure 7.12: Two Consecutive Write Accesses by MVBC to TM

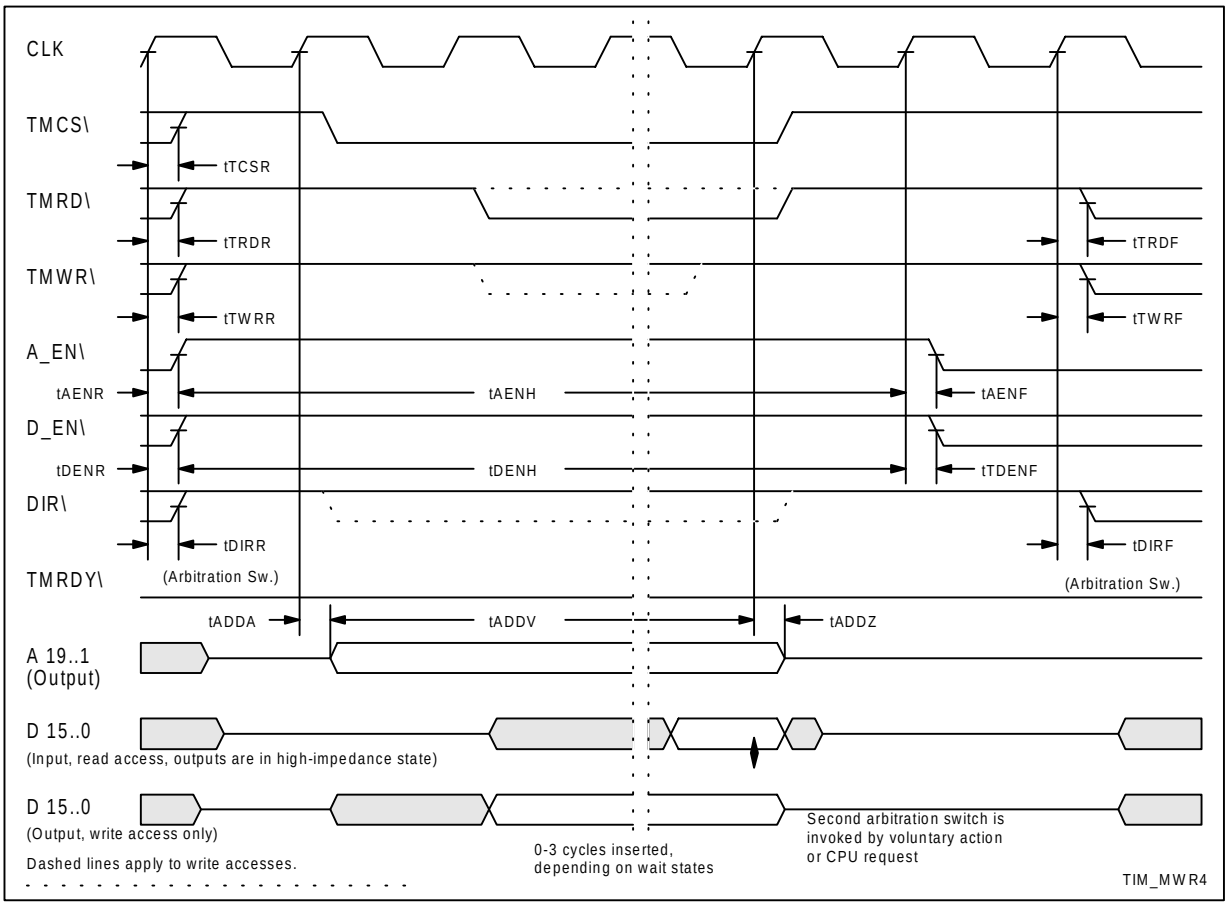


Figure 7.13: Arbitration Switches Before and After Access by MVBC to TM

7.3 Timings for CPU accessing Traffic Memory and MVBC Registers

Timing symbols not found in the following table are listed in the table in section 7.2.

Symbol	Parameter	min	typ	max	Unit
tCSP	TM_REQ_CPU\ passive time	41.7	-	-	ns
tCSL	TM_REQ_CPU\ active time	125	-	-	ns
tCSS	TM_REQ_CPU\ setup time (for recognition at next clock cycle)	5	-	-	ns
tCSH	TM_REQ_CPU\ hold time (Boundary scan only, else 0)	1	-	-	ns
tAP	access (RD\ or WR\) passive time	41.7	-	-	ns
tRDL	RD\ active time	83	-	-	ns
tRDS	RD\ setup time (for recognition at next clock cycle)	5	-	-	ns
tRDH	RD\ hold time (Boundary scan only, else 0)	1	-	-	ns
tRRDH	RD\ active hold time after RDY\ falling [1]	0	-	-	ns
tWRL	WR\ active time	64	-	-	ns
tWRS	WR\ setup time (for recognition at next clock cycle)	5	-	-	ns
tWRH	WR\ hold time (Boundary scan only, else 0)	2	-	-	ns
tRWRH	WR\ active hold time after RDY\ falling [1]	0	-	-	ns
tRDYR	Access[2] beginning to RDY\ rising delay	-	-	27	ns
tRDYH	RDY\ passive time	3	-	65tCKIN	ns
tRDYD	Rising clock edge to RDY\ falling delay	-	-	30	ns
tRDYP	TMRDY\ falling to RDY\ falling propagation delay	36 [3]	-	77	ns
tRDYW	RDY\ passive time if arbitration switch takes place	161	-	-	ns
tACSF	Falling TM_REQ_CPU\ to falling TMCS\ delay	-	-	23	ns
tACSR	Rising TM_REQ_CPU\ to falling TMCS\ delay	-	-	24	ns
tCSCS	TM_REQ_CPU\ to TMCS\ delay in case of arbitration switch [4]	-	-	82	ns
tADEF	Falling TM_REQ_CPU\ to falling D_EN\ delay	-	-	19	ns
tADER	Rising TM_REQ_CPU\ to rising D_EN\ delay	-	-	20	ns
tARDF	Beginning read access[2] to falling TMRD\ delay	-	-	25	ns
tARDR	Ending read access[2] to rising TMRD\ delay	-	-	26	ns
tAWRF	Beginning write access[2] to falling TMWR\ delay	-	-	25	ns
tAWRR	Ending write access[2] to rising TMWR\ delay	-	-	26	ns
tADIF	Beginning write access[2] to falling DIR\ delay	-	-	19	ns
tADIR	Ending write access[2] to rising DIR\ delay	-	-	20	ns
tCSRS	TM_REQ_CPU\ to RD\ Setup Time	0	-	-	ns
tCSWS	TM_REQ_CPU\ to WR\ Setup Time	0	-	-	ns
tRCSS	RD\ to TM_REQ_CPU\ Setup Time	0	-	-	ns
tWCSS	WR\ to TM_REQ_CPU\ Setup Time	0	-	-	ns
tCSRH	TM_REQ_CPU\ to RD\ Hold Time	0	-	-	ns
tCSWH	TM_REQ_CPU\ to WR\ Hold Time	0	-	-	ns
tRCSH	RD\ to TM_REQ_CPU\ Hold Time	0	-	-	ns
tWCSH	WR\ to TM_REQ_CPU\ Hold Time	0	-	-	ns
tADDS	Address bus setup time (accesses to MVBC registers only)	23	-	-	ns
tADAS	Address to access setup time	0	-	-	ns
tADDH	Address bus hold time after A_EN\ becoming inactive	0	-	-	ns
tADDE	Address bus output enable time after A_EN\ falling	0	-	-	ns
tADDI	Address bus idle time (not driven) during arbitration switch	41.7	-	83	ns
tDATI	Data bus idle time (not driven) during arbitration switch	41.7	-	-	ns
tDATX	Ending read access to data bus invalid	-	-	42.7	ns
tRAC	Register read and write access Time	**	-	83	ns
tTACC	CPU to TM access time cycle	125	-	-	ns
tRACC	CPU to MVBC Int. Register access time cycle	167	-	-	ns

[1] The read and write accesses must continue until the rising clock edge has been encountered. Reason: The RDY\-signal has been designed in order to achieve a suitable trade-off between performance and portability. If the host CPU ceases memory access immediately (without waiting for rising clock edge), or the CPU does not run synchronously with the MVBC clock so tRRDH/tRWRH are undercut, then a delay device must be used (D-flip flop triggered at rising or falling clock edge).

[2] Access is understood as TM_REQ_CPU\ combined with RD\ or WR\. Read Access and Write Access correlate to RD\ and WR\ only

[3] Minimum delay: Worst-case propagation delay if TMRDY\ arrives just before set-up time of internal RDY-Flip-Flop is reached. The worst-case delay applies if the set-up time has not been met, so an additional delay of 1 clock cycle occurs

[4] Applies if arbitration switch can take place immediately (MVBC makes no access to TM)

Table 7.5: Timing Symbols for CPU Accesses to TM

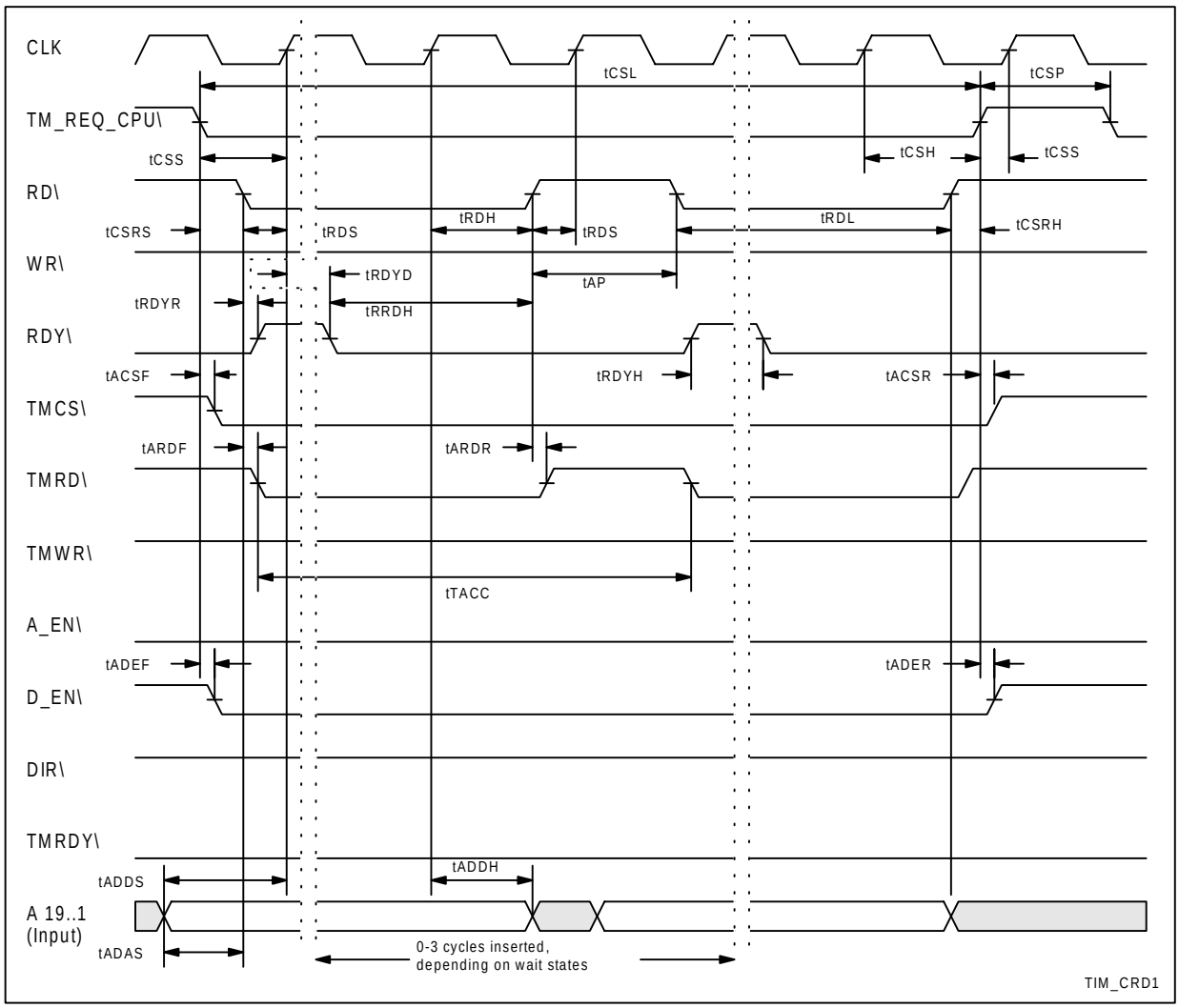


Figure 7.14: Two Read Accesses by CPU to TM

Notes to the timing diagram shown above:

- The address pins A19..1 are in high-impedance state.
- D15..0 is not shown since it affects external RAM chips and not the MVBC

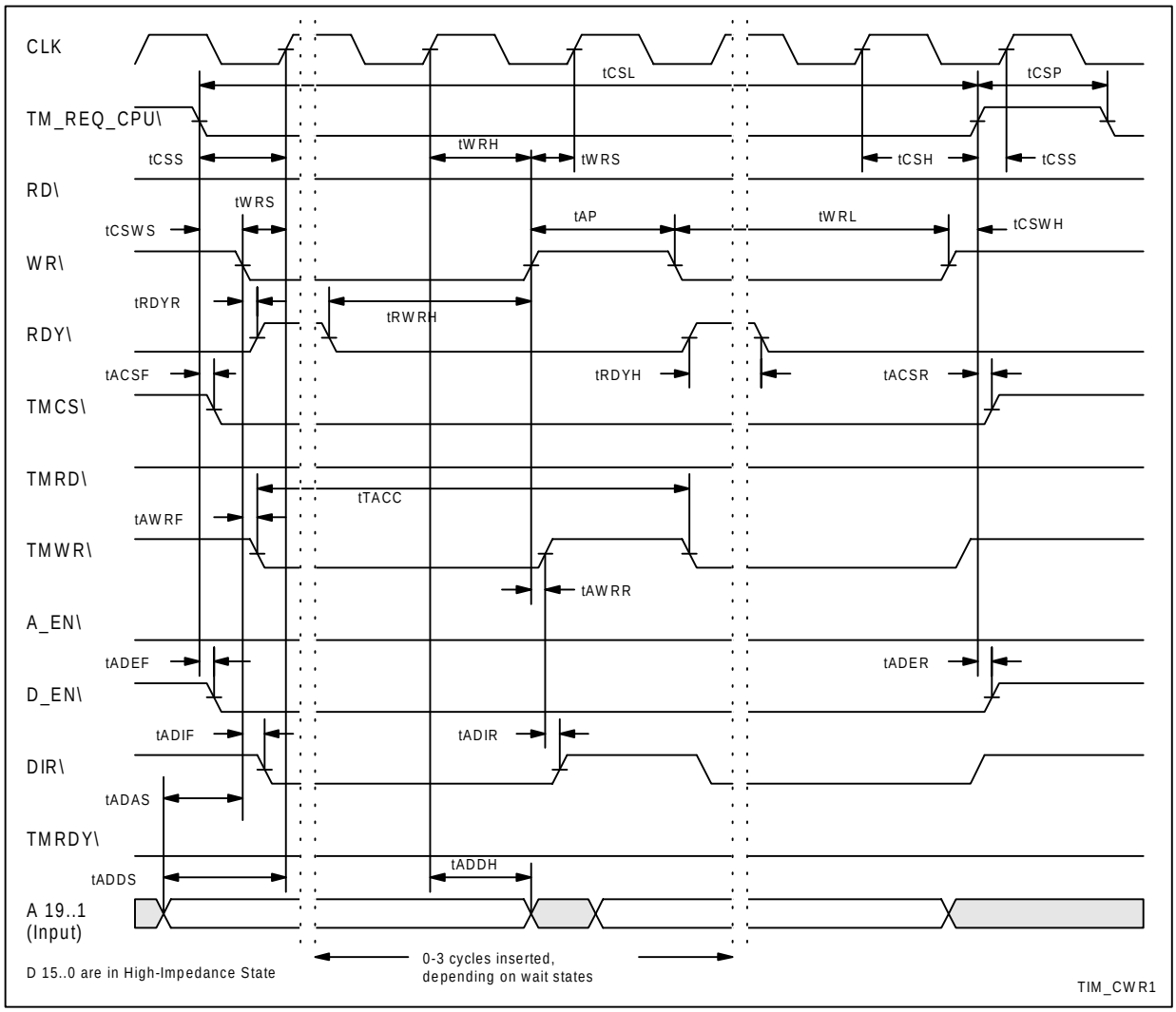


Figure 7.15: Two Write Accesses by CPU to TM

Notes to the timing diagram shown above:

