Preliminary Data Sheet, DS 1, October 2001

IWORX
Interworking Controller
PXF 4225 Version 1.1 A12
with Firmware Release 1.1-2.x.x

Datacom

Infineon technologies

Never stop thinking.

Edition 2001-10-05

Published by Infineon Technologies AG,
St.-Martin-Strasse 53,
D-81541 München, Germany

Attention please!

The information herein is given to describe certain components and shall not be considered as warranted characteristics.

Terms of delivery and rights to technical change reserved.

We hereby disclaim any and all warranties, including but not limited to warranties of non-infringement, regarding circuits, descriptions and charts stated herein.

Infineon Technologies is an approved CECC manufacturer.

Information

For further information on technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies Office in Germany or our Infineon Technologies Representatives worldwide (see address list).

Warnings

Due to technical requirements components may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies Office.

Infineon Technologies Components may only be used in life-support devices or systems with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support device or system, or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body, or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable to assume that the health of the user or other persons may be endangered.

IWORX
Interworking Controller
PXF 4225 Version 1.1 A12
with Firmware Release 1.1-2.x.x

Datacom

Never stop thinking.

PXF 4225
CONFIDENTIAL

Revision History:	**2001-10-05**	DS 1
Previous Version:	Preliminary Data Sheet IWORX V1.1 2001-04-20 DS3	

Page	Subjects (major changes since last revision)
	Changes between IWORX V1.1 and IWORX V1.1 A12 regarding pull-up/ pull-downs in the chapter 2.2 are marked by change bars
	Enhancement of AAL1 traffic class description on page 106, 118
	Correction of start event of RX CAS conditioning on page 72
	Correction of values in chapter 6

For questions on technology, delivery and prices please contact the Infineon Technologies Offices in Germany or the Infineon Technologies Companies and Representatives worldwide: see our webpage at http://www.infineon.com

Table of Contents **Page**

List of Figures **Page**

CONFIDENTIAL

List of Tables **Page**

CONFIDENTIAL

1 Overview

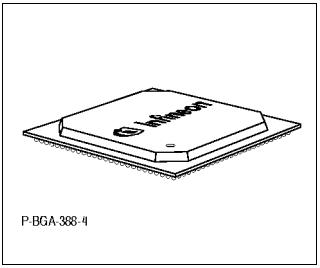
The IWORX is a flexible and upgradable communication network processor for ATM interworking. When used as an interworking element, its applications are in line cards of multiservice base stations or adapting voice, video, and data traffic to ATM. When used in ATM mode, its applications are for providing modular bandwidth for access to ATM networks over multiple lower speed physical links.

In order to satisfy multiservice ATM interworking applications, IWORX supports AAL1, AAL2 adaption layers. In the ATM mode, IWORX provides constant and variable bit rate services to support voice, video, and data applications. Each port can be independently selected for an adaptation protocol. This feature enables switch manufacturers to build a multiservice card that can be flexibly configured for operation in either AAL1, AAL2 or ATM UNI/NNI mode. IWORX also supports sophisticated traffic management functions such as WFQ scheduling and buffer management. IWORX provides independence from the fast moving external RAM market because all required RAM is integrated internally.

For higher performance, processing intensive tasks (e.g., AAL2, AAL1, OAM etc.) are realized in hardware. In order to provide high degree of flexibility for developing derivative SW functions and architectures, power full on chip microcontroller is provided.

Version 1.1 A12

1.1 Features



- Interworking controller that combines G.804, AAL1 and AAL2 CPS
- Support for up to 8 T1/E1/J1 PDH ports
- On chip controller for flexible protocol handling, configurations and customization
- Independence from fast moving RAM market (all required RAM is integrated)
- Port specific mode selection (e.g., rate: T1/E1, service: full or fractional G.804)
- Time slot specific protocol selection (e.g., AAL1, G.804)
- Support for split AAL1 and G.804 on the same T1/E1 link
- Dedicated hardware modules for performance intensive communication functions such as AAL2 CPS, AAL1, OAM and traffic management
- Support for 256 ATM connections
- Supports proprietary ATM cell formats up to 64 bytes (e.g., external routing tag)
- AAL1 mode (as per ATM Forum Circuit Emulation Specification):
  - Support for structured and unstructured CES mode for E1/T1/J1 links
  - Support for Channel Associated Signalling (CAS)
  - Built in SRTS and ACM clock recovery methods for up to 8 E1/T1 links
  - Optionally, it's possible to order the device with FW release 1.1-1.2.1, which comes without SRTS clock recovery.
  - Support for up to 256 AAL1 ATM VCCs
- AAL2 mode (as per ITU-T I.363.2)
  - Mux/demux of up to 249 AAL2 packets per AAL2 ATM VCC
  - Support for up to 249 x 16 = 3984 AAL2 CIDs
  - Support for up to 16 AAL2 ATM VCCs (i.e., VCCs can be arbitrarily associated with any port(s))
  - AAL2 timer_CU is adjustable from 0.1 to 10 ms with a resolution of 0.1 ms
  - Receive ATM cells via Utopia containing exactly one AAL2 minicell
  - Extract minicell from an ATM cell and mux multiple minicells on to ATM cell via AAL2

Type	Package
PXF 4225	P-BGA-388

  - In the reverse direction, extract multiple AAL2 minicells from an ATM cell
  - For each AAL2 minicell, generate an ATM cell towards Utopia
  - AAL2 VCCs traffic management related functions in the transmit direction
- Support for ATM UNI/NNI mode (i.e., only ATM cell mapping into PDH)
  - ATM TC sublayer according to G.804
  - Any T1/E1/J1 link can be configured in ATM mode
  - Support for G.804 on fractional T1/E1 according to ATMF-FN64
- UTOPIA interface according to ATM Forum standard:
  - Up to 64 bytes flexible Utopia cell format
  - Level 1 PHY or ATM mode
  - Level 2 Multi-PHY mode
  - 8 or 16 bit data bus
  - Cell level handshake
  - Maximum frequency: 50MHz
- Traffic Management:
  - Supports CBR, VBR-rt, VBR-nrt, GFR, and UBR+ QoS queues
  - Per VC queuing for up to 255 connections configurable in transmit direction
  - Provides weighted fair queuing (WFQ) scheduling
  - Optional per-VC PCR limiting
  - Buffer management: early packet discard (EPD), partial packet discard (PPD), cell discard (CLP 0+1)
- OAM support :
  - Provides OAM F4 and F5 cell insertion and extraction
  - OAM function Fault Management (AIS,RDI, Loopback, CC)
  - OAM Performance Monitoring for two selected connections
  - Provides on-chip CRC-10 and BIP16 calculation
- Microprocessor Interface:
  - Standard microprocessor interface (Intel or Motorola mode)
  - Data width can be 16/32 bit
- General:
  - Boundary scan support according to IEEE 1149.1
  - Temperature range -40°C to +85°C
  - Package P-BGA-388
  - Max. Power-Dissipation less than 3.0 Watt
  - 1.8 V for logic and 3.3 V for I/O pins
- Support for loop backs (Ingress and egress loops at interfaces)
- SW Deliverables:
  - firmware for embedded controller (executable binary), software device drivers

1.1.1 Functional Characteristics

- Image size is less than 0.5 MByte
- AAL2 channel setup/release in less than 50 µs
- If AAL1 is used one of 255 real-time queue and one scheduler of 16 is assigned for the AAL1 traffic in the traffic management block.
- If AAL2 CPS is used the first 16 connection handles 0..15 are reserved for AAL2 paths
- UTOPIA interface can accept a burst rate of 480 Mbit/s without backpressure
- Up to 256 ATM connections can be handled due to the existence of max. 256 connection handles; a VPC is one of 256 possible ATM connections if the VCI bits of this connection are masked; this masking can be done individually for each connection.
- No stand-alone VCC are expected, VCCs hould have an associated terminated VPC.
- VPI/VCI of the ATM header of an ATM connection is expected to be identical at the RX PHY side and TX PHY side of IWORX.

1.2 Logic Symbol

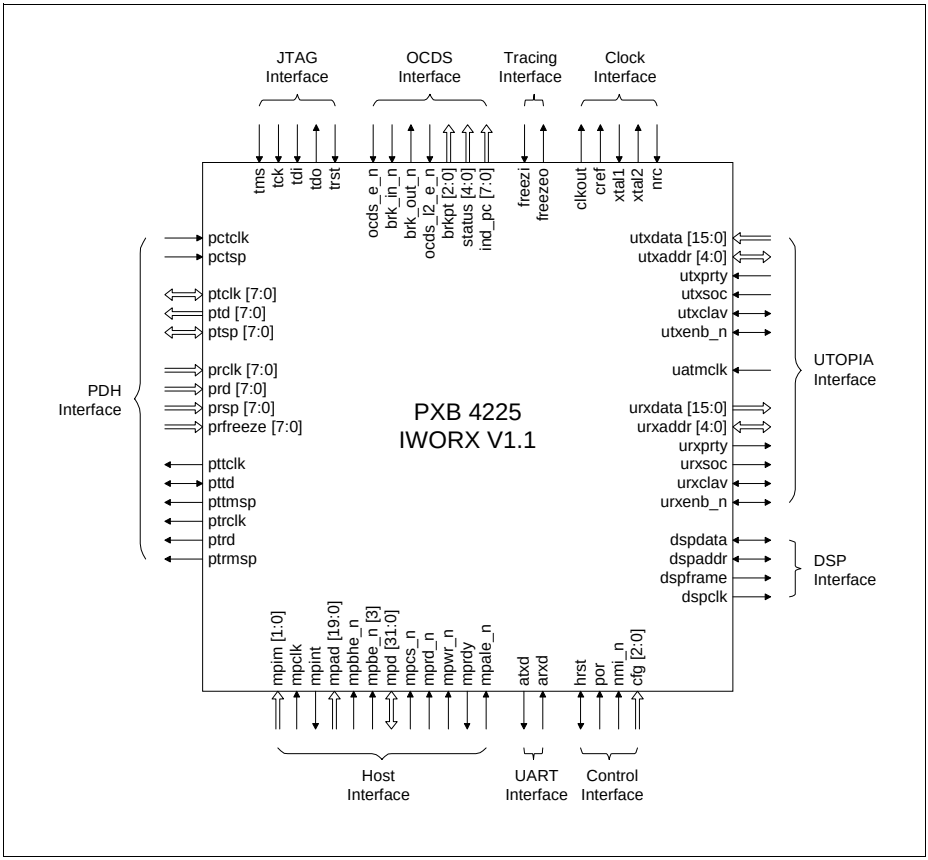


Figure 1 Logic Symbol

1.3 Typical Applications

1.3.1 Introduction

Third-Generation (3-G) Mobile Communication Systems

Due to worldwide trends for connectivity which in part is motivated by the catalytic presence of the Internet, the market for mobile computing is expected to grow explosively. The consumers are demanding services such as seamless Internet/Intranet access and multimedia (voice, video, data) communication capabilities. These demands are driving service providers to evolve their infrastructure to meet new services requirements. Most of these requirements cannot be satisfied by today's second generation (2-G) mobile systems. As a result , intensive research and standardization activities are being focused on the design of third-generation (3-G) mobile communication systems. For instance, Universal Mobile Telecommunications System (UMTS) is one of the standards for 3-G mobile systems which is being harmonized with the International standardization activities for the International Mobile Telecommunications (IMT-2000) standard. These initiatives agree that the introduction of 3-G mobile systems will soon lead to the unification of the existing 2-G mobile system infrastructure.

Mobile Communication System Infrastructure

ATM is the technology of choice for the wired infrastructure in wireless cellular systems. As shown in Figure 3, the wireless infrastructure consists of mobile terminals (MT), air interface (AI), base transceiver stations (BTS) which terminate the air interface, the control nodes which manage the radio resources (in UMTS terminology these nodes are called radio network controllers or RNCs), mobile switching center (MSC), and the point of interface (POI) between wireless and the public switched telephone network (PSTN). The BTS and RNC are typically connected by means of T1/E1 links which are assumed to be ATM-based.

Motivations for AAL2 in Wireless Communication Systems

On the air interface, packets are typically very small and of variable size. For instance, the packet size may vary between 2 to 36 octets for code division multiple access (CDMA). The existing AALs (e.g., AAL1, AAL5) are very inefficient for such applications because these AALs require partial filling of ATM cells. The AAL2 is designed for bandwidth-efficient transmission of low rate, short, and variable packets in delay sensitive applications. For efficient transmission of such small delay-sensitive packets, it is necessary to multiplex packets of several calls over AAL2. Compressed voice packets from multiple users are multiplexed over an AAL2 VC. At the public-wireless carrier interface, these packets are converted back to 64 kb/s PCM streams for transport over PSTN (see **Figure 3**).

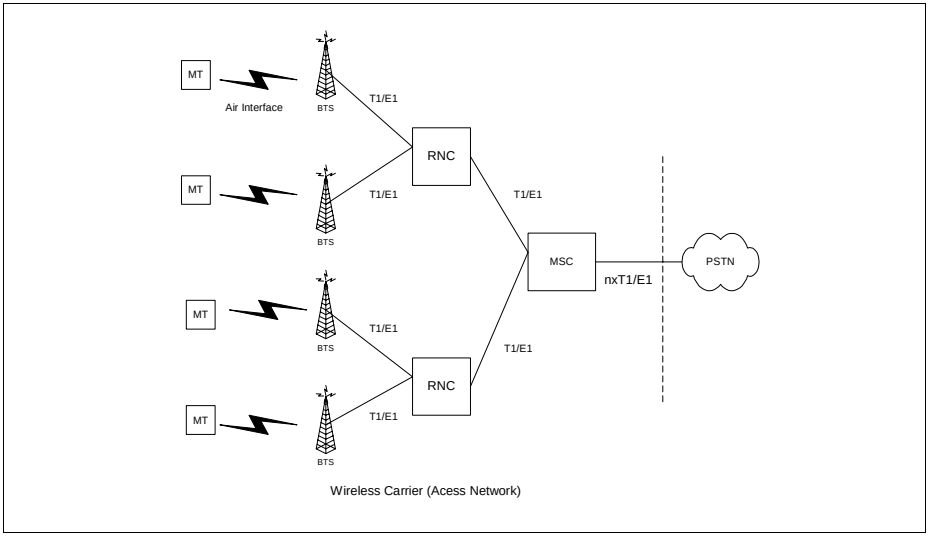


Figure 2 Wireless Communication Network Model

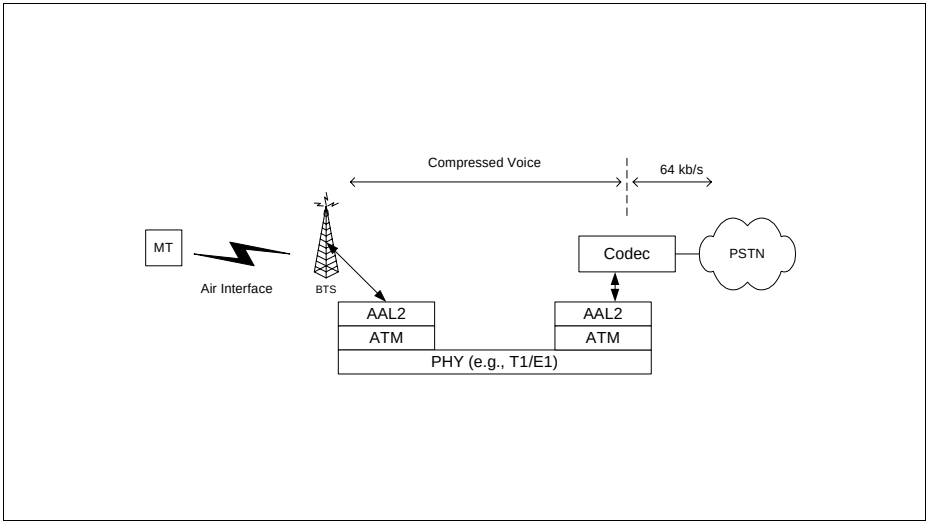


Figure 3 Voice Transmission from Mobile Terminal to PSTN

An End-to-end Application Example

Based on the research and standardization activities, there seems to be general trend that in the 3-G mobile systems the AI would be based on wideband CDMA and the transport infrastructure would be based on ATM. In this section, we present an end-to end application example to gain an better overall understanding of applications. The following assumption are made in this example:

- The BTS implements the CDMA layer 1 functionality (e.g., power control, channel coding, modulation, multiplexing of physical channels etc.). Furthermore it is assumed that AAL2/ATM multiplexing is implemented by the BTS.
- The RNC implements diversity handover (DHO), medium access control (MAC) and the logical link control (LLC).
- The IWU (also known as adaptation function or AF) is responsible for converting and relaying the arriving AAL2 stream over the core network specific transport infrastructure. For example the IWU between MSC and PSTN core network is responsible for conversion between compressed voice to uncompressed 64 kb/s PCM.