- The address and pins A19..1 and data pins D15..0 are in high-impedance state.
- D15..0 is not shown since it affects external RAM chips and not the MVBC

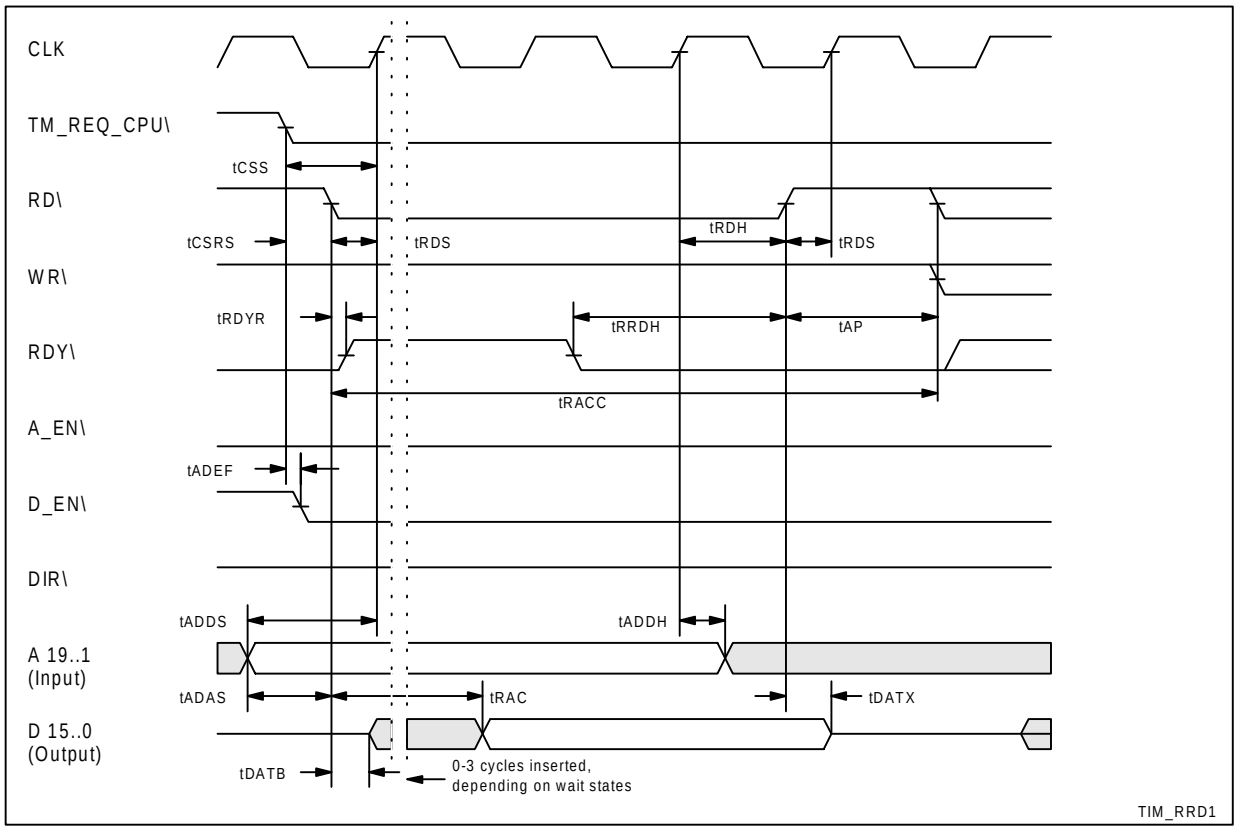


Figure 7.16: One Read Access by CPU to Internal Registers

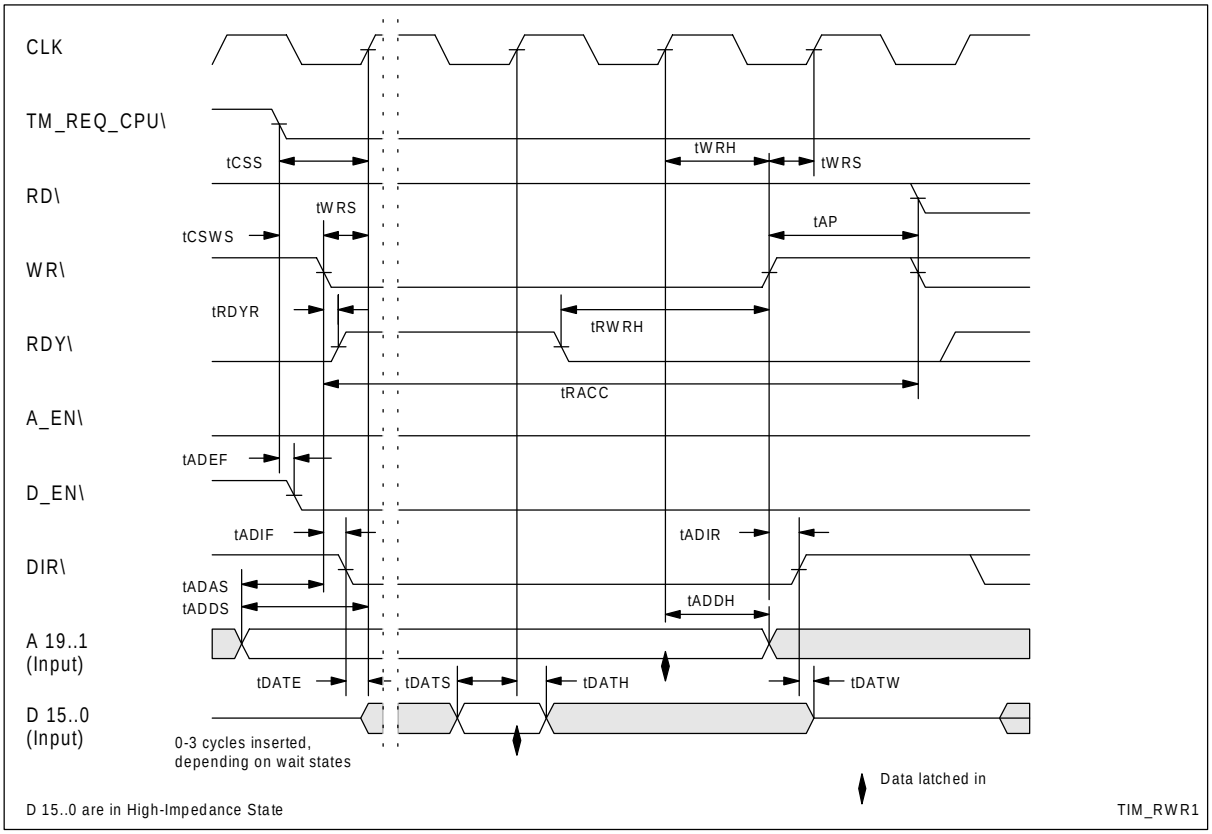


Figure 7.17: One Write Access by CPU to Internal Registers

Notes to the two timing diagrams shown on the previous page:

- The MVBC address pins A19..1 are in high-impedance state.

- Accesses by CPU to Internal Registers
  TMCS\, TMRD\ and TMWR\ remain inactive.
  TMRDY\ is ignored since it affects external components only

- For accesses to Internal Registers, the MVBC assumes at least one wait state in order to synchronize the incoming read instructions and retrieve the data from the requested register.

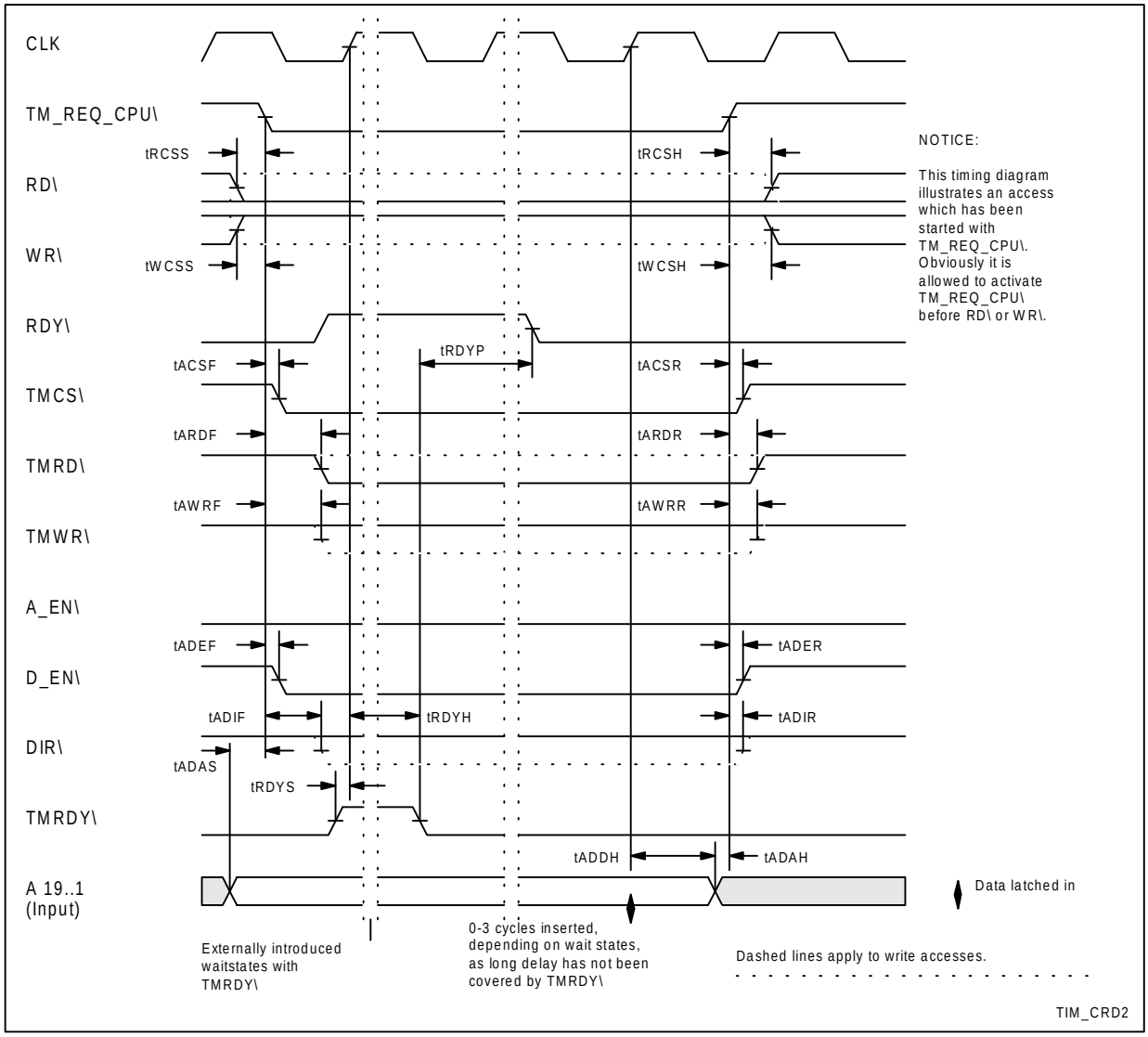


Figure 7.18: TM Access by CPU to TM Delayed by TMRDY

Notes to the timing diagram shown above:

- The address and pins A19..1 and data pins D15..0 are in high-impedance state.
- For obvious reasons, TMRDY\-controlled accesses cannot be made to MVBC Internal Registers.

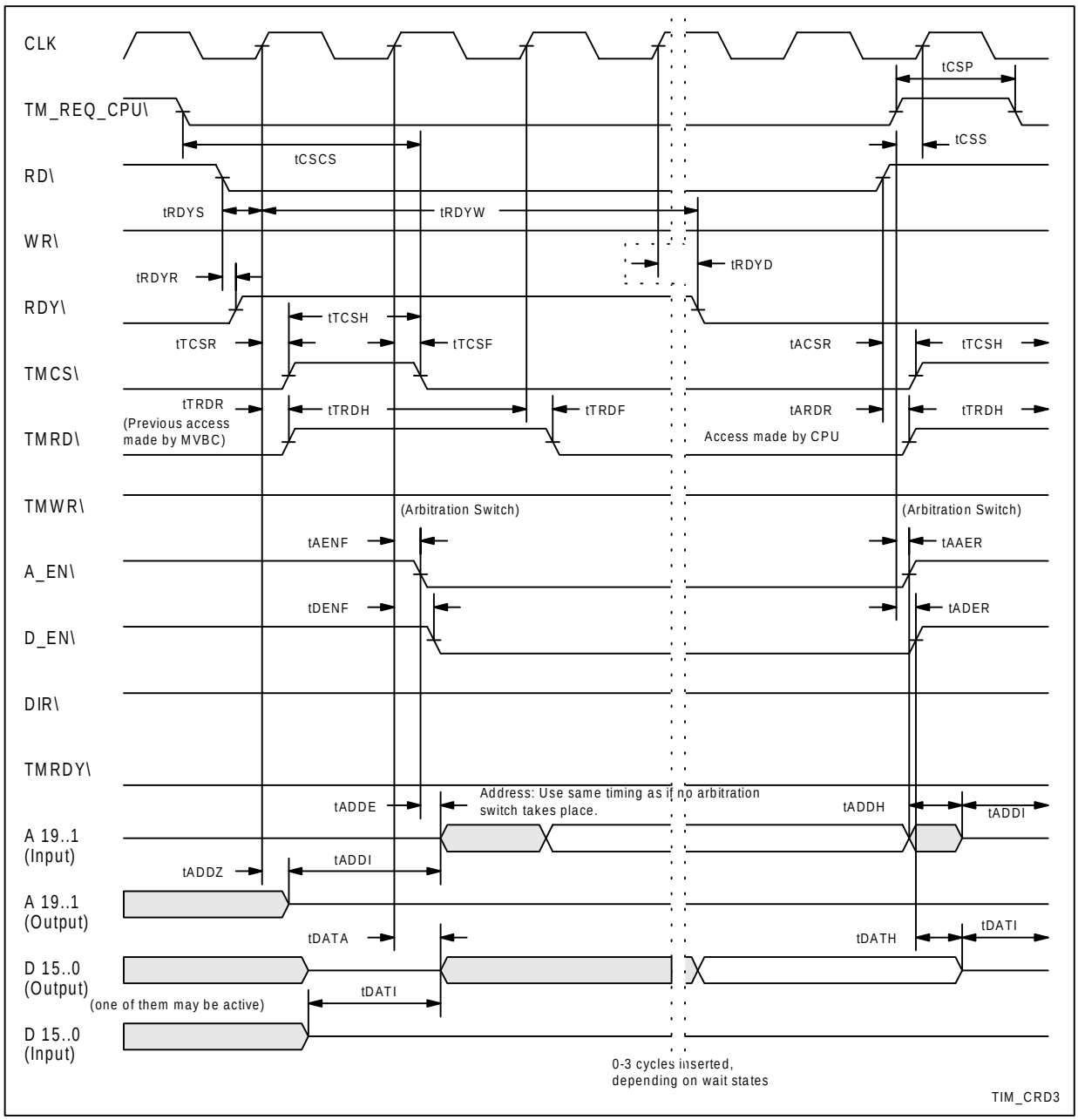


Figure 7.19: TM Read Access by CPU Surrounded by two Arbitration Switches

Notes to the timing diagram shown above:

- Depending on the chosen Arbitration Strategy, the arbitration switch at the end of the CPU access may start when only RD\ or WR\ changes from '0' to '1' while TM_REQ_CPU\ is still active, or when TM_REQ_CPU\ changes from '0' to '1'

- Some timing parameters are listed in the timing table shown in section 7.2.