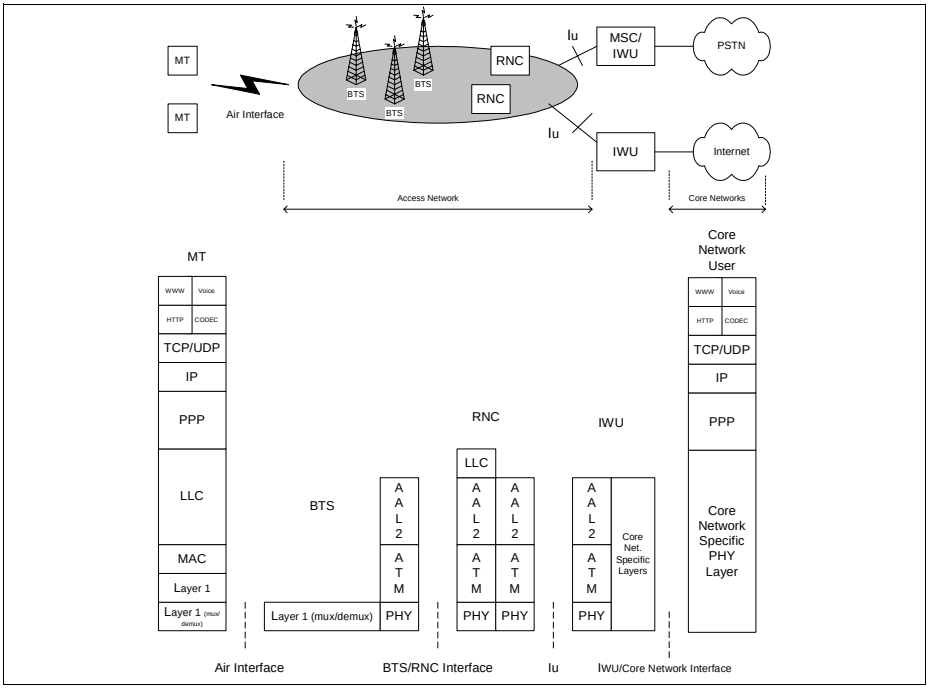


Figure 4 Application Example in 3-G Wireless Communication Systems

1.3.2 IWORX Applications in Cellular Infrastructure

As described earlier, in 3-G mobile systems the transport infrastructure between BTS BSC/RNC and MSC is assumed to be based on ATM. Furthermore AAL2 can be used to efficiently transport small and variable size packets for voice and data over ATM. Thus BTS, BSC/RNC, and MSC in cellular infrastructure provide target locations for application of the IWORX (see Figure 6). In the base station market, there is a general consensus that the AAL2 packets from external modules to IWORX will be transported via Utopia interface. This means that the external module encapsulates one AAL2 packet per ATM cell. Therefore IWORX receives ATM cells from external modules, extracts AAL2 packet from ATM cells, and then multiplex multiple AAL2 packets as per AAL2-CPS over an ATM VCC. In the reverse direction, for a given AAL2 VCC IWORX demultiplexes the AAL2 packets and encapsulates one AAL2 packet per one ATM cell and sends these cells (containing one AAL2 packet per ATM cell) to the desired external module via Utopia interface. In addition to AAL2, IWORX also provides ATM/IMA modes for access to ATM networks over nxT1/E1. Thus IWORX enables base station vendors to develop multiservice and flexible architectures to meet challenging service requirements of the 3-G mobile systems.

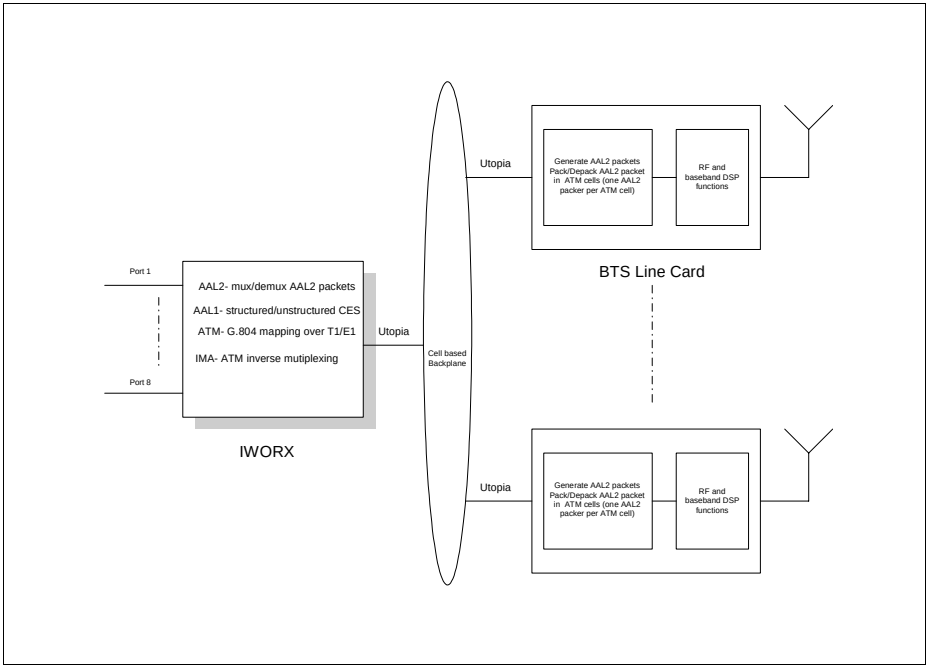


Figure 5 System Level Diagram of IWORX Base Station Application

1.3.3 IWORX Applications in Broadband Wireless Access

The broadband wireless access market is driven by business (e.g., internet/intranet access, voice over IP, corporate video broadcasting, LAN/WAN interconnection) and residential (e.g., internet access, digital broadcast video, voice over IP, e-commerce) applications. A broadband wireless access network consists of customer premises equipment (CPE) including network interface unit (NIU), and indoor unit providing circuit emulation and ethernet data services, and an outdoor integrated transciver and antenna. Paired with the NIUs across the air interface is the base station transceiver which operates in microwave frequency range between 5 to 42 GHz. The base station is typically connected to an WAN switch via an OC-3 link. NIUs typically have 4 to 16 T1 E1 ports and 10 BaseT interfaces to meet required customer bandwidth requirements. The future NIUs will support a wide range of interfaces including DS0, POTS, structured unstructured CES T1/E1,10 BaseT, and frame relay. Due to its rich set of features and high level of integration, IWORX provides an excellent solution for NIU's line cards for supporting AAL1, AAL2, and IMA.

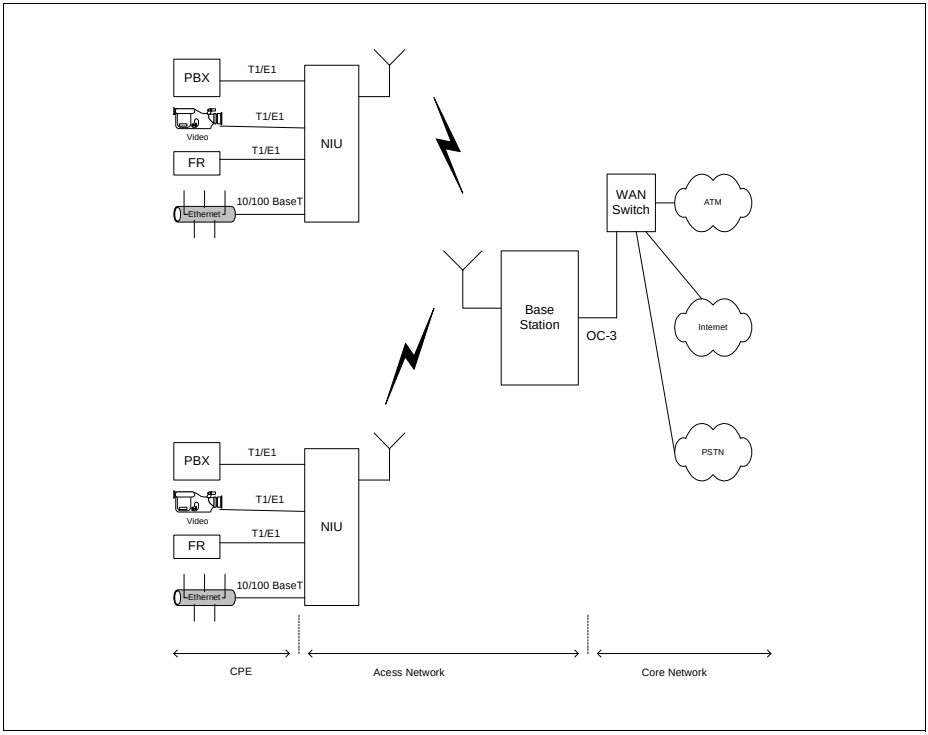


Figure 6 Broadband Wireless Access Infrastructure

1.3.4 IWORX Applications in Integrated Wireless Access

In order to reduce the high cost associated with leased T1/E1 lines, in some areas(e.g., Europe) service providers are considering to replace expensive leased lines with Point to Multipoint broadband wireless links. An added benefit of this approach is that it allows a service provider to offer an integrated (fixed and mobile) broadband wireless solution as depicted in Figure 8. As described before, IWORX can be used to design BTSs and NIUs line cards.

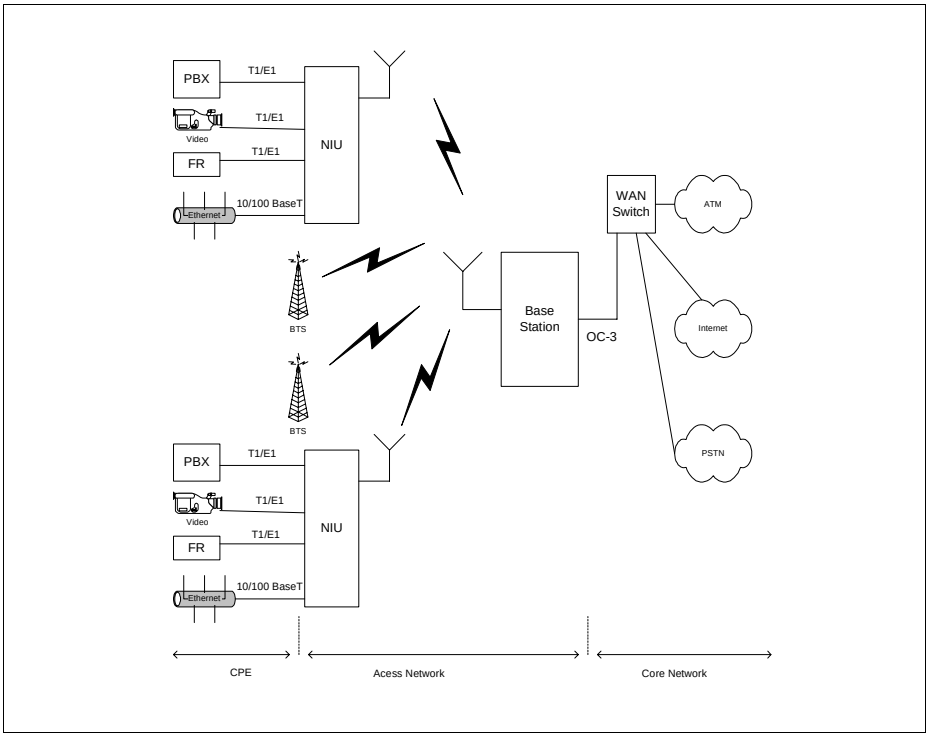


Figure 7 IWORX Applications in Integrated Wireless Access

1.4 Coding of the Firmware Release

The coding of the firmware release is done in the following way:

Release a.b-c.d.e means

- This firmware release can be downloaded to the IWORX HW Va.b (a.b = HW ID)
- This firmware release is step c of the planned firmware steps (c = FW functionality ID)
- This firmware release is variant d of step c with full or reduced functionality of step c (d = FW functionality subset ID)
- This firmware release has been corrected (e - 1) times (e = FW development status ID)

As an example release 1.1-3.2.3 would mean that the firmware is developed for the HW version 1.1 and that this is the 3rd functionality step with the reduced functionality variant 2 (without SRTS), 2 times corrected.

1.5 Differences Between Firmware Release 1.1-1.1.x and 1.1-1.2.x

The IWORX uses an internal clock recovery mechanism (SRTS) which is patented by Bellcore.

Related Patents are:

- Bellcore patent No. 5,260,978 (Synchronous Residual Time Stamp for Timing Recovery in a broadband network)
- Bellcore patent No. 4,839,306 (Method and apparatus for multiplexing circuit and packet traffic)

Infineon Technologies is not allowed to collect SRTS license fees on the IWORX on behalf of Bellcore. Contacts for license issues are given in **Chapter 8**.

Every IWORX customer must get in contact with Bellcore legal department by himself to clarify whether his application needs to license the SRTS functionality.

For customers who do not want to use the built-in SRTS mechanism, Infineon provides a special FW release. The release of this firmware is 1.1-1.2.x and covers the same functionality like the release 1.1-1.1.x but without SRTS clock recovery. SRTS is permanently disabled and can't be enabled by the user, so that no patent fees have to be paid.

For controlling which FW release is working in the IWORX, it exist a FW release output pin FWR. The output behaviour is described in **Chapter 5.7**.

1.6 Terminology

This chapter gives an overview how terms are used in this documentation.

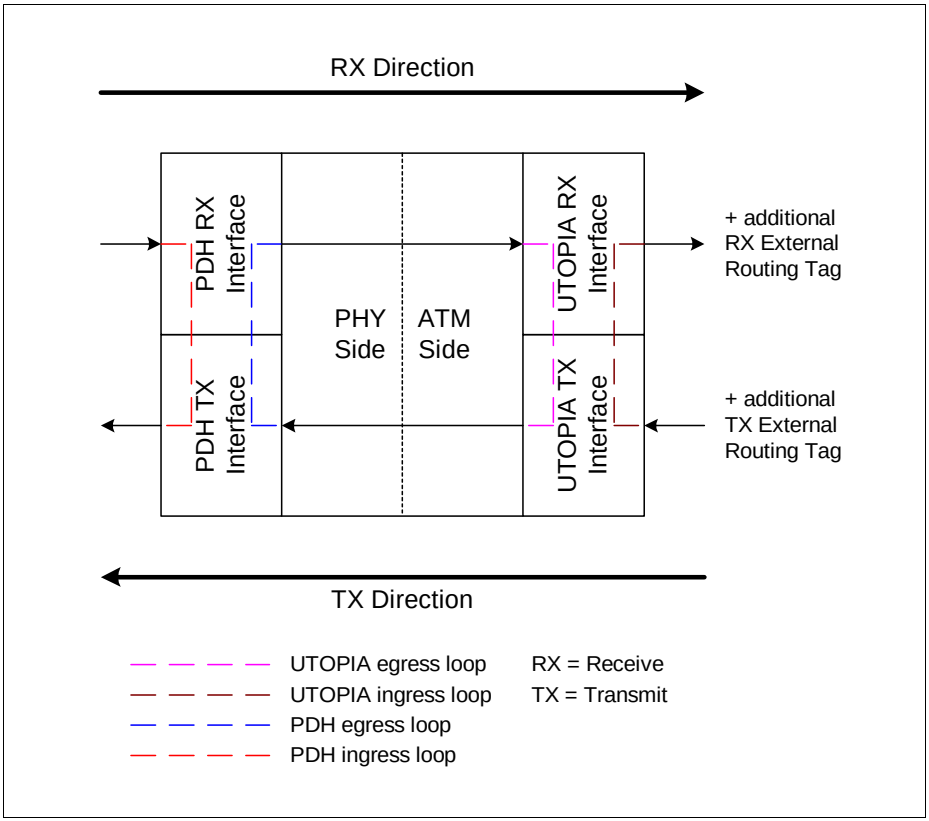


Figure 8 Terminology for Directions and Loops

The abbreviation RX is used for Receive and TX for Transmit

Data flow direction from PDH RX interface to UTOPIA RX interface is called RX direction and the data flow direction from UTOPIA TX interface to PDH TX interface is called TX direction.

The chip can be divided into a part to UTOPIA interface called ATM side and a part to PDH interface called PHY side.

The loop from PDH RX interface to PDH TX interface is called PDH ingress loop. The loop from PDH TX interface to PDH RX interface is called PDH egress loop. The loop from UTOPIA RX interface to UTOPIA TX interface is called UTOPIA egress loop. The loop from UTOPIA TX interface to UTOPIA RX interface is called UTOPIA ingress loop. The terms egress and ingress refere to the interface.

Transparent loops are loops where data is looped backwards but also forwarded as shown for the transparent PDH egress loop in **Figure 9**.

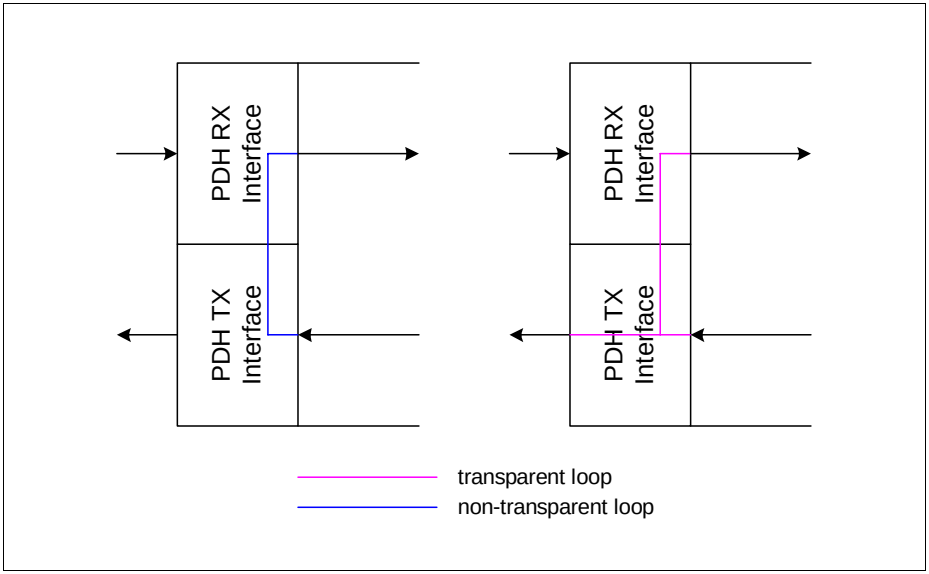


Figure 9 Example for Transparent and Non-transparent Loop

The OAM layer point description is done by a 3-part abbreviation as shown in **Figure 10**. The first part contains the information if it's VPC or VCC. The third part contains the information in which direction the layer point is located, RX or TX. The second part. The second part contains the information if it's an intermediate point IP, Originating e-t-e Point OEP, Terminating e-t-e Point TEP, Originating seg Point OSP or Terminating seg Point TSP. The terms originating/terminating are used instead of the ITU I.610 terms source/sink.