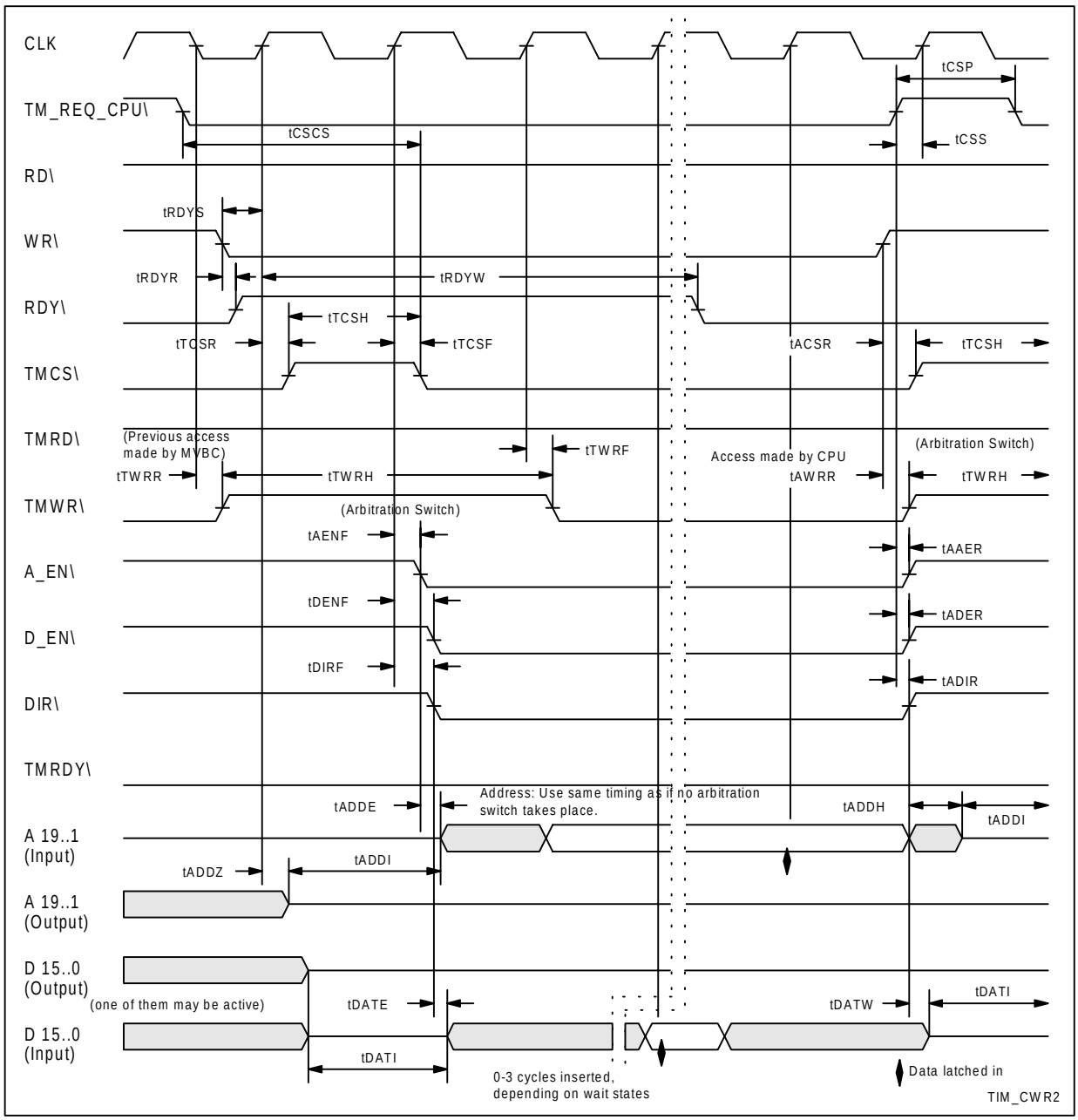


Figure 7.20: TM Write Access by CPU Surrounded by two Arbitration Switches

Notes to the timing diagram shown above:

- Depending on the chosen Arbitration Strategy, the arbitration switch at the end of the CPU access may start when only RD\ or WR\ changes from '0' to '1' while TM_REQ_CPU\ is still active, or when TM_REQ_CPU\ changes from '0' to '1'

MF Dispatcher Latency Data

Symbol	Parameter	min	typ	max	Unit
tSMFM	SMFM/SMFA: Latency btw completed access to MR and SF rising	**	-	-	ns
tSMFT	SMFT: Latency between II3\ and SF rising to transmit MF	**	-	-	ns
tSMFE	SMFE Latency between II3\ and INT0\ (AMFX Interrupt)	**	-	-	ns
tSF	See section 7.1				

Table 7.6: Timing Symbols for MF Dispatcher

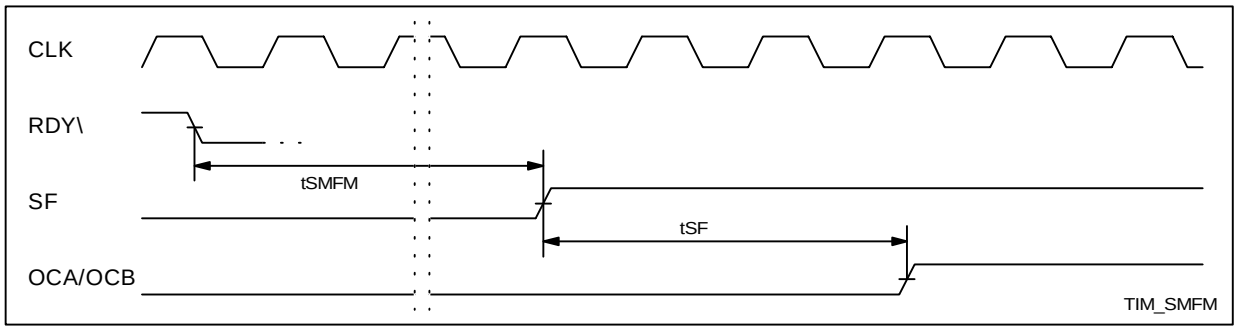


Figure 7.21: SMFM/SMFA MF-Dispatching: Latency between Request and SF

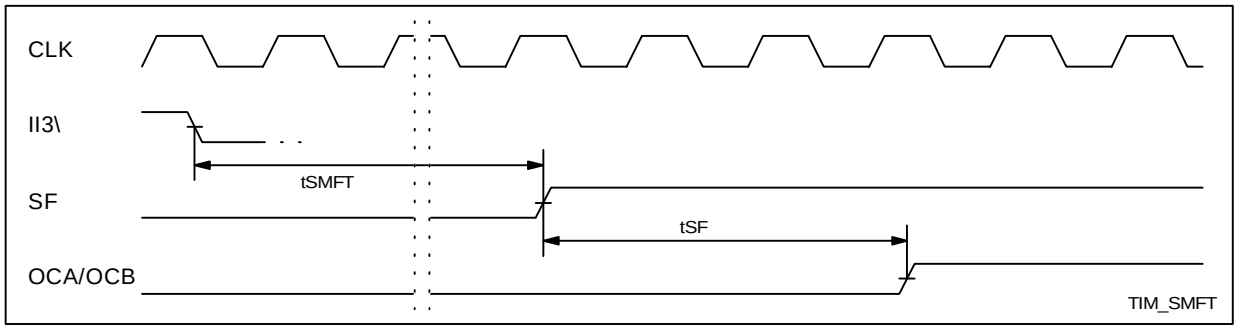


Figure 7.22: SMFT MF-Dispatching: Latency between Ext. Timer Signal and SF

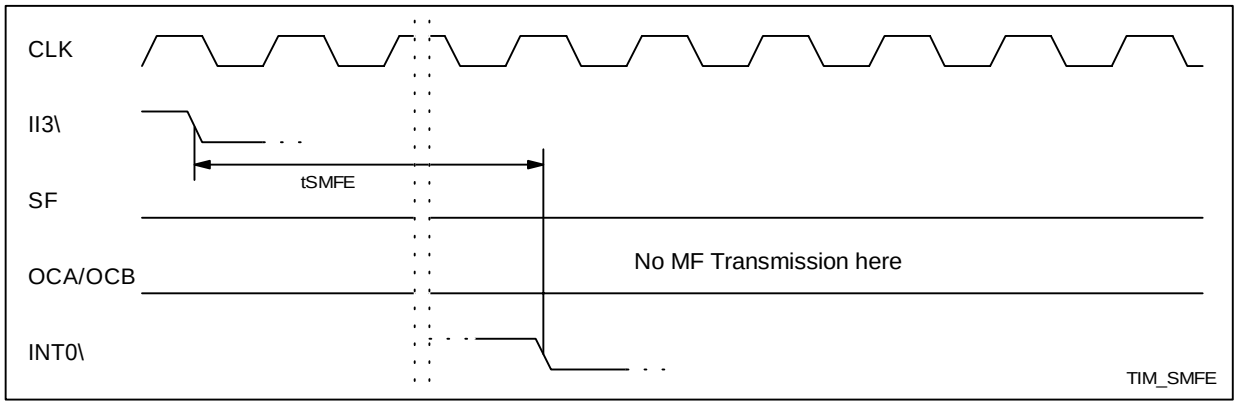


Figure 7.23: SMFE MF-Dispatching: Latency between Ext. Timer Signal and AMFX Interrupt

Notes to the timing diagrams shown above:

- The MVBC starts transmitting the delimiter while the access to Master Frame Slot or MF Table is made. Therefore, the SF signal will not be delayed if the MVBC must wait until clearance is given by the Arbitration Controller.
- RDY\ signal refers to finished write access to the Master Register (MR) which initiates MF dispatching.
- Assumption for II3\-signal: The Interrupt Logic does not process II3\ as a regular external interrupt. The corresponding mask bit masks "External Interrupt 3" out.

Test Mode: Read Access to TXB and RXB

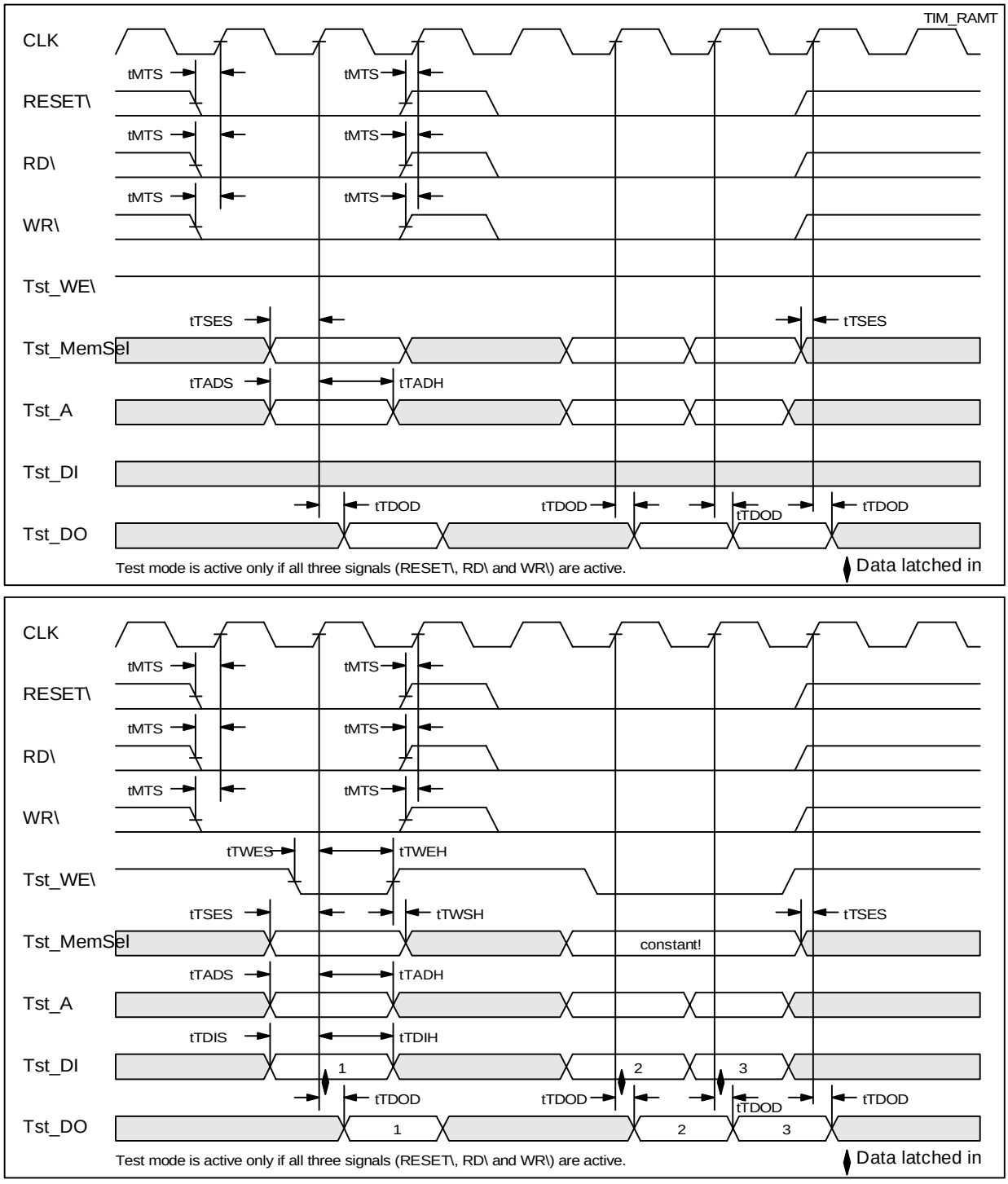


Figure 7.24: TXB/RXB Read Access (top) and Write Access (bottom)

Symbol	Parameter	min	typ	max	Unit
tMTS	Memory test signal to CLK setup time	20	-	-	ns
tSES	Tst_MemSel to CLK setup time	20	-	-	ns
tWES	Tst_WE\ to CLK setup time	20	-	-	ns
tWEH	Tst_WE\ to CLK hold time	0	-	-	ns
tADS	Address lines to CLK setup time	20	-	-	ns
tADH	Address lines to CLK hold time	0	-	-	ns
tDIS	Data input to CLK setup time	20	-	-	ns
tDIH	Data input to CLK hold time	0	-	-	ns
tDOD	CLK to valid data in Tst_DO propagation delay	-	-	25	ns
tWSH	Tst_WE\ rising to Tst_MemSel changing minimum delay	10	-	-	ns

Table 7.7: Timing Symbols Test Modes

Attention: The timing parameters which affect testing the isolated RAM arrays represent safe but empirical values. After actual testing has taken place, then these values will be replaced by more accurate ones.

(Space below has been left blank intentionally)

8 OPERATION IN SYSTEM

8.1 Mounted in system

The MVBCs 100 pin quad flat pack allows easy soldering using commercial reflow equipment. In addition the gullwing pins allow easy visual verification using manual and automated testing equipment.

The following points shall be respected when using the MVBC:

- The JTAG interface pins, if not integrated into a complete JTAG environment (i.e. standalone) should either be made available via an external connector or, in the case of an integrated ICT (In-Circuit Tester), via contact points on the layout which respect the minimum spacing requirements of the test needles.
- If the JTAG interface is not used, then the layout shall be designed in a way that all pins are accessible by the ICT tester.

8.2 Reliability

Source: VLSI Technology Inc: *Failure Rate by Process Technology*, January 1, 1993)

8.2.1 Failure Rate

0.8 µm CMOS: 16 Failures after 2.08 x 108 effective device hours
= 86 FITs: Failures In 1 billion (Tera-) operating hours

Failure Rate:
- Mean Time to Failure (MTTF) and Lambda computation
- MTTF 8.6 x 10^{10} hours,
- Lambda (1/MTTF): 1.16 x 10^{-11} hours^{-1}

Derating: δ (FIT) = $2^{(\delta (t)/10)}$ δ (FIT) = Change in FIT δ(t) = Temperature Change

Notice: Devices without production burn-in have been used for this measurement.

Assumptions:
- Junction Temperature Tj: 55 °C
- Activation Energy Ea: 0.7 eV
- Confidence: 60%

8.2.2 Other Reliability Information

ESD: Mil-Std 883C, Method 3015, Notice 7, Ta=25 °C (Ta = Ambient Temperature)

Latch-Up: Per JEDEC 3.2, with VCC=6.5 V, Ta=80 °C

Hot Electron: DC stress:
- Ambient Temperature: 25 °C
- Drain Voltages: 6.0V, 6.5 V, 7.0V
- Gate Voltage range: 2.0V-3.0V

Gate Oxide Integrity: Ramped voltage step of 0.2 V/100 ms, Ta=25 °C, wafer level test

Electromigration: Ta=200 °C, typically: 300 hours, current density typically at 2-8 MA/cm2 range

8.3 Availability

In case of damages, the MVBC, like any other ASIC, cannot be repaired or reprogrammed. The chip must be replaced.

9 MISCELLANEOUS

9.1 Ordering Information

9.1.1 Adtranz and ABB Companies or Suppliers

ABB Industrie AG
Dept. IFM
P.O. Box
CH-5300 Turgi

Contact: Norbert Gersbach
Phone: +41 56 299 34 88
Fax: +41 56 299 20 06
email: norbert.gersbach@chind.mail.abb.com
Product identification: 3BHC140027R1

9.1.2 Third Parties

Avnet E 2000 GmbH
Elekronische Bauelemente
Stahlgruberring 12
D-81829 München

Phone: +49 89 45 11 001
Fax: +49 89 45 11 0129

Product identification: VY17169-2

Remark: Avnet E 2000 is present in most European countries

(Space below has been left blank intentionally)

9.2 Normal Handling

The device shall be handled according to the Electrostatic standard discharge requirements for CMOS devices.

9.3 Reflow Soldering Conditions:

VLSI (MVBC manufacturer) recommends infrared reflow and vapor-phase reflow. The surface temperature of the package and the temperature profile should be selected according to the table below:

Step	Description	Infrared	Vapor-Phase	Units
1.	Maximum preheating rate	2	2	°C / s
2.	Preheating duration	60	60	s
3.	Maximum heating rate	1-2	1-2	°C / s
4.	Maximum solder duration Maximum solder temperature Maximum time above 200 °C	10 235 30	30 215 40	s °C s

Table 9.1: Soldering Recommendations

Source: VLSI Technology.

9.4 Packing and Unpacking

Antistatic packing material is mandatory. Packing should allow for direct loading of printed circuit board stuffing machine (belts or tubes).

The package is shipped in a vapor-barrier bag which contains desiccants and a humidity indicator card. The bag is vacuum-sealed. This allows storage up to one year under following conditions:

- Temperature Range: 5°C ... 30°C
- Relative humidity: 40% .. 60%

The humidity indicator card proofs the compliance to dry storage. It must be inspected immediately after opening each sealed bag and should not read a value greater than 20%. If it reads more than 20%, the product should be baked at a temperature of 110°C for 22 hours. The MVBC is shipped in trays which can be exposed to high temperatures for baking. The trays are marked accordingly.

Devices must be assembled onto boards within 48 hours after they have been removed from the oven or a sealed bag.

9.5 Other Requirements

There are no additional requirements.

10 APPENDIX A: FUNCTION CODE SUMMARY

F-Code	MF Parameters	Description
0	Logical Address	Process Data, 1 Word
1	Logical Address	Process Data, 2 Words
2	Logical Address	Process Data, 4 Words
3	Logical Address	Process Data, 8 Words
4	Logical Address	Process Data, 16 Words
5	n/a	(reserved) [1]
6	n/a	(reserved) [1]
7	n/a	(reserved) [1]
8	Device Address	Mastership Offer Poll
9	Parameters [2]	Start/Continue Event Polling Round
10	n/a	(reserved) [1]
11	n/a	(reserved) [1]
12	Device Address [3]	Message Transfer
13	Device Group Address	Group Event Polling
14	Device Address	Individual Event Polling
15	Device Address	Device Status Poll

1 The MVBC does not reply to these F-Codes.

2 Event Mode (EM) and Event Type (ET), see 3.5.

3 First word in Slave Frame is checked for Communication Mode (CM) and Destination Device Address

Table 10.1: Function Codes

(Space below has been left blank intentionally)

11 APPENDIX B: PORT PROCESSING OVERVIEW

F-Code	UTQ	UTS	DA-FIT	Event Round	Other	Port A	Port B
0-4	0	0	-	-	-	LA Port	-
0-4	0	1	-	-	-	LA Port	TSNK [1]
0-4	1	0	-	-	-	TSRC [1]	LA Port
0-4	1	1	-	-	-	TSRC	TSNK
5-7	-	-	-	-	-	-	-
8	0	0	0	-	-	MOS	-
8	0	0	1	-	-	FC8	MOS
8	0	1	0	-	-	TSNK	-
8	0	1	1	-	-	FC8	TSNK
8	1	0	0	-	-	MOS	-
8	1	0	1	-	-	TSRC	MOS
8	1	1	0	-	-	TSNK	-
8	1	1	1	-	-	TSRC	TSNK
9	0	0	-	not partic.	-	EFS	-
9	0	0	-	participating	ET=0	EF0	EFS
9	0	0	-	participating	ET=1	EF1	EFS
9	0	1	-	not partic.	-	TSNK	-
9	0	1	-	participating	ET=0	EF0	TSNK
9	0	1	-	participating	ET=1	EF1	TSNK
9	1	0	-	-	-	TSRC	EFS
9	1	1	-	-	-	TSRC	TSNK
10-11	-	-	-	-	-	-	-
12	0	0	0	-	see [2]	MSNK	-
12	0	0	1	-	"	MSRC	MSNK
12	0	1	0	-	"	TSNK	-
12	0	1	1	-	"	MSRC	TSNK
12	1	0	-	-	"	TSRC	MSNK
12	1	1	-	-	"	TSRC	TSNK
13,14	0	0	0	-	-	EFS	-
13,14	0	0	-	0	-	EFS	-
13,14	0	0	1	1	ET=0	EF0	EFS
13,14	0	0	1	1	ET=1	EF1	EFS
13,14	0	1	0	-	-	TSNK	-
13,14	0	1	-	0	-	TSNK	-
13,14	0	1	1	1	ET=0	EF0	TSNK
13,14	0	1	1	1	ET=1	EF1	TSNK
13,14	1	0	-	-	-	TSRC	EFS
13,14	1	1	-	-	-	TSRC	TSNK
15	0	0	0	-	MCM=0	-	-
15	0	0	1	-	MCM=0	FC15	-
15	0	1	0	-	MCM=0	TSNK	-
15	0	1	1	-	MCM=0	FC15	TSNK
15	1	-	-	-	MCM=0	TSRC	TSNK
15	0	0	0	-	MCM>0	DA Port	-
15	0	0	1	-	MCM>0	FC15	DA Port
15	0	1	0	-	MCM>0	TSNK	-
15	0	1	1	-	MCM>0	TSRC	DA Port
15	1	0	-	-	MCM>0	TSRC	DA Port
15	1	1	-	-	MCM>0	TSRC	TSNK

1 Test and Message Ports are equivalent

2 Sink port is processed only if a valid Communication Mode (CM) is specified and at least one of the following conditions are met: 1) DA of MVBC matches with destination DA inside the message; 2) CM equals 15 (Broadcasting); 3) Message Broadcalling (MBC-Bit in SCR) is enabled.

Table 11.1: Port Processing Overview

12 APPENDIX C: REQUIRED PCS SETTINGS FOR ALL PORTS

Port ID or Type		F-Code	SRC/SINK	TWCS	$IE_{2..0}$	$CPE_{1..0}$	QA	NUM	FE	VP	WA
Non-Event-Driven Data Transfers:											
	LA Ports	0..4	Both	Valid	Valid [7]	00	0	Valid [2]	Valid	Valid	Valid
	DA Ports	15	Sink	Valid	Valid [7]	00	0	1	0	Valid	Valid
Event-Driven Data Transfers:											
	LA Ports	0..4	Both	Valid	Valid [7]	01/10	0	Valid [2]	Valid	Valid	Valid
	DA Ports	15	Sink	Valid	Valid [7]	01/10	0	1	0	Valid	Valid
MSRC	Queued Message	12	Source	Valid [1]	Valid [7]	See [3]	1	0	0	Valid [1]	Valid
MSRC	Nonqueued Message	12	Source	Valid	Valid [7]	See [3]	0	0	0	Valid	Valid
MSNK	Queued Message	12	Sink	Valid [1]	Valid [7]	0	1	0	0	Valid [1]	Valid
MSNK	Nonqueued Message	12	Sink	Valid	Valid [7]	0	0	0	0	Valid	Valid
FC15	Device Status Source	15	Source	0 [6]	Valid [7]	0 [6]	0	1	0	Valid	0 [6]
	Device Status Sink	See DA Ports									
FC8	Mastership O. Source	8	Source	0 [6]	Valid [7]	0	0	1	0	Valid	0 [6]
MOS	Mastership O. Sink	8	Sink	0 [6]	Valid [7]	0	0	1	0	Valid	0 [6]
EF0, EF1	Event Source	9 [4]	Source	0 [6]	0 [6]	0 [6]	0	1	0	Valid	0 [6]
EFS	Event Sink	9 [4]	Sink	0 [6]	Valid [7]	0 [6]	0	1	0	Valid	0 [6]
TSRC	Test Source	Valid [5]	Source	Valid	Valid [7]	See [3]	Valid	Valid	Valid [8]	Valid	Valid
TSNK	Test Sink	Valid [5]	Sink	Valid	Valid [7]	See [3]	Valid	Valid	Valid [8]	Valid	Valid

1 TWCS and VP: VP shall be valid if TWCS=1. CS will be transferred from/to the selected halfword of word 3 of PCS. If TWCS=0, then VP will not be checked.

2 Numeric Process Data: Use '1', Non-numeric Process Data: Use '0'.

3 Depends if data transfer follows Event Arbitration or not.

4 For EF0, EF1 and EFS, the MVBC does not check for matching F-Codes. However, configuring this port with F-Code 9 is strongly recommended.

5 Test Ports are enabled only if corresponding UTQ/UTS bits are enabled in the SCR.

6 Recommended, but not required

7 $IE_{2..0}$='111' invokes automatic comparison mechanism. Not suitable for supervisory frames.

8 Applies to Process Data only (F-Codes 0-4). Must be zero for all other F-Codes.

Table 12.1: Required PCS Settings

13 APPENDIX D: SUMMARY OF INTERNAL REGISTERS

Addresses: "y" stands for "3" if MCM=0; "7" if MCM=1; or "F" if MCM = 2, 3 or 4.

Status Control Register (SCR): Address 0yF80H, init. value = 0x0700

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	IM	QUIET	MBC	-	$TMO_{1..0}$		$WS_{1..0}$		$ARB_{1..0}$		UTS	UTQ	MAS	RCEV	$IL_{1..0}$	

Memory Configuration Register (MCR): Address 0yF84H, init. value = 0x0000

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$VERSION_{4..0}$					-				$MO_{1..0}$		$QO_{1..0}$		$MCM_{2..0}$		

Decoder Register (DR): Address 0yF88H, init. value = 0x0008

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	-												LAA	RLD	LS	SLM

Sink-Time Supervision Register (STSR): Address 0yF8CH, init. value = 0x0000

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$SI_{3..0}$				$R_{11..0}$											

Frame Counter (FC): Address 0yF90H, init. value = 0x0000

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$FC_{15..0}$															

Error Counter (EC): Address 0yF94H, init. value = 0x0000

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$EC_{15..0}$															

Master Frame Registers (MFR): Address 0yF98H, init. value = 0x0000
Master Frame Register Duplicate Exception (MFRE): Address 0yF9CH, init. value = 0x0000

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$F\text{-}Code_{3..0}$				$Addr_{11..0}$											

Master Register (MR): Address 0yFA0H, init. value = 0x0000

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$PAR_{1..0}$		$EA_{1..0}$		$EC_{1..0}$		BUSY	CSMF	$SMF_{1..0}$		SMSM	$C_{4..0}$				

Secondary Master Register (MR2): Address 0yFA4H, init. value = 0x0000

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	-								$SMF_{1..0}$		SMSM	$C_{4..0}$				

Dispatch Pointer Register (DPR): Address 0yFA8H, init. value = 0x0000
Secondary Dispatch Pointer Register (DPR2): Address 0yFACH, init. value = 0x0000

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	$DPR_{15..2}$														-	

Interrupt Pending Register (IPR0): Address 0yFB0H, init. value = 0x0000
Interrupt Mask Register (IMR0): Address 0yFB8H, init. value = 0x0000
Interrupt Status Register (ISR0): Address 0yFC0H, init. value = 0x0000

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	EMF	ESF	DMF	DSF	AMFX	MFC	SFC	RTI	BTI	DTI7	DTI7	DTI5	DTI4	DTI3	DTI2	DTI1

Interrupt Pending Register 1 (IPR1): Address 0yFB4H, init. value = 0x0000
Interrupt Mask Register 1 (IMR1): Address 0yFBCH, init. value = 0x0000
Interrupt Status Register 1 (ISR1): Address 0yFC4H, init. value = 0x0000

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	TI2	XI3	XI2	XQE	RQE	XQ1C	XQ0C	RQC	FEV	Not used				TI1	XI1	XI0

Interrupt Vector Register (IVR0): Address 0yFC8H, init. value = 0x0000
Interrupt Vector Register 1 (IVR1): Address 0yFCCH, init. value = 0x0000

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	Not used							IAV	Not used				$VEC_{3..0}$			

Device Address Override Register (DAOR): Address 0yFD8H, init. value = $DA_{11..0}$ pins

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	-				$DA_{11..0}$											

Device Address Override Key (DAOK): Address 0yFDCH, init. value = 0x0000

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	-								$K_{7..0}$							

Timer Control Register (TCR): Address 0yFE0H, init. value = 0x0022

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	Not used										RS2	TA2	n. used	XSYN	RS1	TA1

Timer Reload Register 1 (TR1): Address 0yFF0H, init. value = 0x0000
Timer Reload Register 2 (TR2): Address 0yFF4H, init. value = 0x0000

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	TR1, TR2															

Timer Counter Register 1 (TC1): Address 0yFF8H, init. value = 0x0000
Timer Counter Register 2 (TC2): Address 0yFFCH, init. value = 0x0000

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Symbol	TR1, TR2															

14 APPENDIX E: MVBC TM ACCESS SEQUENCE

The MVBC performs a series of subsequent accesses to the Traffic Memory while a telegram is processed. For an MVBC operating as a master, all diagrams apply. For an MVBC operating as a slave, the diagram "Master_Path" and its child diagrams apply. However, no TM access is made to read a Master Frame.