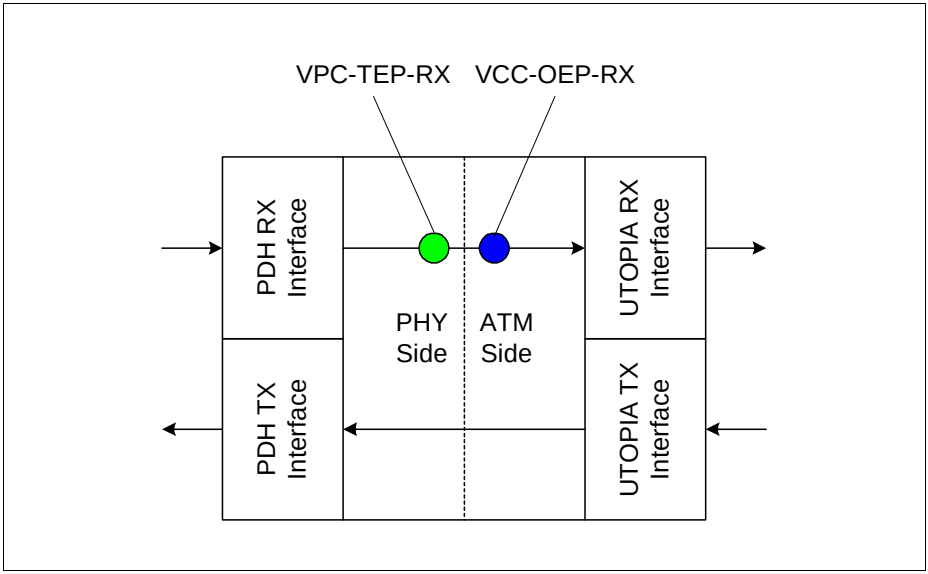


Figure 10 Examples for OAM Layer Point Abbreviations

An AAL2 channel is a connection which contains AAL2 CPS packets from one voice channel (CID is the same in all CPS packets).

An AAL2 path contains several multiplexed AAL2 channels (see also **Figure 34**).

A logical channel is an Nx64Kbit/s fraction within an E1/T1/J1 frame which is described by the used timeslots of the frame.

2 Pin Description

Table 1 gives an overview about the pin utilization per interface. Beyond the signal count of these interfaces there is a variety of test signals, which is made available either directly (using dedicated pins) or multiplexed with existing signal I/O pins. These test signals are not shown in **Table 1**. For complete pinning information, including multiplexed signals, driver strengths, pullups/pulldowns, refer to the following sections, which cover all pins.

Table 1 External Interfaces

Interface	Function	Signal Count
PDH [0..7]	8 TDM highways and common pins	64
DSP	Serial multimaster link	4
UTOPIA	ATM cell based interface 16 bit data, 5 bit addresses	51
MPI	Microprocessor interface in Intel/Motorola format 32 bit data, 20 bit address	61
UART	Universal asynchronous receiver/transmitter interface	2
JTAG	Boundary scan interface	5
OCDS L2	Debugger Interface	19
Scan		2
Tracing	Freeze input/output	2
Clocking		8
Control	Reset, NMI, boot mode cfg	7

2.1 Pin Diagram

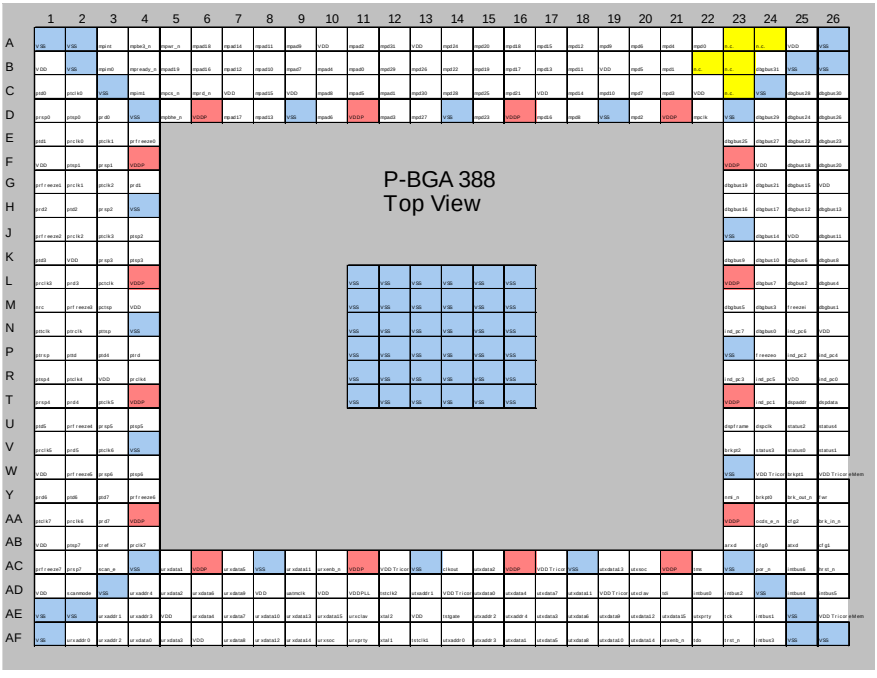


Figure 11 Pin Configuration

2.2 Pin Definitions and Functions

The following tables contain the pin information. The 1st column contains the ball number, the 2nd the pin name. The 3rd column describes if it is an input, output or bidirectional I/O pin. If it's an output or I/O the 4th column lists the driver strength. Strengths of the different driver types are specified in **Chapter 6.4**. The 5th column lists if the pin has an internal pull up PUx or an internall pull down PDx. The third letter of the used abbreviation decribes the strength of the pull up or pull down. Strengths of the different pull up/down types are specified in **Chapter 6.4**. The last column contains a short description of the pin function.

2.2.1 PDH Interface

Table 2 PDH Interface

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
C2	PTCLK0	I/O	D	PDB	PDH Transmit Clock
E3	PTCLK1	I/O	D	PDB	
G3	PTCLK2	I/O	D	PDB	
J3	PTCLK3	I/O	D	PDB	
R2	PTCLK4	I/O	D	PDB	
T3	PTCLK5	I/O	D	PDB	
V3	PTCLK6	I/O	D	PDB	
AA1	PTCLK7	I/O	D	PDB	
C1	PTD0	O	D	PUB	PDH Transmit Data
E1	PTD1	O	D	PUB	
H2	PTD2	O	D	PUB	
K1	PTD3	O	D	PUB	
P3	PTD4	O	D	PUB	
U1	PTD5	O	D	PUB	
Y2	PTD6	O	D	PUB	
Y3	PTD7	O	D	PUB	

Table 2 PDH Interface

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
D2	PTSP0	I/O	D	PUB	PDH Transmit Sync Pulse
F2	PTSP1	I/O	D	PUB	
J4	PTSP2	I/O	D	PUB	
K4	PTSP3	I/O	D	PUB	
R1	PTSP4	I/O	D	PUB	
U4	PTSP5	I/O	D	PUB	
W4	PTSP6	I/O	D	PUB	
AB2	PTSP7	I/O	D	PUB	
D3	PRD0	I		PDB	PDH Receive Data
G4	PRD1	I		PDB	
H1	PRD2	I		PDB	
L2	PRD3	I		PDB	
T2	PRD4	I		PDB	
V2	PRD5	I		PDB	
Y1	PRD6	I		PDB	
AA3	PRD7	I		PDB	
D1	PRSP0	I		PUB	PDH Receive Sync Pulse
F3	PRSP1	I		PUB	
H3	PRSP2	I		PUB	
K3	PRSP3	I		PUB	
T1	PRSP4	I		PUB	
U3	PRSP5	I		PUB	
W3	PRSP6	I		PUB	
AC2	PRSP7	I		PUB	

Table 2 PDH Interface

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
E2	PRCLK0	I			PDH Receive Clock
G2	PRCLK1	I			
J2	PRCLK2	I			
L1	PRCLK3	I			
R4	PRCLK4	I			
V1	PRCLK5	I			
AA2	PRCLK6	I			
AB4	PRCLK7	I			
E4	PRFREEZE0	I		PDB	PDH Receive Freeze, forces CAS freezing
G1	PRFREEZE1	I		PDB	
J1	PRFREEZE2	I		PDB	
M2	PRFREEZE3	I		PDB	
U2	PRFREEZE4	I		PDB	
W2	PRFREEZE5	I		PDB	
Y4	PRFREEZE6	I		PDB	
AC1	PRFREEZE7	I		PDB	
L3	PCTCLK	I			PDH Common Transmit Clock, selectable for all PDH ports
M3	PCTSP	I		PUB	PDH Common Transmit Sync Pulse
N1	PTTCLK	O	D		PDH Test Transmit Clock
P2	PTTD	I/O	D	PUB	PDH Test Transmit Data
N3	PTTSP	O	D	PUB	PDH Test Transmit Sync Pulse
N2	PTRCLK	O	D		PDH Test Receive Clock
P4	PTRD	O	D	PDB	PDH Test Receive Data
P1	PTRSP	O	D	PUB	PDH Test Receive Sync Pulse

2.2.2 UTOPIA Interface

Table 3 UTOPIA Interface

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
AF2	URXADDR0	I/O	C	PUB	UTOPIA Receive Address
AE3	URXADDR1	I/O	C	PUB	
AF3	URXADDR2	I/O	C	PUB	
AE4	URXADDR3	I/O	C	PUB	
AD4	URXADDR4	I/O	C	PUB	
AF4	URXDATA0	O	C	PUB	UTOPIA Receive Data
AC5	URXDATA1	O	C	PUB	
AD5	URXDATA2	O	C	PUB	
AF5	URXDATA3	O	C	PUB	
AE6	URXDATA4	O	C	PUB	
AC7	URXDATA5	O	C	PUB	
AD6	URXDATA6	O	C	PUB	
AE7	URXDATA7	O	C	PUB	
AF7	URXDATA8	O	C	PUB	
AD7	URXDATA9	O	C	PUB	
AE8	URXDATA10	O	C	PUB	
AC9	URXDATA11	O	C	PUB	
AF8	URXDATA12	O	C	PUB	
AE9	URXDATA13	O	C	PUB	
AF9	URXDATA14	O	C	PUB	
AE10	URXDATA15	O	C	PUB	
AD9	UATMCLK	I			UTOPIA Clock, common for RX and TX interface
AF10	URXSOC	O	C	PUB	UTOPIA Receive Start Of Cell
AC10	$\overline{\text{URXENB}}$	I/O	C	PUB	UTOPIA Receive Enable
AE11	URXCLAV	I/O	C	PDB	UTOPIA Receive Cell Available
AF11	URXPRTY	O	C	PUB	UTOPIA Receive Parity

Table 3 UTOPIA Interface

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
AF14	UTXADDR0	I/O	C	PUB	UTOPIA Transmit Address
AD13	UTXADDR1	I/O	C	PUB	
AE15	UTXADDR2	I/O	C	PUB	
AF15	UTXADDR3	I/O	C	PUB	
AE16	UTXADDR4	I/O	C	PUB	
AD15	UTXDATA0	I		PUB	UTOPIA Transmit Data
AF16	UTXDATA1	I		PUB	
AC15	UTXDATA2	I		PUB	
AE17	UTXDATA3	I		PUB	
AD16	UTXDATA4	I		PUB	
AF17	UTXDATA5	I		PUB	
AE18	UTXDATA6	I		PUB	
AD17	UTXDATA7	I		PUB	
AF18	UTXDATA8	I		PUB	
AE19	UTXDATA9	I		PUB	
AF19	UTXDATA10	I		PUB	
AD18	UTXDATA11	I		PUB	
AE20	UTXDATA12	I		PUB	
AC19	UTXDATA13	I		PUB	
AF20	UTXDATA14	I		PUB	
AE21	UTXDATA15	I		PUB	
AC20	UTXSOC	I		PDB	UTOPIA Transmit Start Of Cell
AF21	$\overline{\text{UTXENB}}$	I/O	C	PUB	UTOPIA Transmit Enable
AD20	UTXCLAV	I/O	C	PDB	UTOPIA Transmit Cell Available
AE22	UTXPRTY	I		PUB	UTOPIA Transmit Parity

2.2.3 Microprocessor Interface

Table 4 Microprocessor Interface in Intel Demux Mode

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
D22	Unused	I			Unused
A22	MPD0	I/O	D	PUB	Microprocessor Data
B21	MPD1	I/O	D	PUB	
D20	MPD2	I/O	D	PUB	
C21	MPD3	I/O	D	PUB	
A21	MPD4	I/O	D	PUB	
B20	MPD5	I/O	D	PUB	
A20	MPD6	I/O	D	PUB	
C20	MPD7	I/O	D	PUB	
D18	MPD8	I/O	D	PUB	
A19	MPD9	I/O	D	PUB	
C19	MPD10	I/O	D	PUB	
B18	MPD11	I/O	D	PUB	
A18	MPD12	I/O	D	PUB	
B17	MPD13	I/O	D	PUB	
C18	MPD14	I/O	D	PUB	
A17	MPD15	I/O	D	PUB	
D17	MPD16	I/O	D	PUB	
B16	MPD17	I/O	D	PUB	
A16	MPD18	I/O	D	PUB	
B15	MPD19	I/O	D	PUB	
A15	MPD20	I/O	D	PUB	
C16	MPD21	I/O	D	PUB	
B14	MPD22	I/O	D	PUB	
D15	MPD23	I/O	D	PUB	
A14	MPD24	I/O	D	PUB	
C15	MPD25	I/O	D	PUB	
B13	MPD26	I/O	D	PUB	

Table 4 Microprocessor Interface in Intel Demux Mode (cont'd)

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
D13	MPD27	I/O	D	PUB	Microprocessor Data, cont'd
C14	MPD28	I/O	D	PUB	
B12	MPD29	I/O	D	PUB	
C13	MPD30	I/O	D	PUB	
A12	MPD31	I/O	D	PUB	
B11	MPAD0	I		PUB	16 bit mode: Microprocessor Address 0
	$\overline{\text{MPBE0}}$				32 bit mode: Microprocessor Byte Enable 0
C12	MPAD1	I		PUB	16 bit mode: Microprocessor Address 1
	$\overline{\text{MPBE1}}$				32 bit mode: Microprocessor Byte Enable 1
A11	MPAD2	I		PUB	Microprocessor Address 2..19
D12	MPAD3	I		PUB	
B10	MPAD4	I		PUB	
C11	MPAD5	I		PUB	
D10	MPAD6	I		PUB	
B9	MPAD7	I		PUB	
C10	MPAD8	I		PUB	
A9	MPAD9	I		PUB	
B8	MPAD10	I		PUB	
A8	MPAD11	I		PUB	
B7	MPAD12	I		PUB	
D8	MPAD13	I		PUB	
A7	MPAD14	I		PUB	
C8	MPAD15	I		PUB	
B6	MPAD16	I		PUB	
D7	MPAD17	I		PUB	
A6	MPAD18	I		PUB	
B5	MPAD19	I		PUB	

Table 4 Microprocessor Interface in Intel Demux Mode (cont'd)

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
A5	$\overline{\text{MPWR}}$	I		PUB	Microprocessor Write
C6	$\overline{\text{MPRD}}$	I		PUB	Microprocessor Read
B4	$\overline{\text{MPREADY}}$	O, open drain	D		Microprocessor Ready
D5	$\overline{\text{MPBHE}}$	I		PUB	16 bit mode: Microprocessor Byte High Enable
	$\overline{\text{MPBE2}}$				32 bit mode: Microprocessor Byte Enable 2
A4	$\overline{\text{MPBE3}}$	I		PUB	16 bit mode: unused 32 bit mode Microprocessor Byte Enable 3
C5	$\overline{\text{MPCS}}$	I		PUB	Microprocessor Chip Select
B3	MPIM0	I		PDB	Microprocessor Interface Mode **MPIM[1:0] = 10**
C4	MPIM1	I		PDB	
A3	$\overline{\text{MPINT}}$	O, open drain	D		Microprocessor Interrupt

Table 5 Microprocessor Interface in Asynchronous Motorola Mode

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
D22	Unused	I			Unused
A22	MPD0	I/O	D	PUB	Microprocessor Data
B21	MPD1	I/O	D	PUB	
D20	MPD2	I/O	D	PUB	
C21	MPD3	I/O	D	PUB	
A21	MPD4	I/O	D	PUB	
B20	MPD5	I/O	D	PUB	
A20	MPD6	I/O	D	PUB	
C20	MPD7	I/O	D	PUB	

Table 5 Microprocessor Interface in Asynchronous Motorola Mode (cont'd)