Legend:

Diagram	Description of a series of accesses to Traffic Memory
Big Circles	Read or write access to Traffic Memory
Small Circles	Branching Point (No TM access, see attached notes)
Double Circles	Hierarchical instance: Refer to the corresponding diagram on a later page
Arrows	Transitions from one TM access to the next one
Label: "C: ..."	Branching condition
Label: "A: ..."	Additional actions (i.e. interrupts)

14.1 Top Level Path

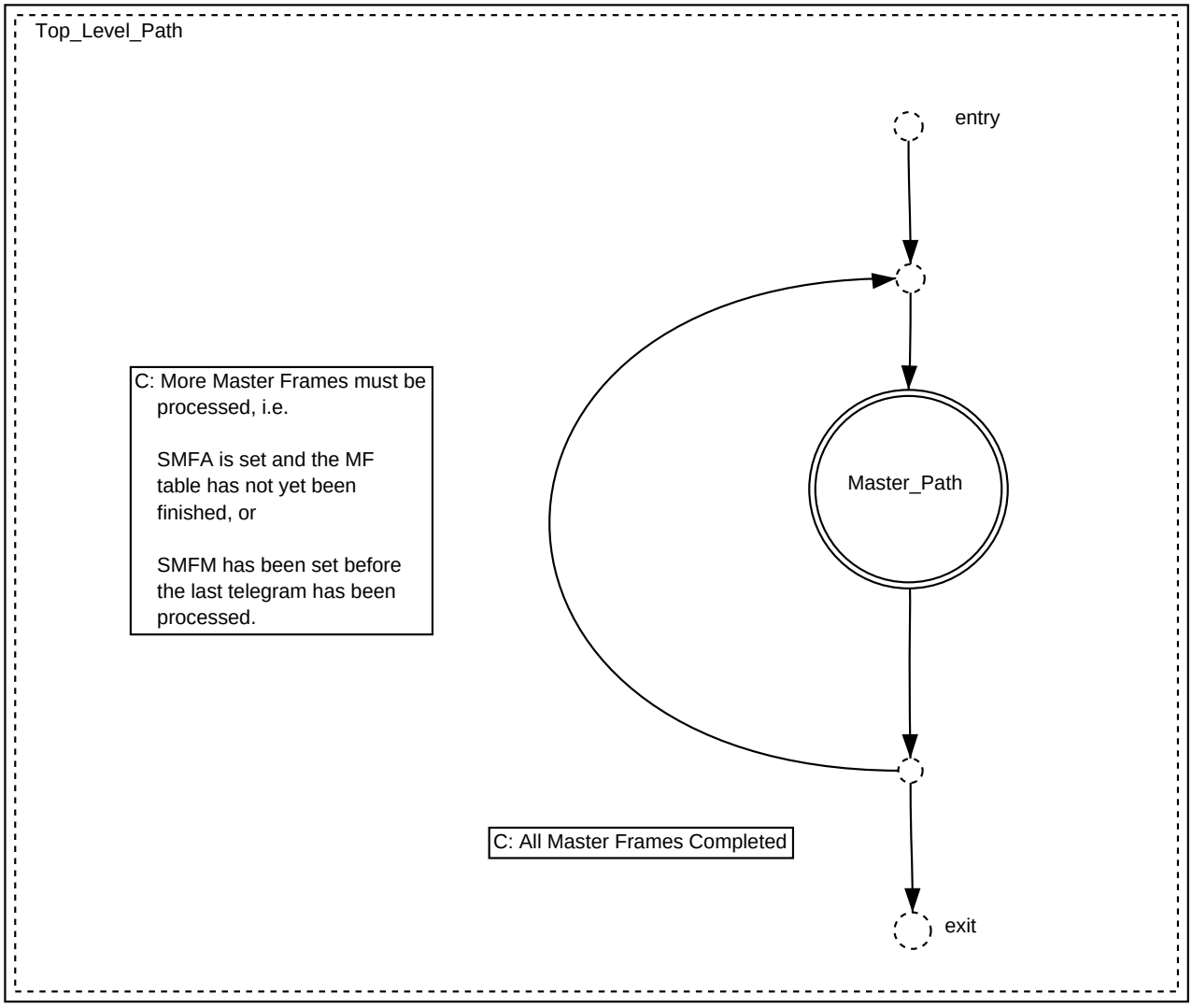


Figure 14.1: MVBC TM Access Sequence: Top Level Diagram for MVB Masters

14.2 Master Path

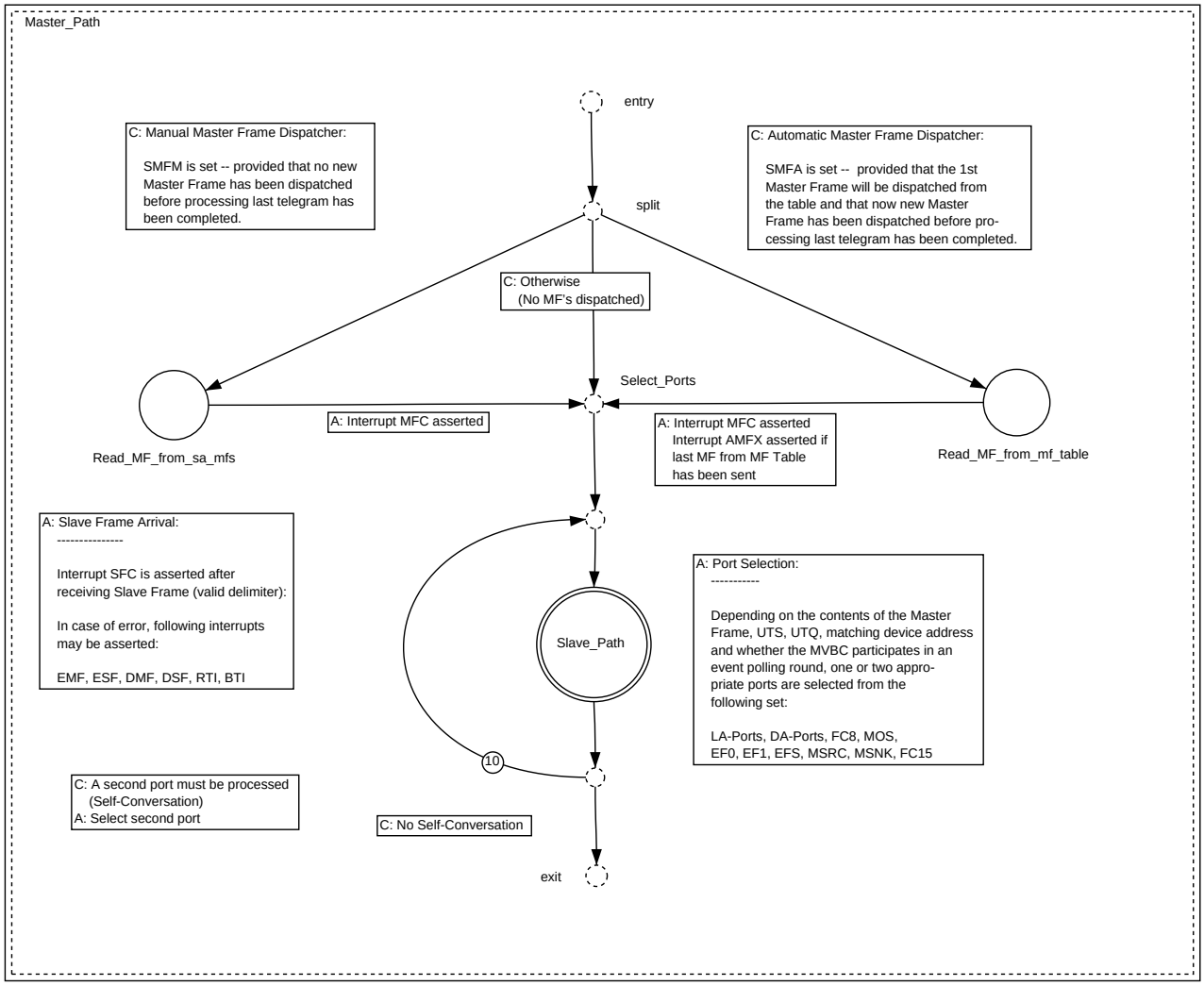


Figure 14.2: MVBC TM Access Sequence: Master Path for MVB Masters and Slaves

The straight downward path is taken by all MVBCs which are acting as slaves.

14.3 Slave Path

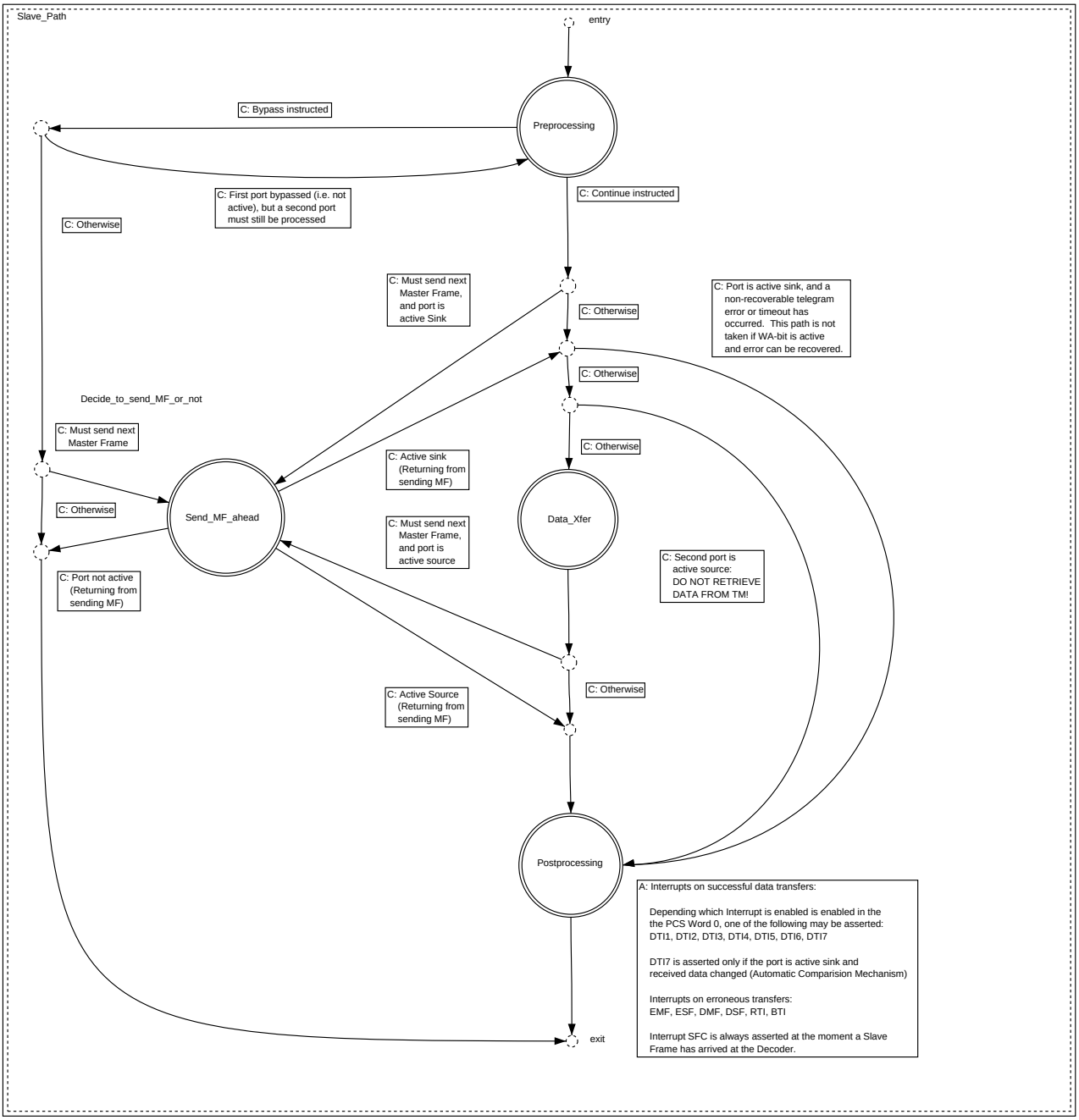


Figure 14.3: MVBC TM Access Sequence: Slave Path: Port Processing

14.4 Port Preprocessing and Postprocessing

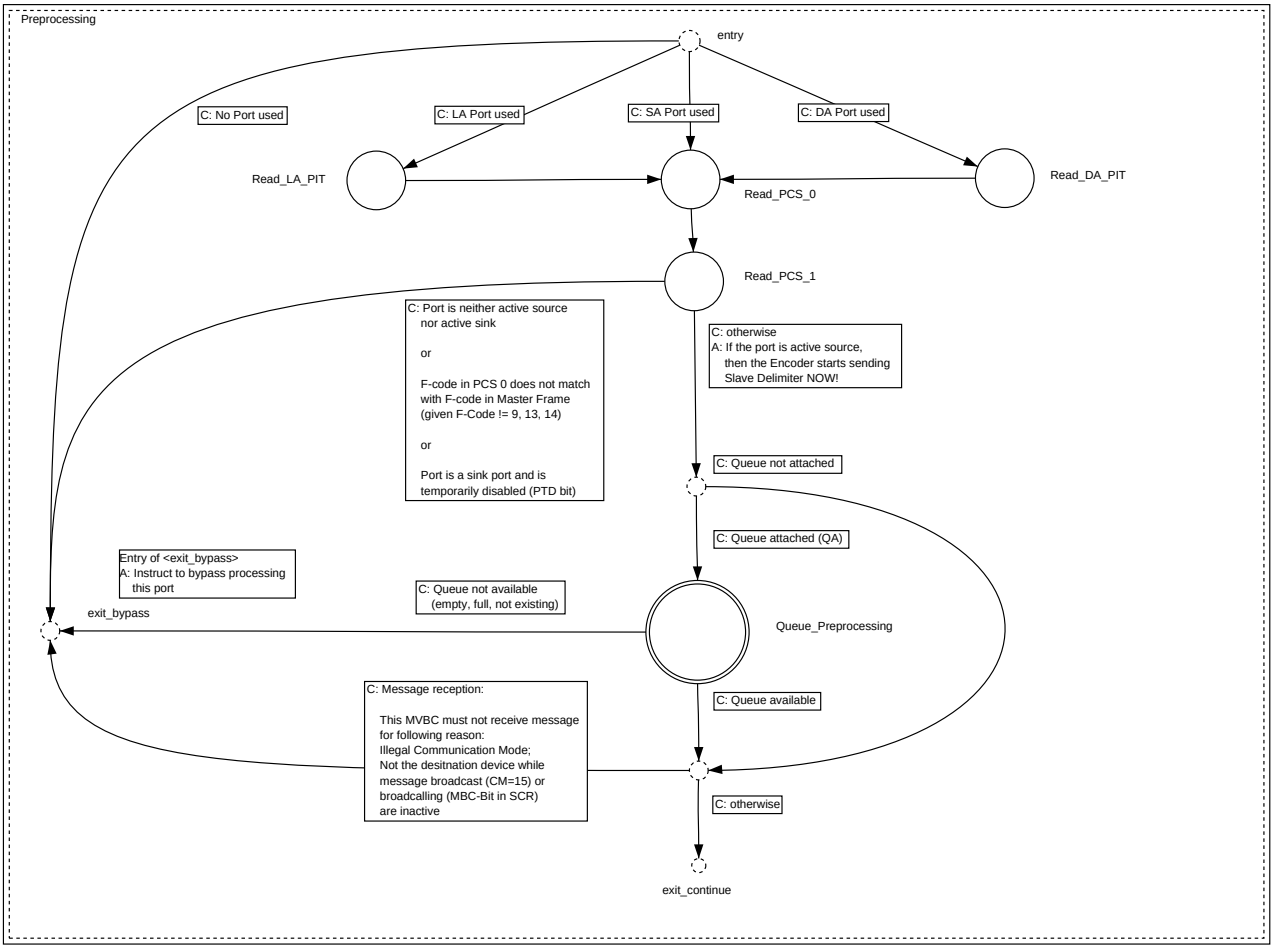


Figure 14.4: Port Preprocessing

(Space below has been left blank intentionally)