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
D18	MPD8	I/O	D	PUB	Microprocessor Data, cont'd
A19	MPD9	I/O	D	PUB	
C19	MPD10	I/O	D	PUB	
B18	MPD11	I/O	D	PUB	
A18	MPD12	I/O	D	PUB	
B17	MPD13	I/O	D	PUB	
C18	MPD14	I/O	D	PUB	
A17	MPD15	I/O	D	PUB	
D17	MPD16	I/O	D	PUB	
B16	MPD17	I/O	D	PUB	
A16	MPD18	I/O	D	PUB	
B15	MPD19	I/O	D	PUB	
A15	MPD20	I/O	D	PUB	
C16	MPD21	I/O	D	PUB	
B14	MPD22	I/O	D	PUB	
D15	MPD23	I/O	D	PUB	
A14	MPD24	I/O	D	PUB	
C15	MPD25	I/O	D	PUB	
B13	MPD26	I/O	D	PUB	
D13	MPD27	I/O	D	PUB	
C14	MPD28	I/O	D	PUB	
B12	MPD29	I/O	D	PUB	
C13	MPD30	I/O	D	PUB	
A12	MPD31	I/O	D	PUB	
B11	MPAD0	I		PUB	Microprocessor Address
C12	MPAD1	I		PUB	
A11	MPAD2	I		PUB	
D12	MPAD3	I		PUB	
B10	MPAD4	I		PUB	
C11	MPAD5	I		PUB	

Table 5 Microprocessor Interface in Asynchronous Motorola Mode (cont'd)

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
D10	MPAD6	I		PUB	Microprocessor Address, cont'd
B9	MPAD7	I		PUB	
C10	MPAD8	I		PUB	
A9	MPAD9	I		PUB	
B8	MPAD10	I		PUB	
A8	MPAD11	I		PUB	
B7	MPAD12	I		PUB	
D8	MPAD13	I		PUB	
A7	MPAD14	I		PUB	
C8	MPAD15	I		PUB	
B6	MPAD16	I		PUB	
D7	MPAD17	I		PUB	
A6	MPAD18	I		PUB	
B5	MPAD19	I		PUB	
A5	$\overline{\text{MPRW}}$	I		PUB	Microprocessor Read $\overline{\text{Write}}$
C6	$\overline{\text{MPDS}}$	I		PUB	Microprocessor Data Strobe
B4	$\overline{\text{MPDTACK}}$	O, open drain	D		Microprocessor Data Acknowledge
D5	$\overline{\text{MPBLE}}$	I		PUB	16 bit mode: Microprocessor Byte Low Enable
	MPTSIZ0				32 bit mode: Microprocessor Transfer Size 0
A4	MPTSIZ1	I		PUB	16 bit mode: Unused 32 bit mode: Microprocessor Transfer Size 1
C5	$\overline{\text{MPCS}}$	I		PUB	Microprocessor Chip Select

Table 5 Microprocessor Interface in Asynchronous Motorola Mode (cont'd)

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
B3	MPIM0	I		PDB	Microprocessor Interface Mode **MPIM[1:0] = 11**
C4	MPIM1	I		PDB	
A3	$\overline{\text{MPINT}}$	O, open drain	D		Microprocessor Interrupt

Table 6 Microprocessor Interface in Synchronous Motorola Mode

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
D22	MPCLK	I			Microprocessor Clock
A22	MPD0	I/O	D	PUB	Microprocessor Data
B21	MPD1	I/O	D	PUB	
D20	MPD2	I/O	D	PUB	
C21	MPD3	I/O	D	PUB	
A21	MPD4	I/O	D	PUB	
B20	MPD5	I/O	D	PUB	
A20	MPD6	I/O	D	PUB	
C20	MPD7	I/O	D	PUB	
D18	MPD8	I/O	D	PUB	
A19	MPD9	I/O	D	PUB	
C19	MPD10	I/O	D	PUB	
B18	MPD11	I/O	D	PUB	
A18	MPD12	I/O	D	PUB	
B17	MPD13	I/O	D	PUB	
C18	MPD14	I/O	D	PUB	
A17	MPD15	I/O	D	PUB	
D17	MPD16	I/O	D	PUB	
B16	MPD17	I/O	D	PUB	
A16	MPD18	I/O	D	PUB	
B15	MPD19	I/O	D	PUB	

Table 6 Microprocessor Interface in Synchronous Motorola Mode (cont'd)

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
A15	MPD20	I/O	D	PUB	Microprocessor Data, cont'd
C16	MPD21	I/O	D	PUB	
B14	MPD22	I/O	D	PUB	
D15	MPD23	I/O	D	PUB	
A14	MPD24	I/O	D	PUB	
C15	MPD25	I/O	D	PUB	
B13	MPD26	I/O	D	PUB	
D13	MPD27	I/O	D	PUB	
C14	MPD28	I/O	D	PUB	
B12	MPD29	I/O	D	PUB	
C13	MPD30	I/O	D	PUB	
A12	MPD31	I/O	D	PUB	
B11	MPAD0	I		PUB	Microprocessor Address
C12	MPAD1	I		PUB	
A11	MPAD2	I		PUB	
D12	MPAD3	I		PUB	
B10	MPAD4	I		PUB	
C11	MPAD5	I		PUB	
D10	MPAD6	I		PUB	
B9	MPAD7	I		PUB	
C10	MPAD8	I		PUB	
A9	MPAD9	I		PUB	
B8	MPAD10	I		PUB	
A8	MPAD11	I		PUB	
B7	MPAD12	I		PUB	
D8	MPAD13	I		PUB	
A7	MPAD14	I		PUB	
C8	MPAD15	I		PUB	
B6	MPAD16	I		PUB	
D7	MPAD17	I		PUB	

Table 6 Microprocessor Interface in Synchronous Motorola Mode (cont'd)

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
A6	MPAD18	I		PUB	Microprocessor Address, cont'd
B5	MPAD19	I		PUB	
A5	$\overline{\text{MPRW}}$	I		PUB	Microprocessor Read $\overline{\text{Write}}$
C6	$\overline{\text{MPTS}}$	I		PUB	Microprocessor Transfer Start
B4	MPTA	O, open drain	D		Microprocessor Transfer Acknowledge
D5	$\overline{\text{MPBLE}}$	I		PUB	16 bit mode: Microprocessor Byte Low Enable
	MPTSIZ0				32 bit mode: Microprocessor Transfer Size 0
A4	MPTSIZ1	I		PUB	16 bit mode: Unused 32 bit mode: Microprocessor Transfer Size 1
C5	$\overline{\text{MPCS}}$	I		PUB	Microprocessor Chip Select
B3	MPIM0	I		PDB	Microprocessor Interface Mode **MPIM[1:0] = 01**
C4	MPIM1	I		PDB	
A3	$\overline{\text{MPINT}}$	O, open drain	D		Microprocessor Interrupt

2.2.4 Control Interface

Table 7 Control Interface

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
AC24	$\overline{\text{POR}}$	I		PUB	Power-On Reset
AC26	$\overline{\text{HRST}}$	I/O, open drain	D	PUB	Host Reset only for use of manufacturer. Leave ball open.
AB24	CFG0	I			Configuration for boot mode, connect external pull-up resistor to CFG0, connect external pull-down resistor to CFG1 and CFG2.
AB26	CFG1	I			
AA25	CFG2	I			
Y23	$\overline{\text{NMI}}$	I		PUB	Non-Maskable Interrupt. Connect external pull-up resistor.
Y26	FWR	O	D		FW release number output

2.2.5 UART Interface

Table 8 UART Interface

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
AB25	ATXD	O	D		Asynchronous Transmit Data
AB23	ARXD	I		PUB	Asynchronous Receive Data

2.2.6 Serial DSP Interface

Table 9 Serial DSP Interface

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
U23	DSPFRAME	O	C		DSP interface functionality is not available in this firmware version. Leave balls open.
T25	DSPADDR	I/O	C	PDB	
U24	DSPCLK	O	C		
T26	DSPDATA	I/O	C	PUB	

2.2.7 Clock Interface

Table 10 Clock Interface

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
M1	NRC	I			Network Reference Clock, 19.44 MHz needed for SRTS CR.
AB3	CREF	O	D		49.408 / 65.536 MHz (basis for SRTS/ACM) output.
AF12	XTAL1	I			Crystal or clock input. Mode decision by pin INTBUS0.
AE12	XTAL2	O			Crystal
AF13	TSTCLK1	I		PDB	Test signal. Leave balls open.
AD12	TSTCLK2	I		PDB	Test signal. Leave balls open.
AE14	TSTGATE	I		PDB	Test signal. Leave balls open.
AC14	CLKOUT	O	D		120 MHz output.

2.2.8 JTAG Interface

Table 11 JTAG Interface

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
AF22	TDO	O	D		Test Data Output
AD21	TDI	I		PUB	Test Data Input
AE23	TCK	I		PUB	Test Clock
AC22	TMS	I		PUB	Test Mode Select
AF23	$\overline{\text{TRST}}$	I		PDB	Test Reset

2.2.9 Tracing Interface

Table 12 Tracing Interface

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
P24	FREEZEO	O	D		Freeze Output (indication)
M25	FREEZEI	I		PDB	Freeze Input (command)

2.2.10 OCDS Interface

Table 13 OCDS Interface

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
AA24	$\overline{\text{OCDS\_E}}$	I		PUB	OCDS enable
AA26	$\overline{\text{BRK\_IN}}$	I		PUB	Break command
Y25	$\overline{\text{BRK\_OUT}}$	O	D	PUB	Break indication
Y24	BRKPT0	O	D	PUB	Breakpoint vector
W25	BRKPT1	O	D	PUB	
V23	BRKPT2	O	D	PUB	
V25	STATUS0	O	D	PUB	Status vector
V26	STATUS1	O	D	PUB	
U25	STATUS2	O	D	PUB	
V24	STATUS3	O	D	PUB	
U26	STATUS4	O	D	PUB	
R26	IND_PC0	O	D	PUB	PC vector
T24	IND_PC1	O	D	PUB	
P25	IND_PC2	O	D	PUB	
R23	IND_PC3	O	D	PUB	
P26	IND_PC4	O	D	PUB	
R24	IND_PC5	O	D	PUB	
N25	IND_PC6	O	D	PUB	
N23	IND_PC7	O	D	PUB	

2.2.11 Test Interface

Table 14 Test Interface

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
AD2	SCANMODE	I			Test pin. Connect to VSS!
AC3	SCAN_E	I		PDB	Test pin. Connect to VSS!

Table 14 Test Interface

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
AD22	INTBUS0	I/O	D	PDB	Test Bus and XTAL1 Clock Mode when input during reset. Connect external pull-up resistor for external clock mode and pull-down for crystal mode.
AE24	INTBUS1	I/O	D	PDB	Connect external pull-up resistor.
AD23	INTBUS2	I/O	D		Leave open.
AF24	INTBUS3	I/O	D		
AD25	INTBUS4	I/O	D		
AD26	INTBUS5	I/O	D		
AC25	INTBUS6	I/O	D	PDB	

Table 14 Test Interface

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
N24	DBGBUS0	I/O	D	PDB	Test bus. Leave open.
M26	DBGBUS1	I/O	D		
L25	DBGBUS2	I/O	D	PDB	
M24	DBGBUS3	I/O	D	PDB	
L26	DBGBUS4	I/O	D	PDB	
M23	DBGBUS5	I/O	D	PDB	
K25	DBGBUS6	I/O	D	PDB	
L24	DBGBUS7	I/O	D	PDB	
K26	DBGBUS8	I/O	D	PDB	
K23	DBGBUS9	I/O	D		
K24	DBGBUS10	I/O	D	PDB	
J26	DBGBUS11	I/O	D	PDB	
H25	DBGBUS12	I/O	D	PDB	
H26	DBGBUS13	I/O	D	PDB	
J24	DBGBUS14	I/O	D	PDB	
G25	DBGBUS15	I/O	D	PDB	
H23	DBGBUS16	I/O	D	PDB	
H24	DBGBUS17	I/O	D	PDB	
F25	DBGBUS18	I/O	D	PDB	
G23	DBGBUS19	I/O	D	PDB	
F26	DBGBUS20	I/O	D	PDB	
G24	DBGBUS21	I/O	D	PDB	
E25	DBGBUS22	I/O	D	PDB	
E26	DBGBUS23	I/O	D	PDB	
D25	DBGBUS24	I/O	D	PDB	

Table 14 Test Interface

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
E23	DBGBUS25	I/O	D	PDB	Cont'd.
D26	DBGBUS26	I/O	D	PDB	
E24	DBGBUS27	I/O	D	PDB	
C25	DBGBUS28	I/O	D	PDB	
D24	DBGBUS29	I/O	D	PDB	
C26	DBGBUS30	I/O	D	PDB	
B24	DBGBUS31	I/O	D	PDB	

2.2.12 Unused Balls

Table 15 Unused Balls

Ball No.	Name	In/Out	Driver	Pullup / -down	Function
A24	N.C.				Not connected. Leave open.
B23	N.C.				
C23	N.C.				
A23	N.C.				
B22	N.C.				

2.2.13 Power Supply

Table 16 Power Supply

Ball No.	Name	Function
A1, A2, A26, B2, B25, B26, C3, C24, D4, D9, D14, D19, D23, H4, J23, L11, L12, L13, L14, L15, L16, M11, M12, M13, M14, M15, M16, N4, N11, N12, N13, N14, N15, N16, P11, P12, P13, P14, P15, P16, P23, R11, R12, R13, R14, R15, R16, T11, T12, T13, T14, T15, T16, V4, W23, AC4, AC8, AC13, AC18, AC23, AD3, AD24, AE1, AE2, AE25, AF1, AF25, AF26	VSS	Common ground rail
D6, D11, D16, D21, F4, F23, L4, L23, T4, T23, AA4, AA23, AC6, AC11, AC16, AC21	VDDP	I/O power supply, 3.3 V
B1, F1, K2, M4, R3, W1, AB1, AD1, AE5, AF6, AD8, AD10, AE13, R25, N26, J25, G26, F24, A25, C22, B19, C17, A13, A10, C9, C7	VDD	Core power supply, 1.8 V
AD11	VDDPLL	Analog PLL power supply, 1.8 V
AD14, AC17, AD19, W24	VDDT	Embedded controller power supply, 1.8 V
AC12, AE26, W26	VDDM	Embedded Controller main memory power supply, 1.8 V

3 Functional Description

From a functional view IWORX can be seen as a system partitioned into several subsystems (= functional blocks). Data traffic can enter the chip at the UTOPIA interface or at the PDH interface.

3.1 Functional Blocks and Data Flows

IWORX V1.1 supports the below listed functional blocks with the first firmware release as described in chapter Operational Description.

- ATM Cell Mapping and Demapping on full and fractional line rate
- AAL2 Multiplexing and Demultiplexing
- AAL1 Segmentation and Reassembly Inclusive Clock Recovery
- Address Reduction, Header Translation and External Routing Tag Addition
- Traffic Management TM
- Operation, Administration, and Maintenance OAM

The following data flow diagrams describe the main data paths within the IWORX. All listed data flows can work in parallel, simultaneously.

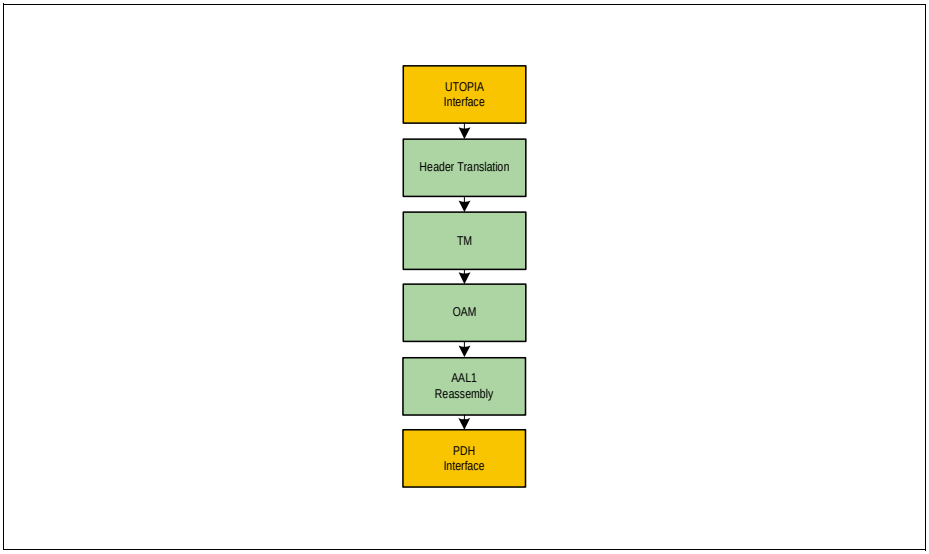


Figure 12 AAL1 Flow from UTOPIA to PDH Interface

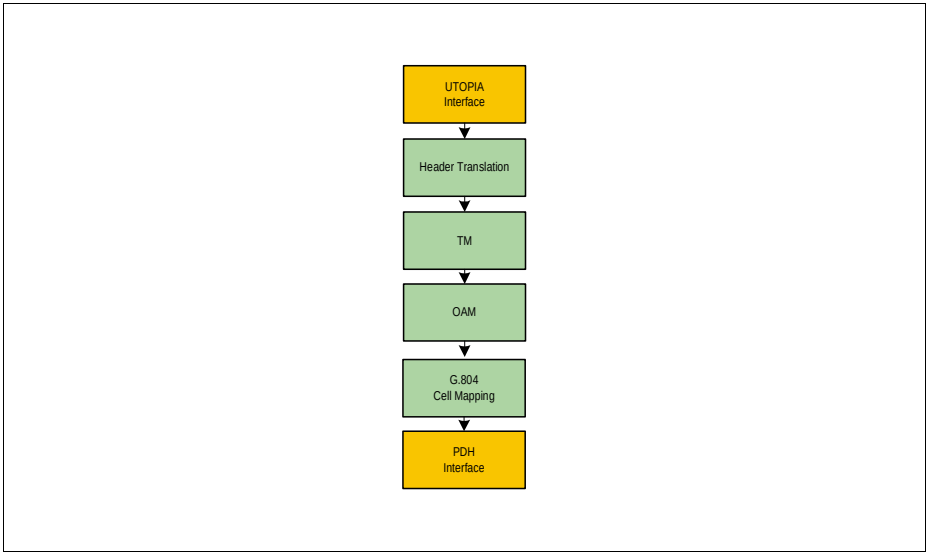


Figure 13 Pure G.804 Flow from UTOPIA to PDH Interface

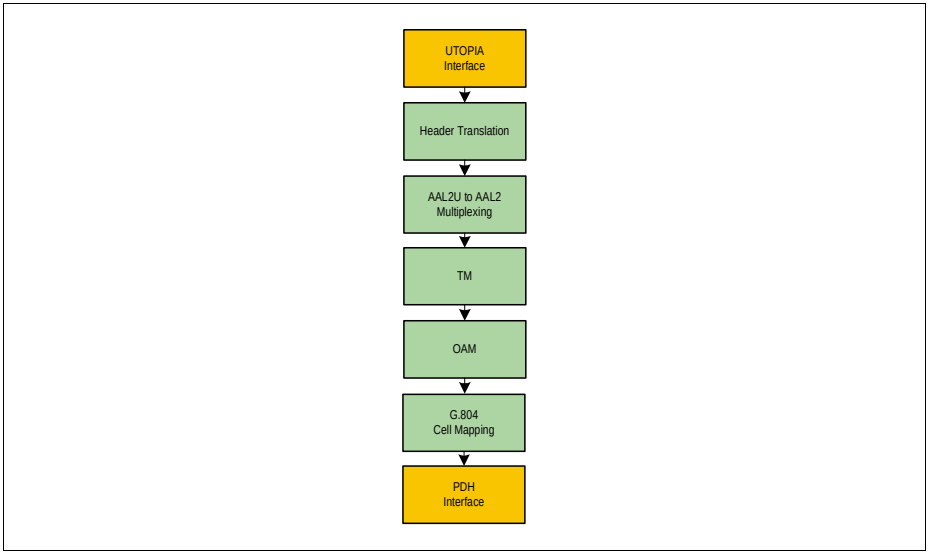


Figure 14 AAL2 Flow from UTOPIA to PDH Interface

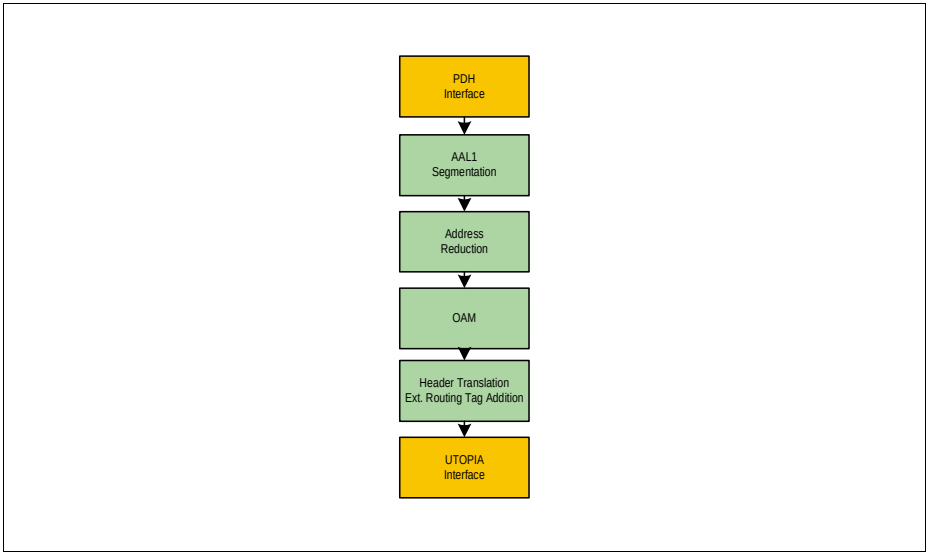


Figure 15 AAL1 Flow from PDH to UTOPIA Interface

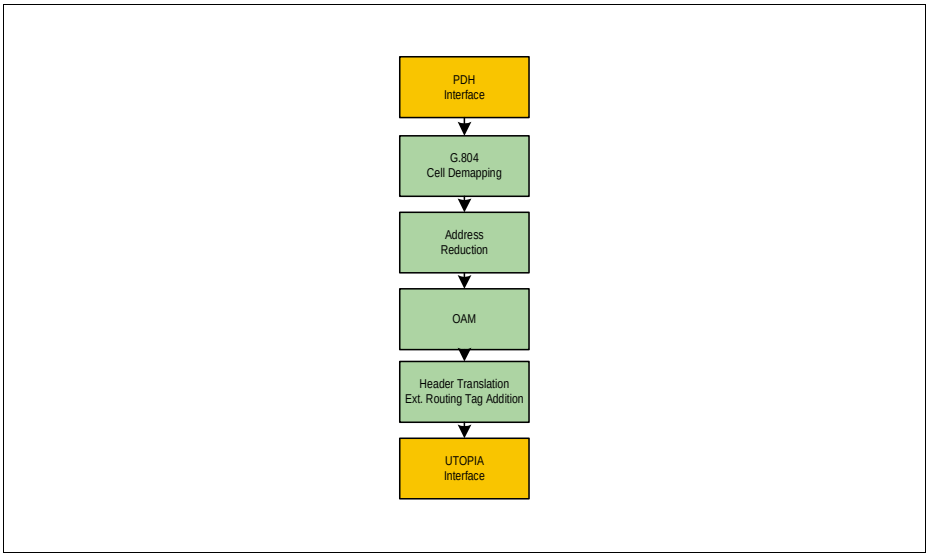


Figure 16 Pure G.804 Flow from PDH to UTOPIA Interface

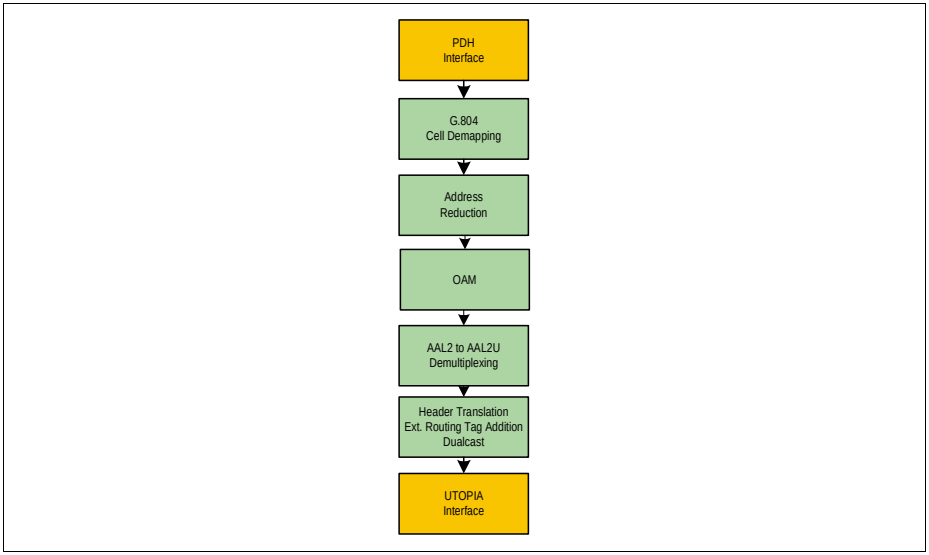


Figure 17 AAL2 Flow from PDH to UTOPIA Interface

3.1.1 Additional Flows for Debug Purpose

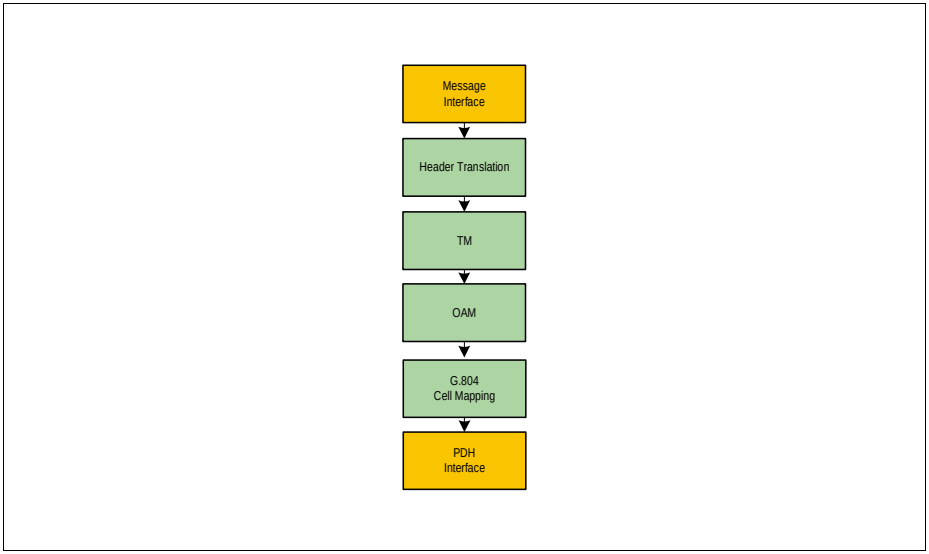


Figure 18 Cell Insertion in Transmit Direction without AAL2 Multiplexing

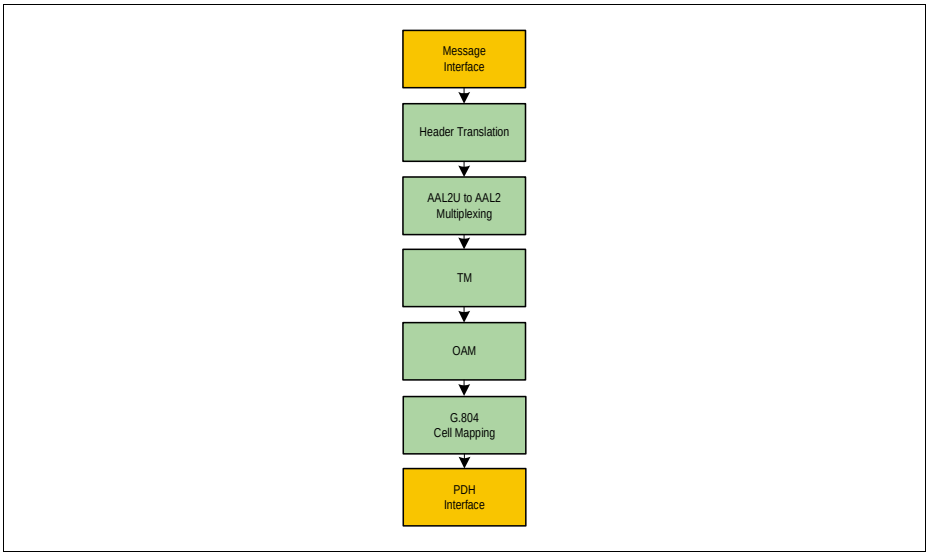


Figure 19 AAL2U Cell Insertion in Transmit Direction with AAL2 Multiplexing

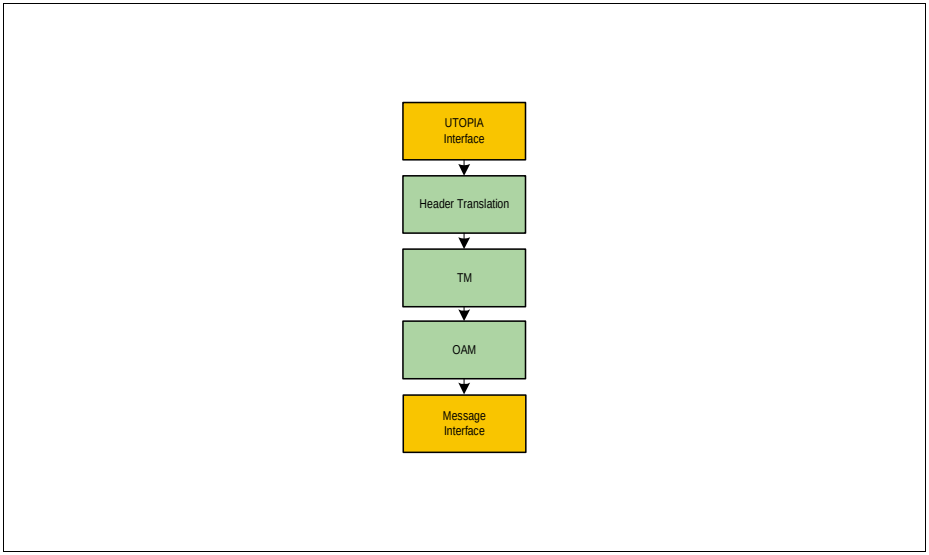


Figure 20 Cell Extraction from Transmit Direction

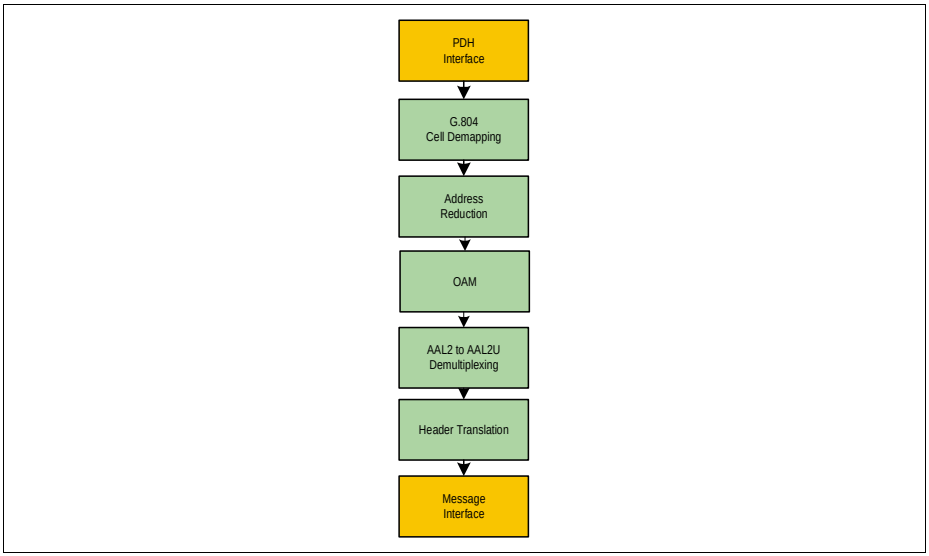


Figure 21 AAL2U Cell Extraction from Receive Direction

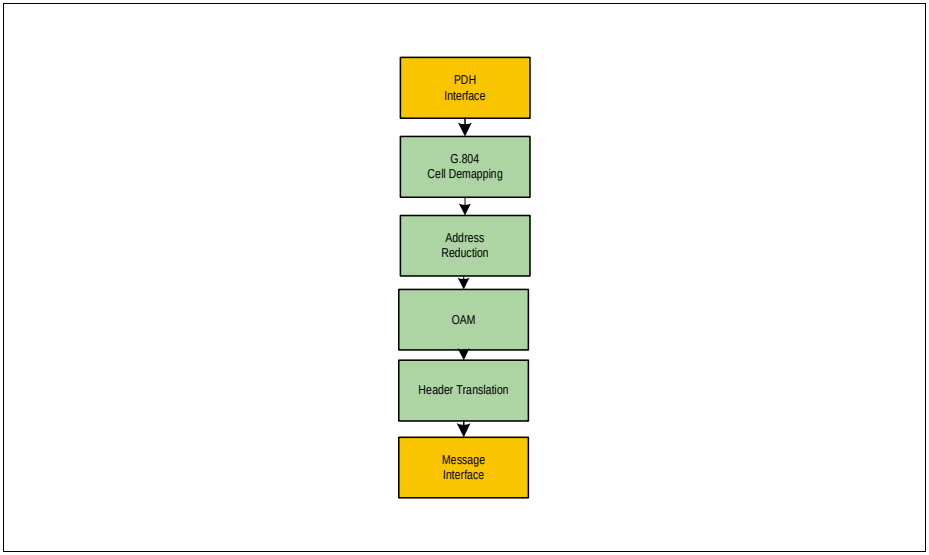


Figure 22 Cell Extraction from Receive Direction without AAL2 Demultiplexing

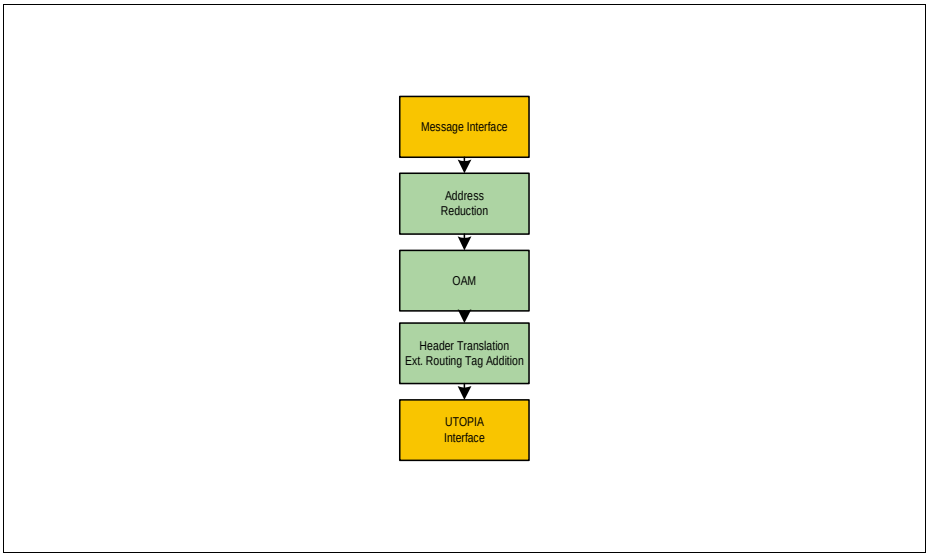


Figure 23 Cell Insertion in Receive Direction without AAL2 Demultiplexing

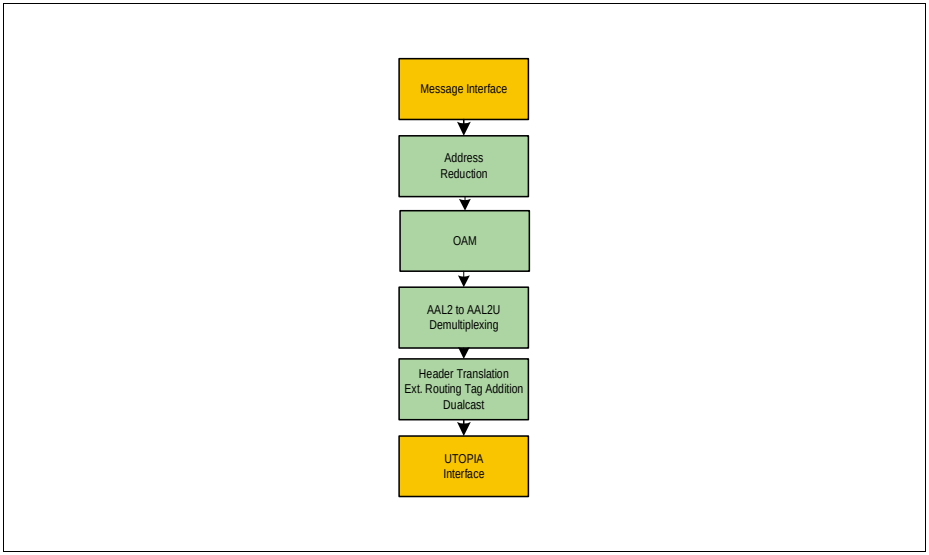


Figure 24 AAL2 Cell Insertion in Receive Direction with AAL2 Demultiplexing

CONFIDENTIAL

3.2 ATM Cell Mapping and Demapping

ATM cell mapping and demapping can be used at E1/T1 data rates according to ITU-T G.804 but also at fractional E1/T1 data rates according to AF-PHY-0130.00. It can be defined up to 32 fractions within an E1 frame with a bandwidth of N x 64Kbit/s (N = 1..31) by setting up a logical channel where timeslots of the frame are assigned to a logical channel. For T1, up to 24 fractions within the frame can be defined with a bandwidth of N x 64Kbit/s (N = 1..24). It exists no restriction regarding the timeslot assignment within a PDH interface port accept that a timeslot can't be assigned to more than one logical channel. The assignment can be done direction-independent.

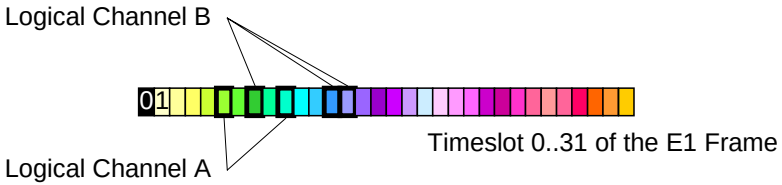


Figure 25 Example for Cell Mapping on 2 E1 Fractions

The figure shows an example where two logical channels have been defined. Logical channel A uses timeslots 5, 9 and operates as a virtual PHY with 2 x 64Kbit/s, logical channel B uses timeslots 7, 12, 13 and operates as a virtual PHY with 3 x 64Kbit/s. The unused timeslots can carry voice data for AAL1.

3.2.1 ATM Cell Mapping Functions

- Cell discarding
- Write ATM cells except of 5th octet to ATM Transmit Buffer of logical channel
- Hold cells for N x 64Kbit/s logical channel in ATM Transmit Buffer (max. 8 x N cells)
- Reading octets from ATM Transmit Buffer
- Cell rate de-coupling: idle/unassigned cell insertion
- Cell payload scrambling
- HEC generation

3.2.1.1 Cell Discarding

The discarding of cells with CLP = 1 (Bit 0 of the 4th ATM header octet) can be enabled and depends on the filling level of the logical channel assigned ATM Transmit Buffer.

Available cell discard options are

- Discarding if CLP = 1 and ATM Transmit Buffer is more than ¾ stuffed
- Discarding if CLP = 1 and ATM Transmit Buffer is more than ½ stuffed

- Discarding if CLP = 1 and ATM Transmit Buffer is more than ¼ stuffed
- Discarding if CLP = 1, independent from ATM Transmit Buffer level

3.2.1.2 Cell Rate Decoupling: Idle/Unassigned Cell Insertion

When the ATM Transmit Buffer of a logical channel is empty, idle or unassigned cells are transmitted to provide cell rate de-coupling.

Idle cells are transmitted as defined in the ITU-T I.361. Unassigned cells can be inserted, as defined in the B-ISDN UNI and NNI physical layer generic criteria Bellcore TR-NWT-001112.

The 4 MSBs of header octet 1 and the 4 LSBs of header octet 4 are programmable for / unassigned cells. All other header bits will be 0.

octet 1	GFC[3:0]/VPI[11:8] = programmable	VPI[7:4] = 0000_B	
octet 2	VPI[3:0] = 0000_B	VCI[15:12] = 0000_B	
octet 3	VCI[11:4] = 0000_0000_B		
octet 4	VCI[3:0] = 0000_B	PTI[2:0] programmable	CLP programmable
octet 5	HEC		
octet 6	0110_1010_B		
.	.		
octet 53	0110_1010_B		

- If idle cell insertion according to ITU-T I.361 or ITU-T I.432.1 is selected, the 4 LSBs of header octet 4 are set to 0001_B and the 4 MSBs of header octet 1 are set to 0000_B.
- If unassigned cell insertion at the NNI or uncontrolled UNI according to ITU-T I.361 is desired, the 4 LSBs of header octet 4 should be set to XXX0 and the 4 MSBs of header octet 1should be set to XXXX. For X any value is allowed.

The payload of idle or unassigned cells consists of the same octet which is repeated 48 times. For ITU-T I.432.1 compliant idle cells, the payload octet is set to 0110_1010_B. The pre-assigned values of the information field of all unassigned cells are for further study (ITU-T I.361) and the payload octet can be set by the user.

3.2.1.3 Cell Payload Scrambling

ITU-T I.432.3 recommends the self-synchronizing scrambler $x^{43}+1$ for payload scrambling at E1 datarates. For T1 scrambling is not recommended according to ITU-T I.432.3 but can be applied.

The scrambler function is implemented in the device. It can be disabled per logical channel.

3.2.1.4 HEC Generation

The HEC generation is implemented according to ITU-T I.432.1 using the generator polynomial $x^8 + x^2 + x + 1$. To significantly improve the cell delineation performance in the case of bit-slips it is recommended that

- the check bits are added (modulo 2) to an 8-bit pattern (coset) before being inserted in the last octet of the header.
- the recommended pattern is “0101 0101" and automatically used by the transmitter.
- the receiver must subtract (equal to add modulo 2) the same pattern from the 8 HEC bits before calculating the syndrome of the header.

As an example, if the first 4 octets of the header were all zeros the generated header before scrambling would be “00000000_00000000_00000000_00000000_01010101”. The starting value for the polynomial check is 0s (binary)

3.2.2 ATM Cell Demapping Functions

- Cell delineation
- HEC check: Header error detection and correction
- Cell payload descrambling
- Idle or Unassigned Cell Deletion
- Statistics counter event generation (see appendix of the Programmer’s Reference Manual)
- Write cells except of UDF octet to ATM Receive Buffer (max. 16 cells per port)
- Read cells from ATM Receive Buffer and forward to next functional block

3.2.2.1 Cell Delineation

The cell delineation algorithm is implemented according to the ITU-T Recommendation I.432.1.

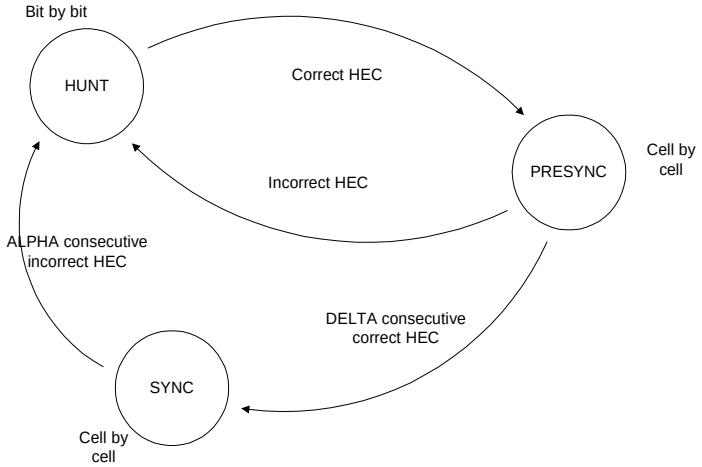