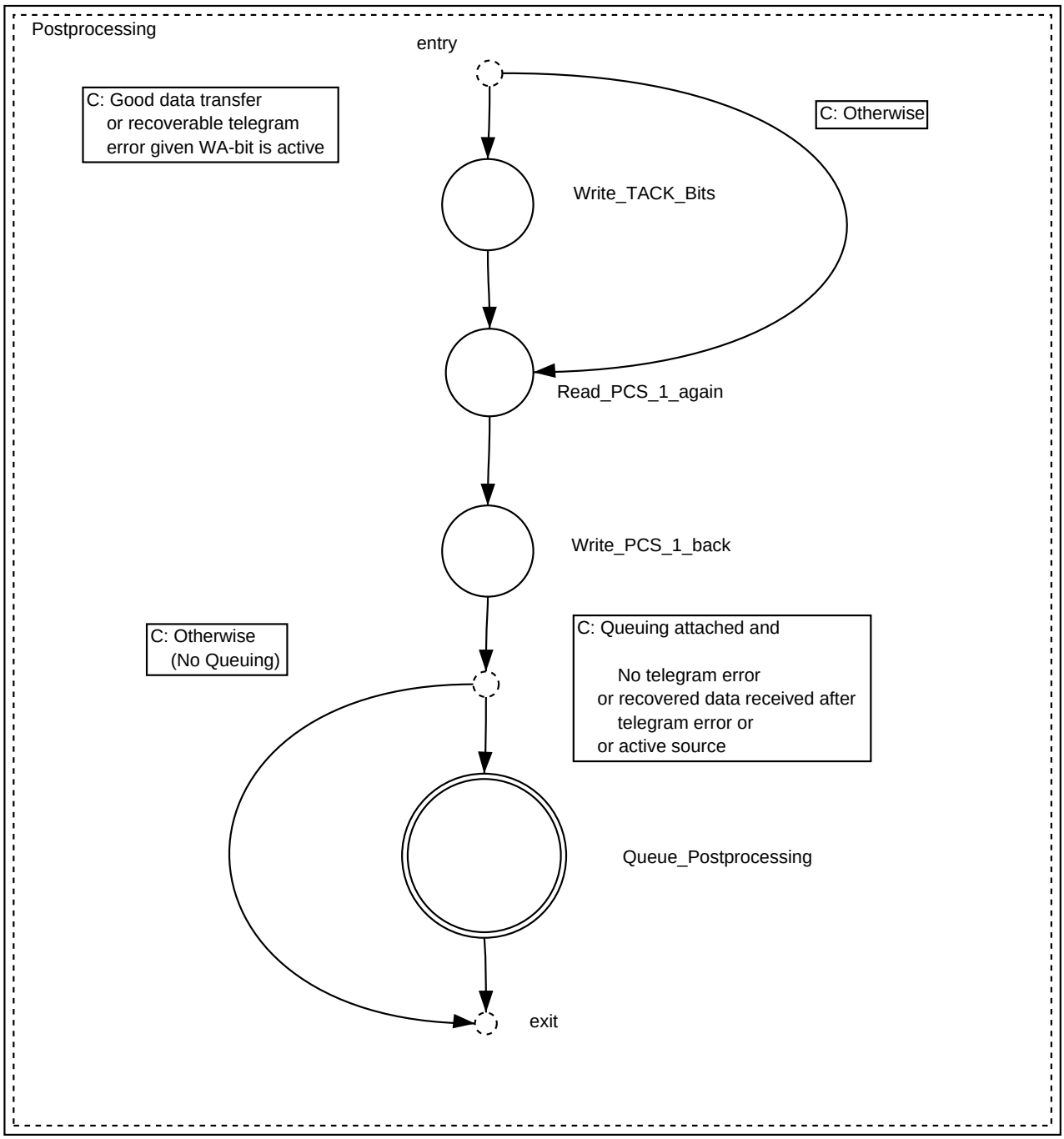


Figure 14.5: Port Preprocessing and Postprocessing

14.5 Queue Preprocessing

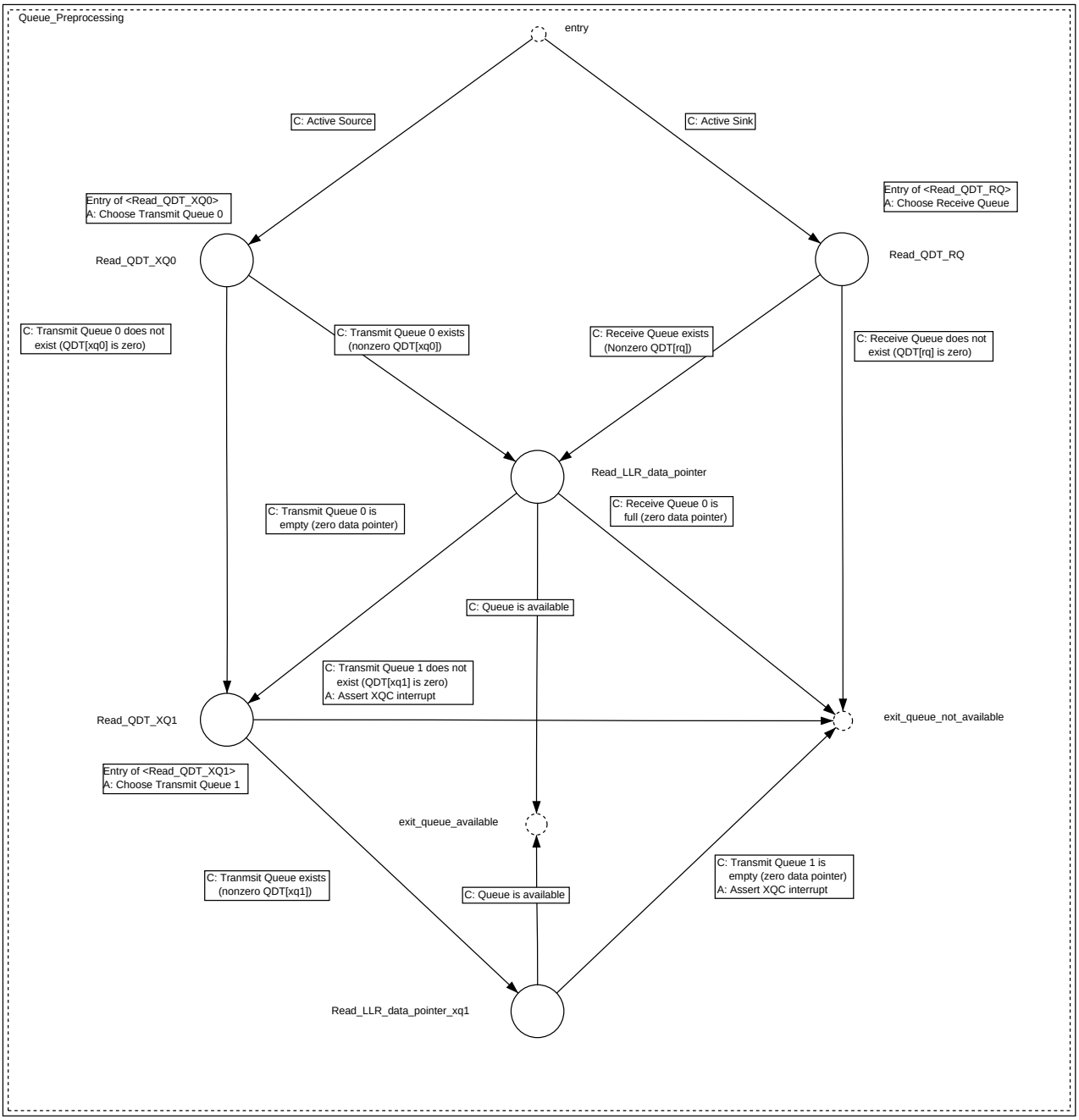


Figure 14.6: Queue Preprocessing

14.6 Queue Postprocessing

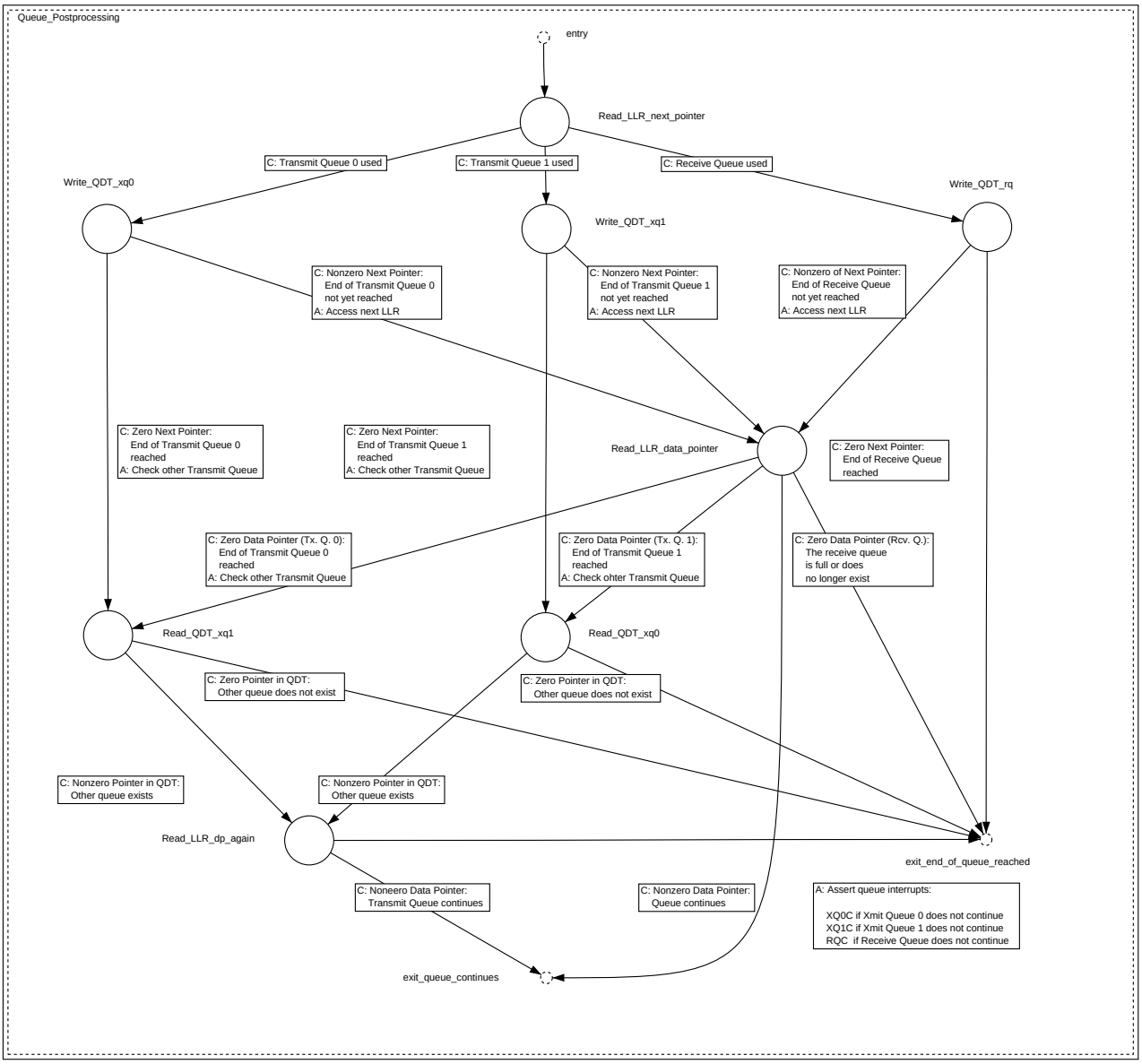


Figure 14.7: Queue Postprocessing

14.7 Data Transfer

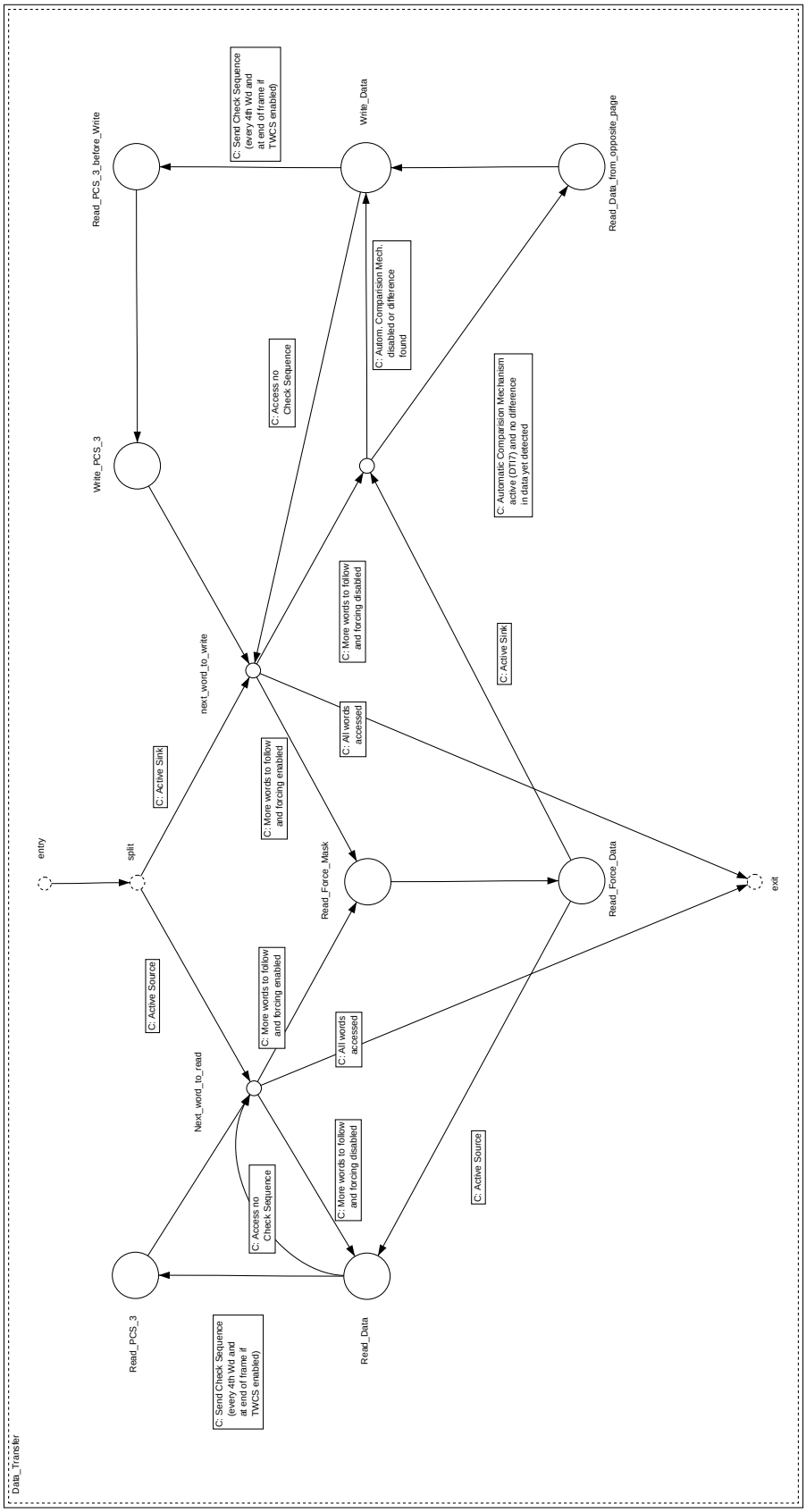


Figure 14.8: MVBC TM Access Sequence: Data Transfer (Data, Force Table, Check Sequences)

14.8 MF Transmission and Sink-Time Supervision

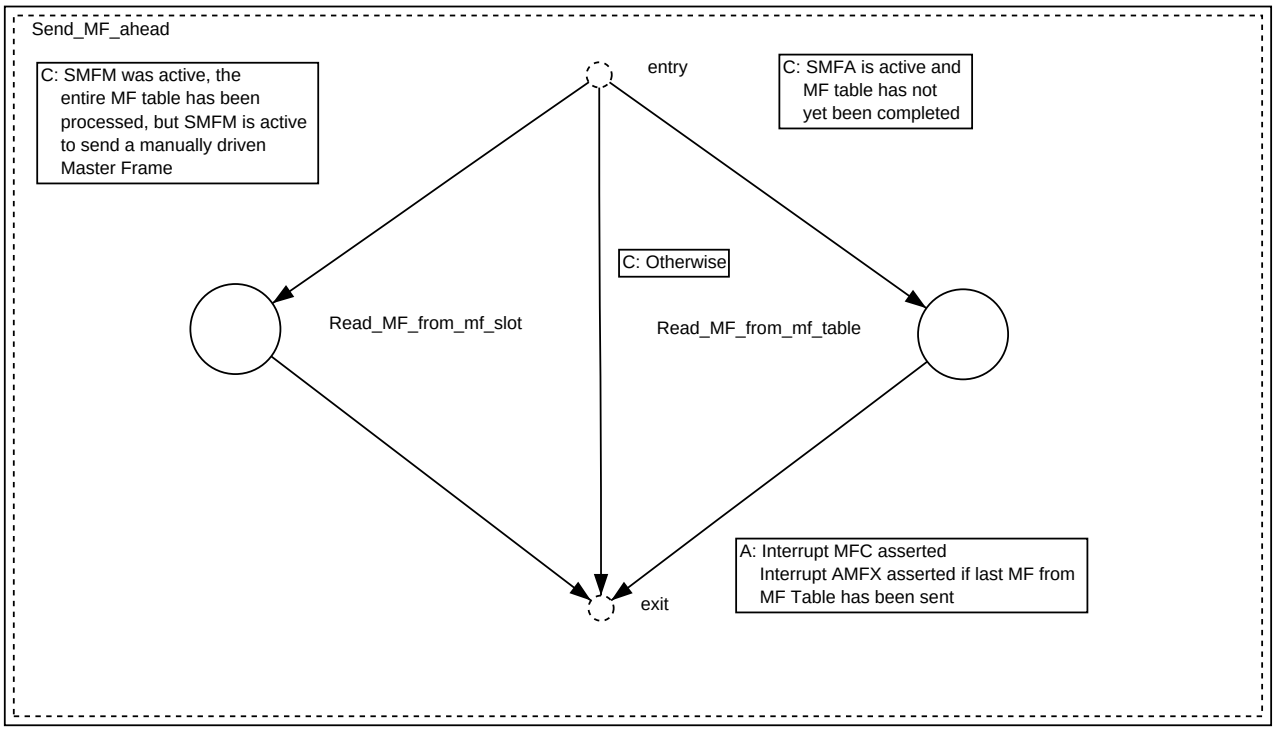


Figure 14.9: MF Transmission

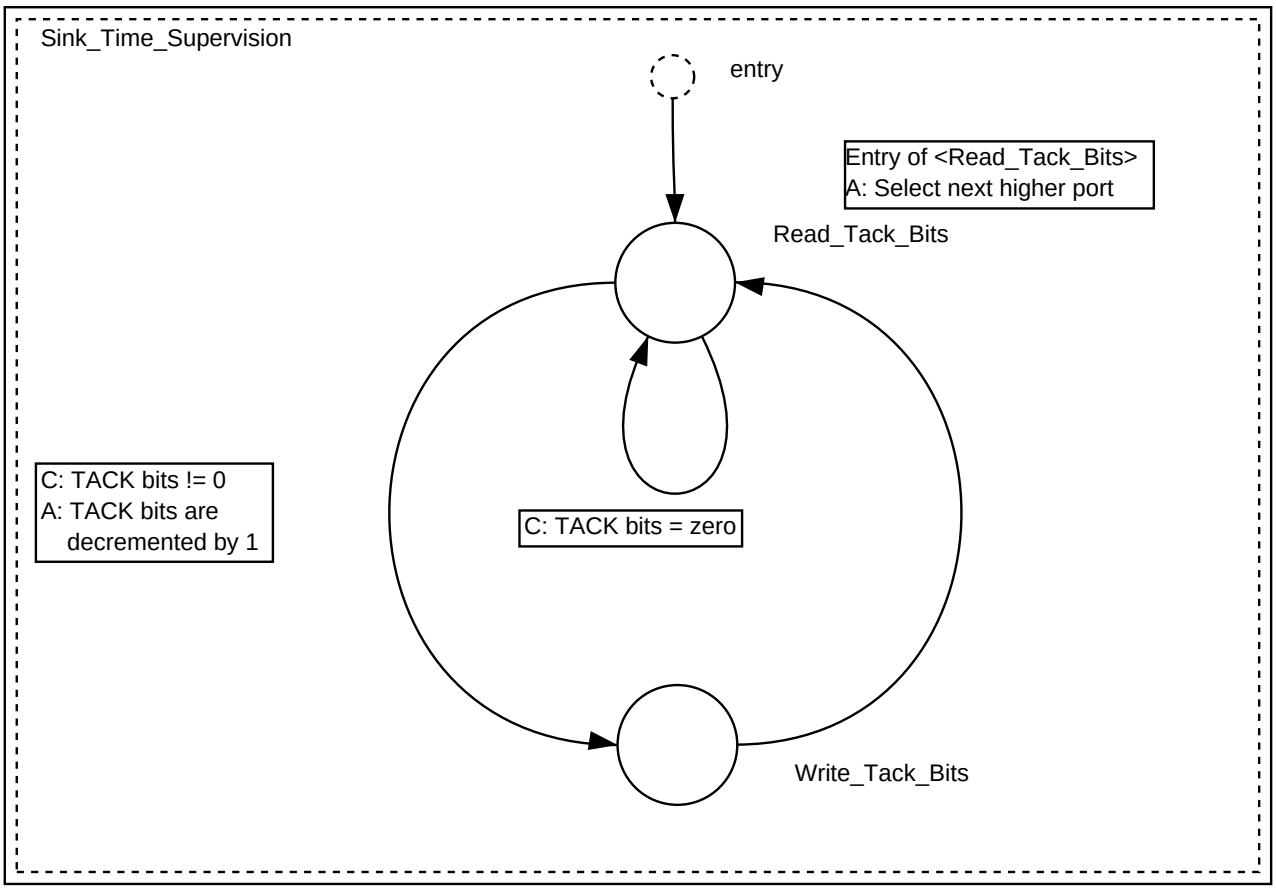


Figure 14.10: Sink-Time Supervision

15 APPENDIX F: PAD CHARACTERISTICS

The I/O signals are described in section 2.5.1. The pad cells have been obtained from the 0.8-Micron Pad Library from VLSI Technology Inc.

Cell name	Description	
PC6C34	CMOS Non-inverting clock pad	CLK
PC6D00	TTL input pad	All inputs except TRST\, TCK\, TMS\ and TDI\
PC6D10	TTL input pad with pull-up resistances 1	TRST\, TCK\, TMS\ and TDI\
PC6S02	4 mA CMOS Tri-state outputs, slew-rate controlled	All outputs
PC6S42	4 mA CMOS Tri-state I/O, slew-rate controlled	Address (A19..1), Data (D15..0)
PC6VD1	Core Vdd inputs	All Core Vdd (+ 5 V)
PC6VS1	Core Vss inputs	All Core Vss (ground)
PC6VD2	Pad Vdd inputs	All Pad Vdd (+ 5 V)
PC6VS2	Pad Vss inputs	All Pad Vss (ground)

1 Resistance: Minimum: 35 KΩ, Maximum: 150 KΩ

Table 15.1: Pad Characteristics

(Space below has been left blank intentionally)

16 APPENDIX G: RESET BEHAVIOR

Asynchronous Reset

The asynchronous reset is activated with RESET\. All flags, registers and internal control states are initialized. The internal Transmit and Receive Buffers will not be reset, but this does not affect the operation.

Test Reset

The reset is activated with TRST\. The JTAG boundary scan system (TAP controller, etc) is initialized. Normally, TRST\ is activated along with RESET\ to start up the MVBC.

Synchronous Reset

The synchronous reset is activated by selecting Initialization Level (IL) to 0, deactivated by setting IL to a different value. This reset shall be active for at least 5 clock cycles.

- Decoder Register (DR) is reset (Line A selected). Incoming data is not affected except when the reset caused a line switch (Line B -> Line A).
- Telegram Analysis Unit: All Timeout counters, Frame Counter (FC), Error Counter (EC), FEV-interrupt reset
- Device Address: Override state. The Device Address from the DA input pins becomes effective.
- Memory Configuration Register (MCR)
- MCU enters idle state within 5 clock cycles. This applies for both synchronous reset and configuration mode (IL = 0 and 1).
- Status Control Register (SCR), except Initialization Level bits
- Master Register (MR, MR2)
- Universal Timers disabled, Timer Counter Registers (TC1/2) set to 0, Timer Control Register (TCR) reset. TR1 and TR2 will not be reset since counters are already inactive.
- All registers in the Interrupt Logics (IPR0/1, IMR0/1, ISR0/1, IVR0/1) are set to zero. The frozen states are released.

Following parts remain unaffected by the synchronous reset:

- MFR and MFRE are not reset.
- STSR Sink-Time Supervision Register stays unchanged. Sink-Time supervision activity is already ceased since IL is forced to 0. Sink-Time Supervision is running only if IL = 2 or 3.

Encoder: Transmission of frames already underway will be completed properly

- Dispatch Pointer Registers (DPR, DPR2)
- Traffic Memory access mechanisms such as Arbitration Controller and TM Controller in order to maintain access to TM and registers
- Device Address Override Register (DAOR) and Key (DAOK)

17 APPENDIX H: BIBLIOGRAPHY

[1] International Electrotechnical Commission
Train Communication Network TCN IEC 9/4/413/CDV, Part 3: Multifunction Vehicle Bus, 1997

[2] Procontrol 215 Datenblatt BAP 15-2/3,
TN EPH-86/279 Änderung A, January 29th 1988

[3] BAC Bus Arbitration Chip,
ABB Transportation Systems Limited, 3EHL 420 001, February 17th 1992

18 APPENDIX I: SUMMARY OF KNOWN ERRATA

18.1 Signals and Timing CPU / MVBC / Traffic Memory

18.1.1 Address Bus Hold Time

Severity: All nodes in Class 2/3/4 mode

Summary: According to a static timing analysis, the address bus ($A_{19..1}$) must be kept stable during the entire MVBC access to any internal register after RDY\ changed from '1' back to '0' until the next rising clock edge. This issue is critical for write accesses to internal registers (see Figure 7.17: One Write Access by CPU to Internal Registers). This problem should not occur when the CPU operates synchronously with the MVBC or RDY\ leads into an "Asynchronous Ready"-input pin of a processor with internal synchronization using a D-Flip Flop stage. The address may not get properly captured if this RDY\ signal enters a CPU and the CPU quits the memory access after less than 40.67ns.

Workaround: 2 choices:

1. Use registers with tri-state outputs rather than uni-directional buffers (see also Figure 4.1: MVBC with 16-bit Traffic Memory) to delay the address by 1 clock cycle.
2. Extend the active level duration of RDY\ by 1 clock cycle.

18.1.2 MVBC Data Buffer Release

Severity: All nodes in Class 2/3/4 mode

Summary: Statement from Siemens Plessey: The data buffers between MVBC and µP are not turned off until TM_REQ_CPU\ signal is released from the µP. TM_REQ_CPU\ is the last signal to be active in the transaction between the µP and the MVBC as it is effectively an enable signal to the Traffic Memory. If a write access follows a read access, then the delay will mean data being left driven on both sides, causing a data clash.
Supplementary statement from ABB: The chip select signal may differ among various CPUs. Therefore, some system solutions will require combinational logic, some other solutions not.

Workaround: Check behavior of user-defined chip-select signal and use external logic if needed.

18.1.3 Intercycle Delay

Severity: All nodes in Class 2/3/4 mode

Summary: Statement from Siemens Plessey: Consider an MVBC read access to TM followed by an MVBC write access: There is one MVBC clock between the cycles (41ns). TMRD\ may rise 0-35ns after the MVBC clock rises. 41ns-35ns=6ns for the memory being read to tristate its bus before the next rising MVBC clock. If you assume 0ns, we are left with 6ns for tristating the bus.
Supplementary statement from ABB: It is most unlikely, that, at a given environment, one signal exhibits best and the other worst case delays. This problem does not appear that serious. However, some slow memory systems (like slow DPRAMs) require longer pauses between 2 consecutive accesses.

Workaround: Glue logic (1 PAL) bridging TMRD\, TMWR\ and TMRDY\.

18.1.4 Address Buffers

Severity: All nodes in Class 2/3/4 mode

Summary: Statement from Siemens Plessey: Consider any µP access followed by an MVBC access. 16ns are left for the address buffer to be tri-stated. AC logic does it in 12.2ns.

Workaround: Fast AC logic for address buffers.

18.1.5 Fast DIR-Signal

Severity: All nodes in Class 2/3/4 mode

Summary: According to the timing diagrams (see also Table 7.5: Timing Symbols for CPU Accesses to TM) DIR may become inactive earlier before TMWR\ becomes inactive. This means that the bi-directional data buffers may be turned off before the write access is ended. This problem is compensated by the propagation delay imposed by the bi-directional buffers:

```
TMWR\ rising -> Traffic Memory
DIR   rising -> Buffers to High-"Z" -> Traffic Memory
```

Worst-case delay difference: 5 ns.

Workaround: If necessary, increase load capacitance of DIR with an extra capacitor in order to align delay times with TMWR\.

1. Use registers with tri-state outputs rather than uni-directional buffers (Figure 4.1: MVBC with 16-bit Traffic Memory) to delay the address by 1 clock cycle.
2. Extend the active level duration of RDY\ by 1 clock cycle.