Detection of “Out of Cell Delineation” (OCD) anomalies and “Loss of Cell Delineation” (LCD) defect is supported whenever the SYNC state is left or entered. The generation of interrupts due to LCD defects can be enabled. It is also possible to see the current state of the cell delineation FSM (Finite State Machine) on demand.

As octet boundaries are available within the receive physical layer prior to cell delineation, the cell delineation process is performed octet by octet in the HUNT state. As long as the cell delineation is not in the SYNC state, received octets are discarded.

The ALPHA and DELTA parameters influence the robustness of the algorithm against false misalignment due to bit errors (ALPHA) and false delineation in the re

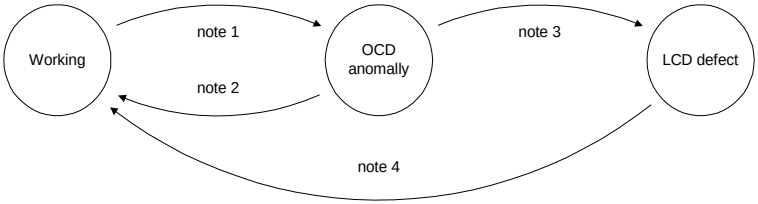
synchronization process (DELTA) The used values are common for all logical channels according to the ITU-T I.432.1 recommendation:

- for the Frame-based Physical Layer, ALPHA = 7 and DELTA = 6.



Note - The "correct HEC" means the header has no bit error (syndrome is zero) and has not been corrected

Figure 26 Cell Delineation State Diagram (Figure 5/I.432.1)



note1 Triggered by state transition (Case A) due to alpha consecutive incorrect HEC´s in the cell delineation process (Fig. 5 of ITU-T Recommendation I.432.1)

note2 Triggered by state transition (Case B) due to delta consecutive correct HEC´s in thecell delineation process (Fig. 5 of ITU-T Recommendation I.432.1)

note3 Triggered by 50 continuous ms in the OCD anomaly maintenance state

note4 Triggered by 50 continuous ms in the cell delineation "Sync" state (Fig.5 of ITU-T Recommendation I.432.1)"

Figure 27 Maintenance State Transition Diagram for Cell Delineation Events (Figure 2/ I.432.3)

The Loss of Cell Delineation (LCD) state is entered whenever the Out of Cell (OCD) state is continuously active for more than 50 ms (as recommended by ITU-T I.432.1).

For each logical channel a separate timer is available. The global preload value is 50 ms. After expiration of each timer, an “lcd_start” event is generated.

The timer is started at the transition from SYNC to OCD-state. After expiration LCD state is entered. Whenever the SYNC state is entered before the timer expires, the timer is reset.

The transition from LCD to Working state follows the same procedure. If after the LCD state the SYNC state is entered again, the timer is started and after expiration the maintenance state machine is in working state again. In parallel an “lcd_end” event is generated. If synchronization is lost again during the timer period, LCD state is reentered and the timer is reset.

3.2.2.2 HEC Check: Header Error Detection and Correction

The Header Error Control (HEC) is implemented according to the ITU-T I.432.1 B-ISDN user-network interface - Physical layer specification.

According to the HEC algorithm, cells are discarded when a multi-bit header error is detected in the Correction mode or a header error is detected in the Detection mode.

According to the HEC algorithm, cells are corrected when a single-bit error is detected in the Correction mode.

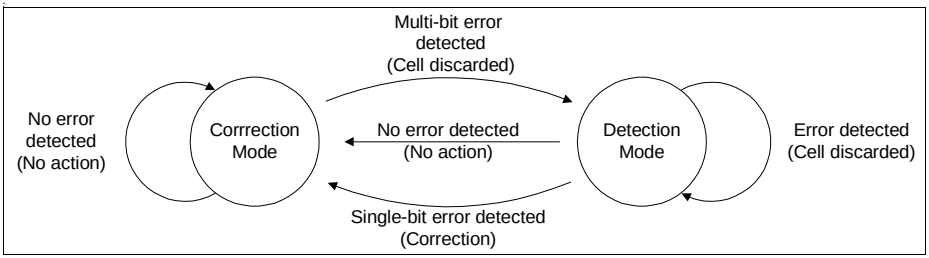


Figure 28 HEC: Receiver mode of Operation (Figure 3/ITU I.432.1)

The pure HEC detection mode as recommended by the ATM Forum can be selected instead of HEC handling according to ITU I.432.1.

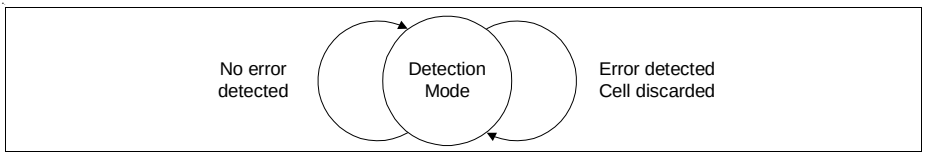


Figure 29 HEC Detection According to ATM Forum

CONFIDENTIAL

No discarding of HEC errored cells as an option is available - internally but not at the message interface - for future upgrades with IMA.

3.2.2.3 Cell Payload Descrambling

ITU-T I.432.3 recommends the self-synchronizing scrambler $x^{43}+1$ for payload scrambling at E1 datarates. For T1 no scrambling is recommended.

The self-synchronizing scrambler function is implemented in the device. It can be disabled per logical channel.

3.2.2.4 Idle, Physical Layer or Unassigned Cell Deletion

According to ITU-T I.361, idle cells, physical layer OAM cells and cells reserved for use by the physical layer are not passed to the ATM layer at the UNI.

	Octet 1	**Octet 2**	**Octet 3**	**Octet 4**
Idle cell identification (Notes 1 and 2)	0000/0000	0000/0000	0000/0000	0000/0001
Physical OAM cell identification (Note 2) layer	0000/0000	0000/0000	0000/0000	0000/1001
Reserved for use of the physical layer (Notes 1, 2 and 3)	PPPP/0000	0000/0000	0000/0000	0000/PPP1

P: Indicates the bit is available for use by the physical layer
Values assigned to these have no meaning with respect to the fields occupying the corresponding bit positions at the ATM layer

Notes:

1 In the case of physical layer cells, the bit in the location of the CLP indication is not used for the CLP mechanism as specified in 3.4.2.3.2/I.150.

2 Cells having header values which are identified as idle, physical layer OAM, and reserved for use by the physical layer are not passed to the ATM layer from the physical layer.

3 Specific pre-assigned physical layer cell header values are given in Recommendation I.432

Figure 30 Pre-assigned Cell Header Values at the UNI for Use by the Physical Layer (Excluding the HEC Field) (Table 1/I.361)

In contrast to this the ATM-Forum recommends in the User-network interface specification that the receiving ATM entity is responsible for extraction and discarding of unassigned and idle cells.

Use	Octet 1	Octet 2	Octet 3	Octet 4
invalid	XXXX/0000	0000/0000	0000/0000	0000/XXX1
unassigned	0000/0000	0000/0000	0000/0000	0000/XXX0
X: Indicates “don’t care” bits				

Figure 31 Pre-defined Header Field Values by ATMF UNI V3.1

The implemented cell filter can be used in order to achieve ITU-T or ATM-Forum compliance.

The 4 MSBs of header octet 1 and the 4 LSBs of header octet 4 of the received cells to be discarded are programmable in the cell filter. It is possible to mask all or some of the 8 bits by the cell filter mask. The masked bits are considered as “don’t care”. All other header bits must be 0.

- If ITU-T I.361 compliance is desired, the cell filter for the 4 MSBs of octet 1 should be set to 0000 and for the 4 LSBs of octet 4 should be set to 0001.
  If only idle cells shall be deleted, the cell filter mask for the 4 MSBs of octet 1 should be set to 0000 and for the 4 LSBs of octet 4 should be set to 0000.
  If all physical layer cells should be deleted, the cell filter mask for the 4 MSBs of octet 1 should be set to 1111 and for the 4 LSBs of octet 4 should be set to 1110.
- For ATM Forum compliance, the cell filter for the 4 MSBs of octet 1 should be set to 0000 and for the 4 LSBs of octet 4 should be set to 0000.
  The cell filter mask for the 4 MSBs of octet 1 should be set to 1111 and for the 4 LSBs of octet 4 should be set to 1110.
  This configuration will delete all unassigned cells.

The deletion of idle, physical layer or unassigned cells is always active during logical channel usage.