18.1.6 Race Condition (RDY\ inactive permanently)

Severity: Severe! All nodes in Class 2/3/4 mode

Summary: If the MVBC accesses is controlled with TM_REQ_CPU\ (becoming active after RD\ or WR\), then RDY\ may become inactive permanently which results to a system hangup. This problem occurs only at a critical time delay after the rising clock edge which reflects a race condition problem. The problem lies in the synthesized model where gate optimization has departed from the original design!

Workaround: Synchronization of TM_REQ_CPU\ to the falling (!) clock edge.

According to my applications where the MVBC has been used:
At least activation of TM_REQ_CPU\ must be synchronized. Synchronization of deactivation of this signal is not mandatory.
In addition, I have gated RD\ and WR\ with active low synchronized TM_REQ_CPU\.
This solution has proven to be working.
See also section 18.1.7, so this workaround covers both problems.

18.1.7 TMRD\ and TMWR\ active while TMCS\ inactive

Severity: All nodes in Class 2/3/4 mode
Summary: If the MVBC arbitration logic is in "CPU" mode, then RD\ and WR\ propagate to TMRD\ and TMWR\, regardless if TM_REQ_CPU\ is active or not.
Workaround: Use all three signals to access memory: TMCS\, TMRD\ and TMWR\! Or, gate TM_REQ_CPU\ with RD\ and WR\.

18.1.8 Startup Problems

Severity: All nodes in Class 2/3/4 mode
Summary: The MVBC may not operate properly after startup: No telegrams sent or properly received. In some target systems (e.g. EST-23 500AIP100), the problems require software reset (IL=0) to recover MVBC. In other target systems (e.g. EST-23 I-PACK), a hard reset is required. I assume that the cause of this problem must sit somewhere in the test support. Probability: Those requiring soft resets: ca 1:30. Those requiring hard resets: ca 1:200. Diagnosis: Send a telegram in test mode and check for success.
Workaround: MVBC Reset, try soft reset first. If it does not work, try hard reset.

18.1.9 Reset Signal

Severity: All MVBC's
Summary: Currently, the reset signal is distributed by a strong clock driver to all flip flops. However, the rising edge seems to need some synchronization, otherwise some flip flops start running and some other's don't if the reset deactivates in a near the rising clock edge.
Workaround: Synchronize MVBC reset signal with falling (evtl rising possible) clock edge

18.1.10 Race Condition (RDY\ inactive permanently) - New Findings

Severity: All nodes in Class 2/3/4 mode, not operating in arbitration mode 0
Summary: Following malfunction is observed:

1. The CPU accesses the Traffic Memory or MVBC and that was successful.
2. One clock cycle later, the MVBC tries to access the Traffic Memory and hangs up. TMCS\ and one of TMWR\ or TMRD\ stay active permanently.
3. When the CPU also attempts to access the MVBC or Traffic Memory, then RDY\ changes from 0 to 1 and stays at 1 permanently. Now, both CPU and MVBC are hung.

This problem is observed only in some MVBC's. The possible cause is an internal race condition. Seen from the CPU, the MVBC behavior is similar to that described in section 18.1.6
Workaround: Use Arbitration Mode 0

18.2 MVB Physical Layer Signals

18.2.1 Interframe Spacing

Severity: All nodes - not a real problem
Summary: The MVBC maintains a minimum frame spacing period between end of Master Frame and beginning of Slave Frame of 1.6 µs instead of 1.4 µs. The frame spacing between Slave and following Master Frame of 4.0 µs stays unchanged.
Workaround: This bug should not cause any problems. The loss of effective bandwidth is not significant.

18.2.2 SF Signal

Severity: All nodes - not a real problem
Summary: SF becomes active between 40.67 and 333 ns (steps of 40.67 ns) before start bit is transmitted. Normally, 125 ns is foreseen.
Workaround: Not necessary unless application contains some very strict timing. In this case, OC would need to be delayed by one clock cycle.

18.2.3 Device Status Report / Nonconformance

Severity: Harmless - It's a change of requirement
Summary: In Class 1 Mode, MVBC replies with device type "1111" inside a Device Status Report. The new normative documents require "0000" to be replied.
Workaround: Not necessary

18.3 Telegram Analysis Unit

18.3.1 Frame and Error Counters, Part 1

Severity: All nodes which do error statistics are affected
Summary: Normally, Event Polls (F-Codes 9, 13 and 14) shall not affect the Frame and Error Counters (FC, EC). Only the Master Frames are counted. Reason: Reply Timeouts and collisions are legal. The MVBC obeys this rule only if it processes a port (actually sending or receiving Event Frames). If the MVBC does not process a port (inactive sink port, or no event announced or participating), then all Event Frame and possible telegram errors are counted. This leads to invalid, very pessimistic, error statistics.
Workaround: If the counters are checked by software, then the MVBC should always receive Event Frames into the Event Frame Sink Port (EFS), even if the node is not a bus administrator. EFS must be configured as an active sink port.

18.3.2 Frame and Error Counters, Part 2

Severity: All nodes which do error statistics are affected
Summary: A Bus Timeout situation (1.3 µs elapsed) leads to incrementing the Error Counter (EC) by 1. The Frame Counter (FC) remains unchanged.
Workaround: Either, the SW accepts Bus Timeouts as part of its statistics, or it must correct the EC value whenever a BTI occurs.

18.3.3 RLD Bit not Cleared in Redundant Mode

Severity: All MVBC's operating with 2 redundant lines (Decoder: SLM-Bit=0). It is an inconvenient, but not a dangerous bug.
Summary: Reports have been repeated why the MVBC's addressed with Device Status Polls (F-Code 15) do not clear the RLD bit if operating in redundant mode, but everything works fine in non-redundant mode.

Reason: The MVBC memorizes line switching requests including the reason during the time while the RLD bit is active. The MVBC shall forget such requests, but does not.

Assume following scenario:

0. Starting condition: Decoder bits: LAA=1, RLD=1, LS=0, SLM=0
1. Valid MF is received. No slave replies. Timeout is detected on both lines A and B.
   Decoder bits: LAA=1, RLD=1, LS = 0, SLM=0
   The MVBC memorized that case and activates the hidden flag LS_DUE_TO_FAULT.
   It will stay active until satisfied with a true line switch -- rather than forgetting this.
2. A device status poll addresses the MVBC. It clears the RLD bit.
   Decoder bits: LAA=1, RLD=0, LS = 0, SLM=0 (LS_DUE_TO_FAULT = 1)
3. A 2nd device status poll addresses the MVBC. It enables line switching
   Decoder bits: LAA=0, RLD=1, LS = 0, SLM=0 (LS_DUE_TO_FAULT -> 0)
   The line switched, but RLD is set again, because to LS_DUE_TO_FAULT. This flag is cleared now, and automatically set again when the next timeout (e.g. during event polling) is detected.
4. Assuming no timeout took place, and a 3rd device status poll is made, following is revealed:
   Decoder bits: LAA=0, RLD=0, LS = 0, SLM=0 (LS_DUE_TO_FAULT stays at 0)
   RLD bit is erased. OK.
5. A 4th device status poll is made. A correct line switch takes place.
   Decoder bits: LAA=1, RLD=0, LS = 0, SLM=0 (LS_DUE_TO_FAULT stays at 0)

The lesson: The MVBC shall not set LS_DUE_TO_FAULT if the redundant line is incorrect. The cause of this problem is known.

Workaround: Try multiple consecutive device status polls to come to better results. If you make a write access to the Decoder Register, then LS_DUE_TO_FAULT will always be cleared. It will also help if you do this on a regular schedule.

18.3.4 Line Switch on Timeout

Severity: All MVBC's operating with 2 redundant lines (Decoder: SLM-Bit=0). It is an inconvenient, especially if the MVBC is the bus administrator and one MVB line is disturbed.

Summary: If RLD-bit is 0, and a reply timeout (e.g. after 42.7 us) occurs, then a line switch takes place. Usually, a line switch shall not take place since the timeout has indeed occurred on both lines.

The situation becomes nasty if following scenario applies (own MVBC is bus administrator):

1. Line A (or B) works fine, line B (or A) is broken (no replies)
2. LAA = 1 (for line B: LAA=0), SLM=0, RLD=0. The MVBC listens to the good line.
3. A Master frame is sent and then received on both lines (local echo effect)
4. No one replies (e.g. missing device) , so a timeout occurs on both lines A and B
5. The MVBC recognizes the timeout, switches lines, then sets RLD=1
6. A second master frame is sent+received, but the MVBC listens on the wrong line.
7. The 1.4 ms timeout will force the MVBC to switch back to the working line.

Alternative situaton (own MVBC is bus administrator):

1. Line A (or B) works fine, line B (or A) is broken (no replies)
2. LAA = 0 (for line B: LAA=1), SLM=0, RLD=0. The MVBC listens to the bad line.
3. A Master frame is sent and then received. It works fine due to local echo effect.
4. A slave replies and is recognized on line A (or B) - it is the observed but not trusted line
5. The MVBC recognizes the timeout, switches lines, then sets RLD=1
6. A second master frame is sent+received, but the MVBC listens on the right line now.

Effect: The MVBC misses one telegram. The situation can occur after a device status poll which cleared the RLD bit.

Workaround: Add an uncritical telegram transfer to a device which is not missing and is able to reply. In this case, the slave frame is received on one line and RLD is set accordingly.

18.4 MVBC Operation

18.4.1 Communication Startup Timeouts

Severity: All nodes affected if interrupts are used

Summary: When the MVBC is switched into test or full operational mode (Initialization Level $IL_{1..0}$=2 or 3, part of Status Control Register SCR), the timeout counters in the Telegram Analysis Unit start running as if a Master Frame has just arrived at this moment. The following symptoms appear:

1. After the Reply Timeout period has elapsed, the interrupt "RTI" (Reply Timeout Interrupt) is asserted. The Reply Timeout period depends on the setting of the Timeout Coefficient $TMO_{1..0}$ found in the SCR.
2. If the Decoder is not operating in Single-Line Mode (Decoder Register (DR): SLM=0), and the redundant line is not yet disturbed (DR: RLD=0), the redundant MVB input lines are switched when the Reply Timeout period has elapsed.
3. After 1.3 ms, the Bus Timeout Interrupt (BTI) is asserted.
4. The Telegram Analysis increments the error counter (EC) by 1 after Reply Timeout and again by 1 after Bus Timeout.

Workaround: Awareness of such timeouts, ignore interrupts at beginning

18.4.2 Cancel / Suspend sending Master Frames

Severity: Active masters affected

Summary: This error involves automatic transmission of multiple Master Frames (MF) from MF-Tables using Master Register (MR) commands SMFA and SMFT. The MVBC must be active sending Master Frames, meaning that, if SMFT is used, the timer has started the MF transmission.
This process can be stopped with CSMF (Clear Sending Master Frames). The MCU will discontinue processing the MF-Table. This process can also be suspended by setting the MAS bit (inside SCR) to zero and back to one.
After sending MFs has been stopped or temporarily suspended, then the subsequent MF-table will not be processed correctly. One Master Frame may be skipped.
This error does not occur if the Master Frame transmission request has been submitted with SMFT, and the timer pulse has not yet been issued.

Workaround: After canceling MF transmission, make an automatic MF transmission with SMFA which transfers 1 non-critical telegram, i.e. a Device Status Poll. Normally, CSMF is not used unless a bus administrator system must stop operation instantly, i.e. for diagnosis purposes.

18.4.3 MCU Deadlock

Severity: Bus administrators affected

Summary: This situation applies for bus masters only. The MCU enters a deadlock situation if all of the following conditions are met:

1. A Master Frame has been transmitted by this MVBC;
2. The Master Frame does not echo back properly into the incoming MVB lines (Possible causes: optical link without local or remote echo, heavy MVB interference, SCR QUIET-bit active, FCL\-Pin active);
3. Reply Timeout (RTI) and Bus Timeout (1.3 ms BTI) have already elapsed before this MF transmission.

After the MVBC is instructed to send a Master Frame (SMFM, SMFA, or SMFT after the timer has reached zero), the BUSY-bit becomes active and no condition to reset BUSY appears. Therefore, a mechanism must be activated to reset the BUSY-bit.

Workaround: One of the following actions must be taken in order to rescue the MVBC:

1. A foreign bus administrator detects the silent bus and takes over bus mastership. Upon arrival of a Master Frame, this MVBC continues operation.

2. The software detects the silent state (absence of interrupts), changes Initialization Level $IL_{1..0}$ to 1 (Configuration Mode) and back to running or test mode. The MVBC takes advantage of the error summarized in section 18.4.1 since the Reply Timeout will reset the BUSY-bit and the MCU can continue with the next Master Frame
3. The tough approach: Software reset ($IL_{1..0}$=0). This approach works, but makes no sense since item 2 works well. Item 3 resets most MVBC registers!

18.4.4 Queued Message Data and Sink-Time Supervision

Severity: MVBC in class 2/3/4 mode are affected

Summary: Simultaneous use of sink-time-supervision and queued message data transfers does not work. The sink-time supervision messes up the message data queue.

Workaround: Two choices

1. Take advantage of MVBC's queued message data transfers, but realize sink-time-supervision completely with software (MVBC supervision must be turned off!)
2. Take advantage of MVBC's sink-time-supervision, but do not use queued message data transfers (QA-bits in PCS must be inactive). Incoming message data packets must be retrieved directly from the message sink port (MSNK) and outgoing message data packets must be placed into the message source port (MSRC).

18.5 Interrupt Controller

18.5.1 Interrupt Status: Last-Minute Interrupts

Severity: All nodes affected if interrupts are used

Summary: The Data Sheet says that a read access to an interrupt status register ISR0 (or ISR1) freezes the Interrupt Controller part zero (or one). The freezing takes place 1 clock cycle after the read access has taken place. This means:
In very rare cases, a last-minute interrupt may slip through the 1-clock cycle lasting *yellow-light* period into the Interrupt Status Register. Therefore, a second read access to the ISR0 or IVR0 (or ISR1/IVR1) is necessary.

Workaround: If interrupts are processed by examining the ISR0 (or ISR1):

1. If interrupts are processed by examining the ISR0 (or ISR1):
   ISR0 (or ISR1) must be read at least twice.
2. ISR0 (or ISR1) must be read once, then processing IVR0 (or IVR1) can start.
   ISR0 (or ISR1) must be read once, then processing IVR0 (or IVR1) can start.

18.6 Device Address

18.6.1 Very Noisy Device Addresses

Severity: All nodes in Class 2/3/4 mode

Summary: If the device address applied to the MVBC is not stable (e.g. it is not fixed to fixed voltage levels, or signal disturbance is big enough to induce successful crosstalk which leads to continuously changing device addresses), the MVBC may hang up by interpreting all incoming Master Frames as "Master Frame Error" and all slave frames as "Slave Frame Error", regardless of the quality of the signal. The hang-up does typically take place when the MVBC processes a device-addressed telegram and the device address changes at that moment. The only way to revert this is a hard reset

Workaround: Make sure the device address is stable (crosstalk shall not reach HCMOS switching level). If this cannot be achieved, then read the device address from MVBC register DAOR, write it back to DAOR and set DAOK to activate device address overriding. If this is made, then the unstable or disturbed device address can no longer influence the MVBC.

19 APPENDIX K: INDEX

(Space below has been left blank intentionally)

Document: 3EHL 420 441

Issue: -

Revision: C / BAL 8104

ABB Daimler-Benz Transportation (Switzerland) Ltd
P.O. box 8384
CH-8050 Zürich
Switzerland
Telefon +41 1 318 33 33
Telefax +41 1 312 61 59