3.3 AAL2 Multiplexing and Demultiplexing

ITU-T recommendation I.363.2 specifies ATM Adaptation Layer (AAL) type 2. The AAL2 provides for the bandwidth efficient transmission of low-rate, short and variable length packets in delay sensitive applications. More than one AAL type 2 user information stream can be supported on a single ATM connection.

Functional Feature List

- **Standard:**
  AAL2-CPS is supported in line with the standard ITU-T I.363-2. It is assumed that the CPS packet payload (CPS-SDU) does not exceed 45 Bytes. AAL2 SSCS is not supported by the first FW release.
- **Number of Connections:**
  Up to 16 VCs are supported with multiplexing/demultiplexing of up to 249 AAL2 connections per VC
- **Dualcast:**
  The dualcast function in RX direction which is required to support soft handover is implemented for all 4k AAL2 channels.
- **Throughput:**
  Support of cell processing within 24 core clock cycles on ingress/egress from/to SSCS interface with 60 MHz core clock
- **Switching:**
  Switching of AAL2 VCs in RX direction (demultiplexing) is supported by demultiplexing each AAL2 VC to an ATM/AAL2U VC and subsequent translation to independent egress AAL2U VCs.
- **Timer_CU:**
  Support of one Timer_CU per VC with a range of 0.1 to 10 ms, and with a resolution of 0.1 ms
- **Layer specific OAM:**
  Statistics counter are implemented for two configurable AAL2 VCs and over all AAL2 VCs, which allows maintenance of a (non-standardized) AAL2 MIB (see appendix of the Programmer's Reference Manual).

CONFIDENTIAL

AAL2 is divided into the Common Part Sublayer (CPS) and the Service Specific Convergence Sublayer (SSCS). **Figure 32** refers to the CPS only. It shows how CPS packets of 4 … 48 Bytes length, carrying in the CPS info field the user related data, are multiplexed in the CPS-PDU Payloads – thus creating a channel substructure within the respective ATM connections. Comparable to other adaptation layers, field pointers, sequence numbers and error correction are supported.

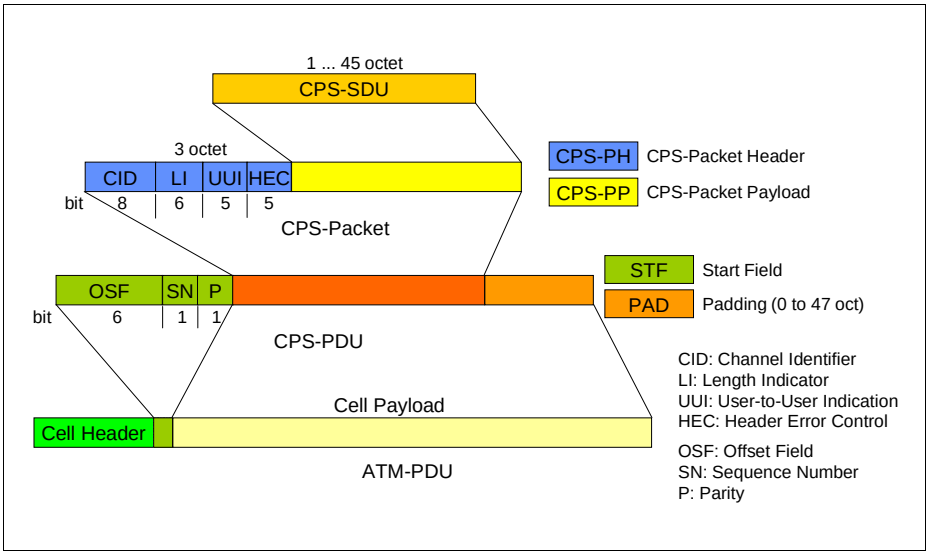


Figure 32 AAL2 Common Part Sublayer

The transmission of mini-cells on the air-link of mobile and fixed wireless access networks is the standard choice combined with the use of sophisticated voice compression algorithms. The currently non-standardized but well specified AAL2U proposal as shown in **Figure 33** addresses how to convey the payload of just one AAL2 channel by the use of just one ATM cell. The payload part of the AAL2U cell contains one CPS packet with a length of 4 ... 48 bytes. Unused payload is filled with padding octets coded with the value zero.

Existing implementations of switching planes or cell buses can be reused to route cells to the units capable of handling AAL2.

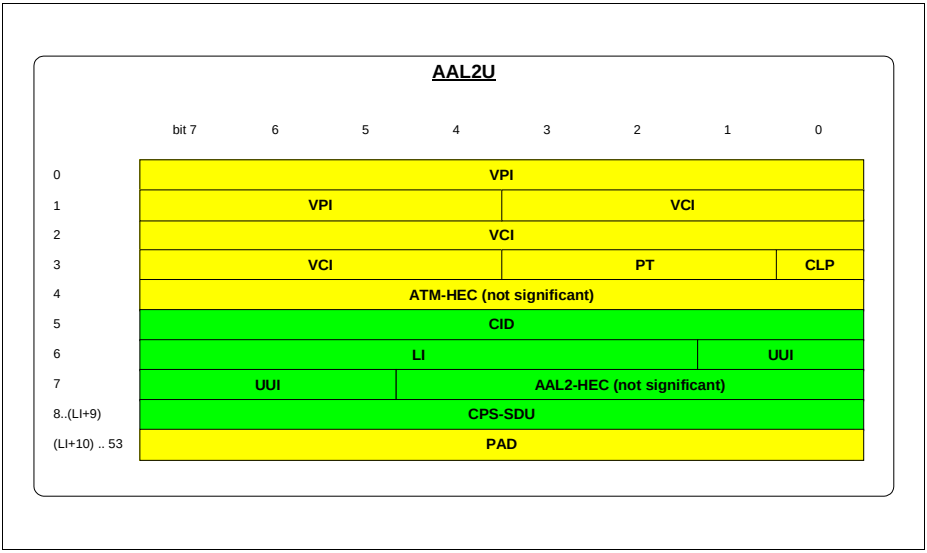


Figure 33 AAL2U Cell Format

After passing through the functional block Header Translation in TX direction, the AAL2U cell has a new user defined VPI/VCI value which represents the AAL2 multiplex sink.

3.3.1 AAL2 CPS Transmitter

The multiplexing function in the Common Part Sublayer Transmitter merges several streams of CPS-Packets onto a single ATM connection. The Common Part Sublayer receives CPS-SDUs from one or more SSCS transmitter processes via AAL2U cells. It multiplexes and packs CPS-Packets extracted from the AAL2U cells into CPS-PDUs.

Figure 34 illustrates the multiplexing function.

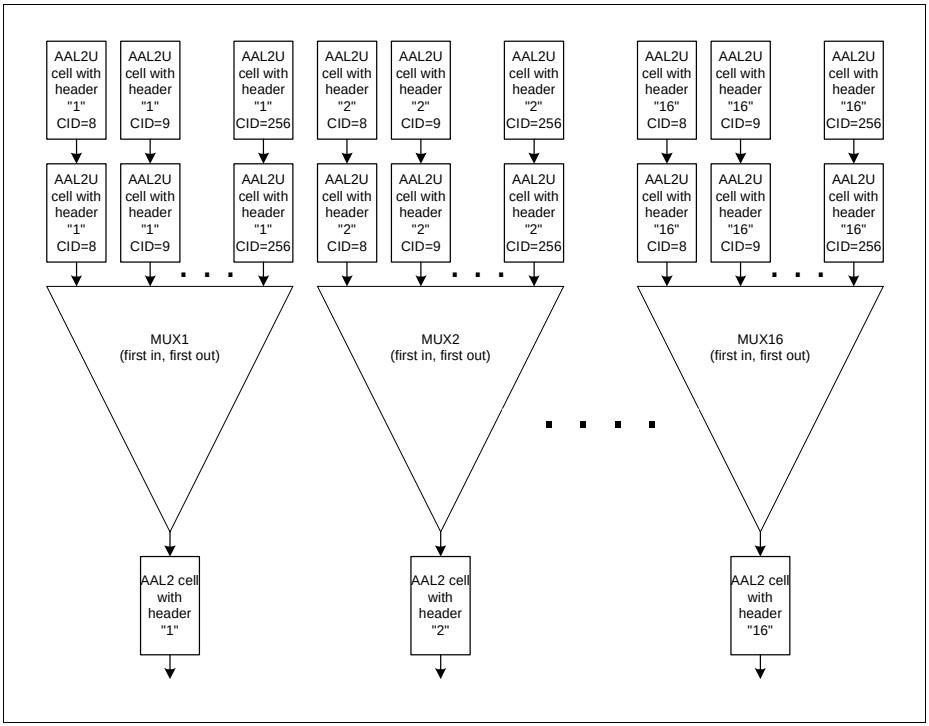


Figure 34 AAL2 Multiplexing

Up to 16 AAL2 multiplexer can be used. Each of them is able to merge up to 249 AAL2 channel streams. CID 0 and 2..7 are not supported according to ITU-T I.363.2 Table 4.

Table 4/I.363.2 – Coding of the CID Field

CID value	use
0	Not used (0 octet used for padding)
1 2 .. 7	Reserved for Layer Management peer-to-peer procedures Reserved
8 .. 255	Identification of AAL type 2 CPS user entity

Figure 35 ITU-T Table 4/I.363.2

At the ingress following checks and derived actions are executed after CPS packet extraction from the AAL2U cells as shown in **Table 17**. The treatment of AAL2U cells before the CPS transmitter is described in **Chapter 3.5**.

Table 17 AAL2U Cell Bit Field Handling at AAL2 Transmitter Ingress

Bit Field	Action
5th header octet ATM-HEC	not evaluated, don't care
AAL2-HEC	not evaluated, new generated
CID	Discard CPS packet if CID is invalid, CID = 2..7 Transparent forwarding if CID is not invalid
UUI	Discard CPS packet if UUI is invalid, UUI = 28..29 Transparent forwarding if UUI is not invalid
LI	Discard CPS packet if LI is invalid, LI > 44 Transparent forwarding if LI is not invalid
PAD	Padding octets are not evaluated, don't care

Generation of the CPS Packet HEC

The transmitter calculates the remainder of the division (modulo 2), by the generator polynomial x5 + x2 + 1, of the product of x5 and the contents of the first 19 bits of the CPS packet header. The coefficients of the remainder polynomial is inserted in the HEC field with the coefficient of the x4 term in the most significant bit of the HEC field.

CONFIDENTIAL

ATM Header of AAL2 VC

The ATM header of the AAL2 cells coming out of a multiplexer is equivalent to the ATM header of the AAL2U cells coming into this multiplexer.

Generation of the Start Field STF for the AAL2 Cells

Offset Field (OSF):

This field carries the binary value of the offset, measured in number of octets, between the end of the STF and the first start of a CPS-Packet or, in the absence of a first start, to the start of the PAD field. The value 47 indicates that there is no start boundary in the CPS-PDU payload. Values greater than 47 are not allowed.

Sequence Number (SN):

This bit is used to number (modulo 2) the stream of CPS-PDUs.

Parity (P)

This bit is used by the receiver to detect errors in the STF. The transmitter sets this bit value such that the parity over the 8-bit STF is odd.

CPS-PDU Padding

Unused payload is filled with padding octets coded with the value zero.

Timer_CU

The "combined use" Timer_CU assures that CPS-Packets with one or more octets already packed wait at most the duration of Timer_CU before being scheduled for transmission.

16 counters are maintained for sixteen AAL2 muliplexers. The resolution for each of the timer is 0.1 ms. Each timer can be configured from 0.1 to 10 ms by the user. The timer starts when a new construction of CPS-PDU starts. If the timer expires before the construction of CPS-PDU has been finished, the partially packed AAL2 cell is padded and sent to the egress. If multiple connections have Timer_CU's expired, the selection is based on a round robin sequence.

3.3.2 AAL2 CPS Receiver

At the Common Part Sublayer Receiver, the CPS-Packets are unpacked and demultiplexed and passed to one of the SSCS receivers via AAL2U cells.

Figure 36 illustrates the demultiplexing function.

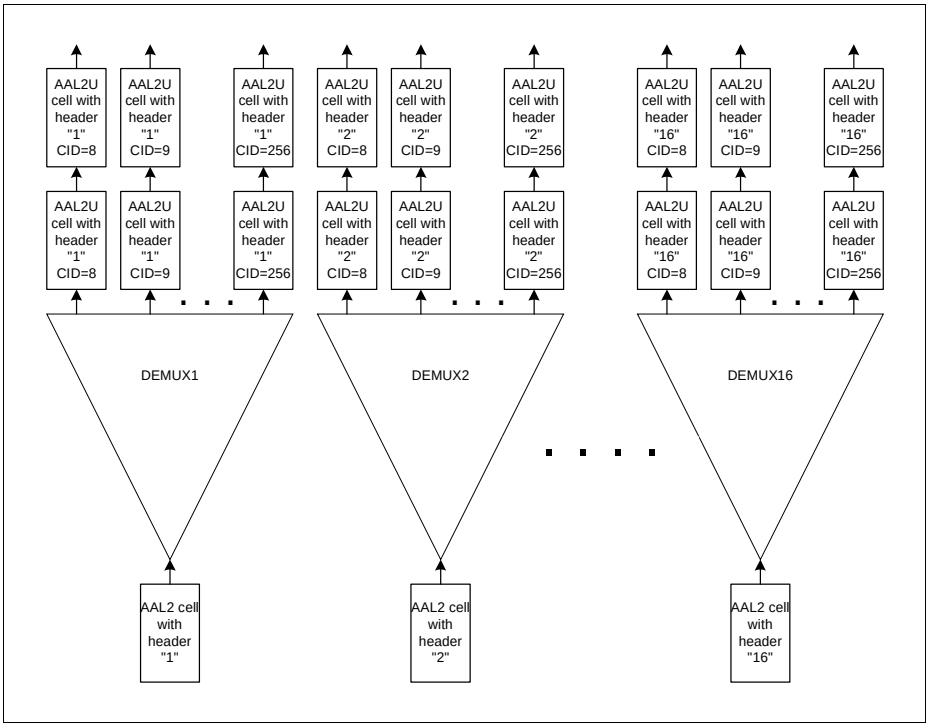


Figure 36 AAL2 Demultiplexing

Up to 16 AAL2 demultiplexer can be used. Each of them is able to split up to 249 AAL2 channel streams. CID 0 and 2..7 are not supported according to ITU-T I.363.2 Table 4. At the ingress following checks and derived actions are executed as shown in **Table 18**.

Table 18 AAL2 Cell Bit Field Handling at AAL2 Receiver Ingress

Bit Field	Action
5th header octet ATM-HEC	Discard CPS-PDU if ATM-HEC is wrong
AAL2-HEC	Discard CPS packet if AAL2-HEC is wrong

CONFIDENTIAL

Table 18 AAL2 Cell Bit Field Handling at AAL2 Receiver Ingress

Bit Field	Action
CID	Discard CPS packet if CID is invalid, CID = 2..7 Transparent forwarding if CID is not invalid
UUI	Discard CPS packet if UUI is invalid, UUI = 28..29 Transparent forwarding if UUI is not invalid
LI	Discard CPS packet if LI is invalid, LI > 44 Transparent forwarding if LI is not invalid
OSF of start field	Discard CPS-PDU if OSF is invalid, OSF > 47
SN of start field	If sequence number is not matching with expected value, partial CPS-SDU if any is discarded and CPS-PDU processing starts at the location pointed by OSF
P of start field	Discard CPS-PDU If parity is wrong
PAD	Discard AAL2 cell if padding octet is not equal zero

The ATM header of the incoming AAL2 cell is added to the extracted CPS packets for construction of AAL2U cells. Unused octets of AAL2U cells are padded with the value zero. The AAL2U cells are forwarded to the functional block Header Translation. The dual cast functionality is supported for all possible AAL2 channels (16 x 249) by the functional block Header Translation and described in the respective chapter.

3.4 AAL1 Segmentation and Reassembly

A logical channel that is configured to AAL1 mode offers ATM Forum compliant circuit emulation services according to ATMF af-vtoa-0078.000 via AAL1 as defined in ITU-T I.361.1.

Unstructured CES Mode

A 2.048 Mbit/s (E1) or 1.544 Mbit/s (T1) bitstream - full line rate of a PDH port - is packed into ATM cells without any framing. No alignment between octets in E1 or T1 frames and octets in the ATM cells is done.

For this Unstructured T1/E1 Circuit Emulation Service (CES) the ATM adaptation layer type 1(AAL1) with Unstructured Data Transfer (UDT) as defined in ITU-T I.363.1 is used. The use of partially filled cells is not supported due to the recommendation in ATMF af-vtoa-0078.000 to use only fully filled cells in case of unstructured CES.

Both clock recovery methods, the Synchronous Residual Time Stamp (SRTS) method and Adaptive Clock Method (ACM), are supported for unstructured CES as recommended by ATMF af-vtoa-0078.000.

Per PDH interface port a Segmentation Buffer with a maximum size of 16 cells and a Reassembly Buffer with a maximum size of 256 cells is implemented.

Structured CES Mode

The structured circuit emulation service is intended to carry N of the 24 (T1) or 32 (E1) timeslots across the ATM network.

An emulated Nx64 kbit/s circuit will be referred to as a logical channel throughout this document. It is possible that several logical channels share the same physical interface port.

In structured CES mode neither SRTS nor ACM clock recovery is possible. The DS1/E1 Nx64 kbit/s Service requires the use of synchronous circuit timing, as recommended by ITU-T I.363.1.

Multiframe based SDT without CAS should not be used. For correct operation in frame based SDT service without CAS, the number of used PDH interface timeslots for the channel has to be smaller or equal to the used cell filling level for the channel.

3.4.1 AAL1 Segmentation Functions

- Segmentation
- SN/SNP generation
- SDT pointer generation
- RTS value generation and insertion
- Statistics counter event generation (see appendix of the Programmer's Reference Manual)
- Write to Segmentation Buffer
- Hold data in Segmentation Buffer for building cell payload
- Read cells from Segmentation Buffer and forward to next functional block
- Padding of partially filled cells

3.4.1.1 Segmentation

The segmentation function generates the standard AAL type 1 SAR-PDU

AAL1 Type 1 SDT Structure Length

For Structured Circuit Emulation Service as defined by the ATM-Forum in "Circuit Emulation Services Version 2.0" Structured Data Transfer (SDT) is used. The structure length used for SDT in ATM cells is:

- N if frame-based SDT is selected
- N x 16 if CRC multiframe-based SDT is selected for E1 ports
- N x 24 if superframe-based SDT or extended superframe-based SDT is selected for T1 ports.

3.4.1.2 Transport of CAS Information

The four CAS bits for each time slot are transported within one multiframe from the framer to the IWORX. A signalling buffer (256 x 4 x 2bit) holds the CAS bits from the PDH interface. The activation of CAS packetization can be enabled or disabled per PDH port.

The CAS bits will be packed in a signalling substructure after the payload of one multiframe has been packetized.

3.4.1.3 CAS Conditioning and Freezing in RX Direction

Usually the framer device is responsible for signalling freezing or signalling conditioning in case of line failure. If the framer doesn't support the feature the IWORX can also fulfill the requirements according to Bellcore TR-NWT-000170.

Pin "RFREEZE = 1" indicates that the CAS information from the framer device is invalid and CAS freezing in RX direction is starting. This state remains active until valid CAS bits are available indicated by "RFREEZE = 0".

CAS freezing means that the last valid CAS information is repeated as long as the error cause exists. In case of CAS conditioning for each timeslot individual user defined CAS

conditioning nibbles are sent instead. CAS conditioning in receive direction can be enabled by the user at any time.

3.4.1.4 Segmentation Buffer

The Segmentation Buffer provides 256 bytes of memory for each timeslot, totalling to 64 KB for 8 PDH interface ports with 32 timeslots each. The buffer for each timeslot consists of 4 blocks with 64 octets:

Buffer size = 8 Ports x 32 Channels x 4 Blocks x 64 Octets (1)

In unstructured CES mode, one Segmentation Buffer per PDH port provides 16 blocks.

In structured CES mode, a Segmentation Buffer per logical channel with a variable capacity depending on the number of timeslots and the cell filling level is automatically configured. The number of memory blocks depends on the bandwidth of the channel. Thus for structured CES with N x 64-kbit/s there are N x 4 blocks per connection. Each logical channel can occupy 1, 2 or 4 block-groups (4, 8 or 16 blocks). The first block-group defines the first slot number of the logical channel. The second, third and fourth block-groups define the number of the corresponding interface slot of the logical channel.

The one-to-one relationship between timeslots and groups of memory blocks allows dynamic re-configuration of a specific logical channel without disturbing other logical channels of the same PDH interface port.

Table 19 Relationship between Cell Filling and Segmentation Buffer Subblock Size

Cell Filling AAL1, no SDT (octets)	Cell Filling AAL1, with SDT (octets)	Octets per block	Cells per block	Octets per cell
25-47	25–47	64	1	64
4-24	4–24	64	2	32

3.4.1.5 Padding Partially Filled Cells

The value, used for dummy fill of partially filled cells, is programmable. The fill octets carry no information and are ignored at the receiver.

Table 20 shows valid values for the cell filling level, which can be configured. All other values are illegal.

Table 20 Cell Filling Level Values

ATM Adaptation Layer Type	Partially Filled		Completely Filled
	Minimum	Maximum	
AAL1	4	46	47[1)]
AAL1 with CAS	4+Cb[2)]	46	47[1)]

[1)] P format: Cells have only 46 user data octets in P format

[2)] Cb: Required bytes for the CAS subblock in an ATM cell = RoundUp(N/2)

3.4.2 Substitute Mode for AAL1 Segmentation Channels

In “Substitute” mode, data is accepted from the PDH receive interface port, but substituted by a programmable byte-pattern. Cells are written into the Segmentation Buffers. This mode can be used for trunc conditioning to indicate idle (bit pattern = 0x7F) or out-of-service conditions (bit pattern = Ox1A) according to af-vtoa-0078 and TR-NWT-000170 per PDH port. The programmable substitute byte pattern is valid for all PDH ports.

3.4.3 RTS Value Generation and Insertion

The SRTS method is based on a central precise SDH network clock which is available at transmitter and receiver. This network reference clock has to be connected to the pin nrc. At the segmentation side, the difference between the SDH network clock and PDH network clock is built, coded in 4 bit RTS values.

The generated RTS values are packetized in the AAL1 header (one nibble every 8-cell-cycle in the CSI bit of the SN field). 8 ATM cells contain 8 x 47 byte x 8 bit/byte = 3008 bits of data. In case of an E1 line, the data arrives with 2.048 Mbit/s, thus after 3008 bit / 2.048 Mbit/s = 1,46875 ms a complete RTS value is received. The frequency of generated RTS values is 681 Hz.

The RTS value is calculated in the following way:

In N = 3008 cycles of Fdata, we have Mq cycles of the reduced network clock. The reduced network clock Fnx has to fulfil the following equation: 1 <= Fnx / Fdata < 2. This defines the value of x in the equation: Fnx = 8 kHz X 19440 / 2^x. For a full E1 line Fdata = 2.048 MHz, x = 6 and Fnx = 2.43 MHz. The maximum input frequency deviation of 200 ppm (E1 lines: less than 50 ppm) of the data clock translates in a deviation from Mq. At the receiving side, the same network clock is available and the numbers N and x are known. As a result, the nominal value Mnom of Mq is known, and only the deviation from Mnom has to be transmitted. The number of bits to transmit the deviation (p = 4) has to be sufficient for the maximum frequency deviation.

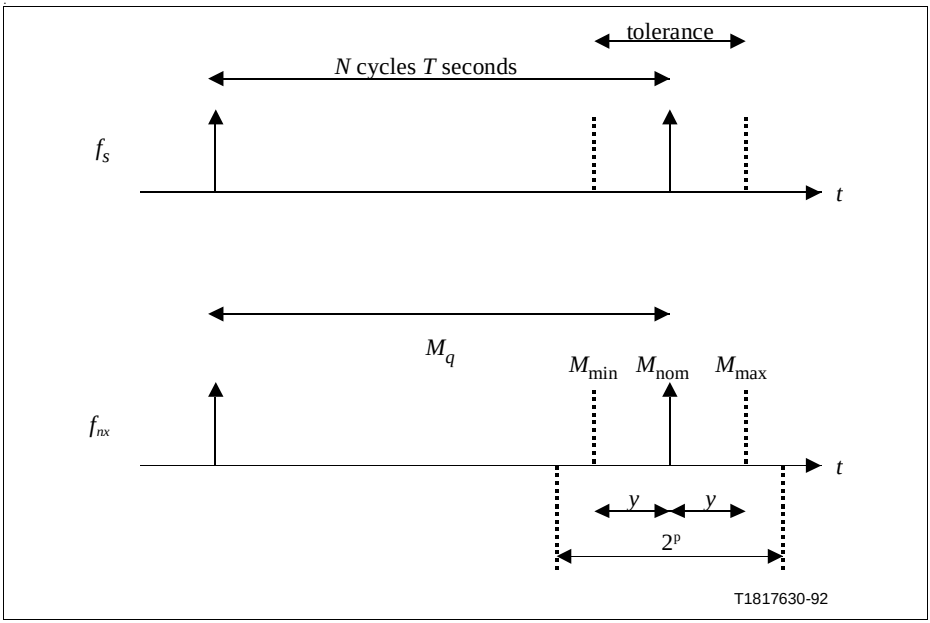


Figure 37 The Concept of Synchronous Residual Time Stamp (SRTS) (Fig. 5/ I.363.1)

RTS values are generated as follows:

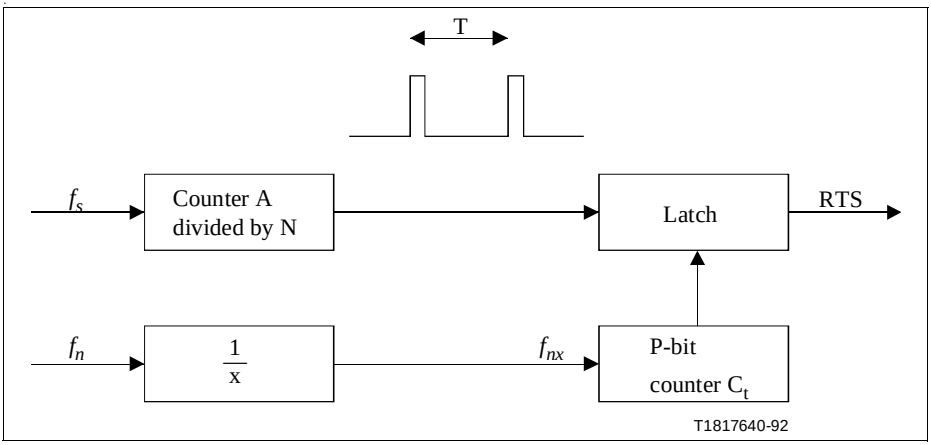


Figure 38 Generation of Residual Time Stamp (RTS) (Fig.6/ I.363.1)

CONFIDENTIAL

The generated RTS values are stored in a PDH port specific RTS buffer before packetized in ATM cells. To guarantee that the value stored in the RTS buffer of the PDH port is correct, the procedure indicated in Figure 39 is followed.

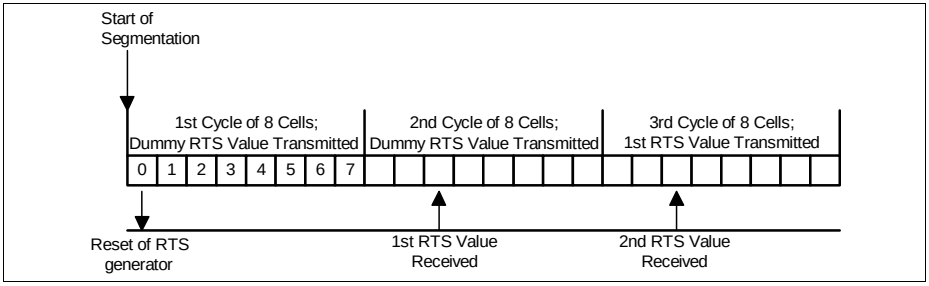


Figure 39 Synchronization of SRTS Generation with the Start of Segmentation

After the start of segmentation, during the 1st cycle of 8 cells, the RTS generator of the corresponding PDH port is reset. During this cycle a dummy RTS value 0001_B is transmitted.

During the 2nd cycle of 8 cells, it's expected to receive the first valid RTS value while transmitting a dummy RTS value.

During the following cycles of 8 cells the RTS value received in the previous cycle will be transmitted.

3.4.4 AAL1 Reassembly Functions

- SNP field check
- SN field check
- SDT pointer detection and verification
- RTS value extraction
- SRTS or ACM clock recovery
- CAS conditioning and freezing in TX direction
- Insertion of dummy cells at cell loss
- Write to Reassembly Buffer
- Read octets from Reassembly Buffer
- Handling of Reassembly Buffer Overflow
- Handling of Reassembly Buffer underflow
- CDV compensation
- Synchronization of structure start in reassembly buffer with structure start of PDH interface port
- Statistics counter event generation (see appendix of the Programmer's Reference Manual)

3.4.4.1 Sequence Number Protection Field Check

When an uncorrectable multi-bit error is detected the Sequence Number (SN) field of the SAR-PDU header is declared invalid, otherwise the SN field is valid. This function is always enabled.

3.4.4.2 Sequence Number Field Check

This function implements the sequence number processing. Selection can be made between Robust and Fast Sequence Count Algorithm as defined in the ITU-T I.363.1.

Robust Sequence Count Algorithm

This algorithm is completely described in annex D of the ETSI B-ISDN AAL type 1 Specification and ITU-T I.363.1 and is shown in **Figure 40**.

The algorithm is described by a state machine of 5 states. A change in states within the state machine is indicated by an arrow, on which there are two distinct values represented. The first value refers to the event that originates the state change, and the second value refers to the action to be taken as a result of that event.

A decision in this algorithm is taken after evaluation of 2 consecutive SN. This means that when a cell is received it must be temporarily stored, waiting for the next cell before it is finally passed to the reassembly buffer. In the state machine, an action to be taken (accept or discard) always refers to the stored cell.

The sequence counting of modulo 8 permits that the algorithm detects a maximum of to 6 consecutive lost cells and 1 misinserted cell, assuming that misinsertion of one cell is at least as probable as the loss of 7 consecutive cells.

Lost cells are compensated by inserting an appropriate number of dummy cells into the transmitted data of the channel. This is required to maintain bit count integrity. The number of octets inserted per dummy cell is equal to the number of user information octets in the SAR-PDU payload of each cell.

When one misinserted cell is detected, the algorithm is able to delete the misinserted cell, because of the delay of one cell in taking a decision.

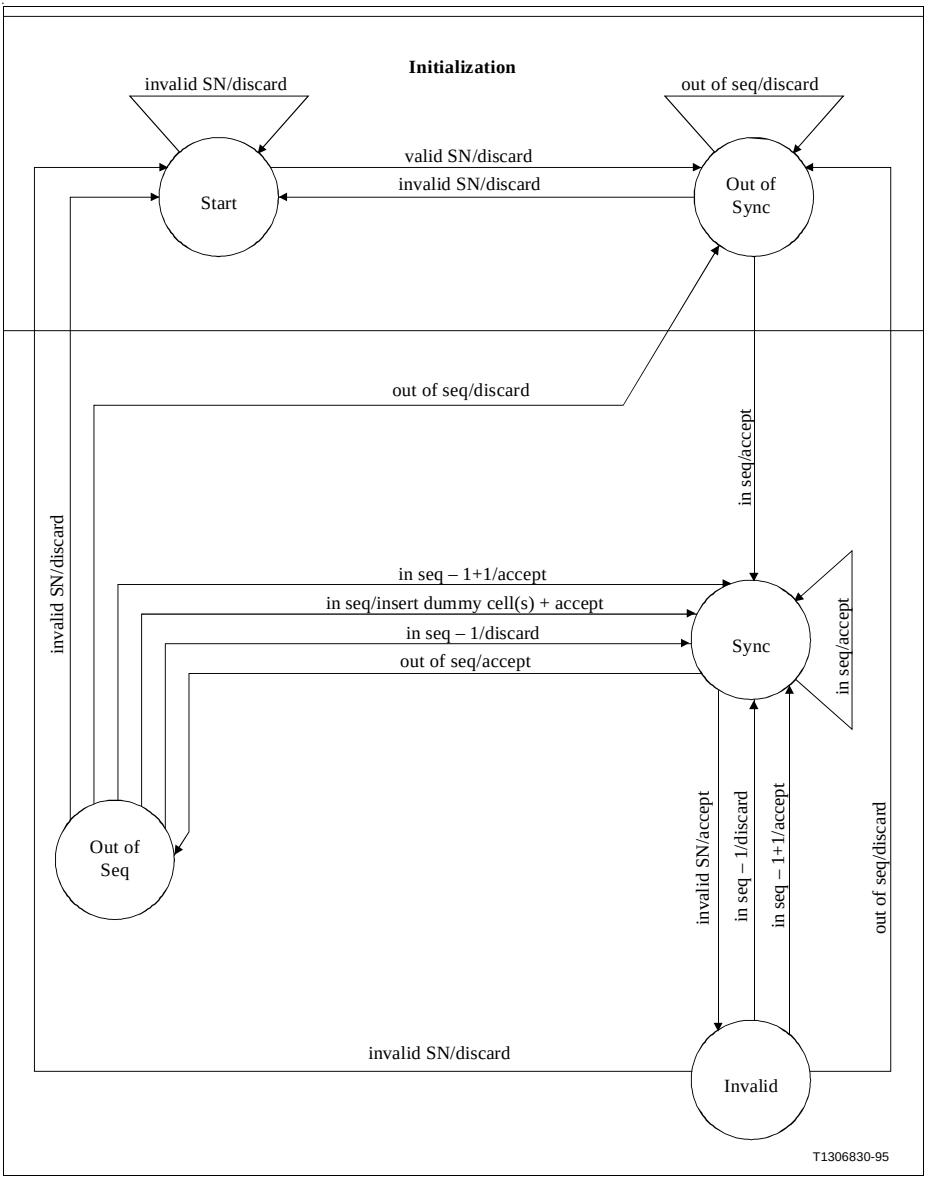


Figure 40 Informative and Example Algorithm State Machine (Fig. III.2/I.363.1)

CONFIDENTIAL

Fast Sequence Count Algorithm

The state machines of the robust SC algorithm and the fast SC algorithm are the same. The only difference is that in the fast algorithm, the action to be taken always refers to the currently received cell, while in the standard algorithm it refers to the temporarily stored cell. Therefore the fast SC algorithm does not introduce additional one-cell delay.

In the fast SC algorithm, a misinserted cell is immediately accepted in the reassembly buffer. Only at the arrival of the next cell, it is detected that the previous cell was misinserted. Because the misinserted cell was already accepted, the current (in sequence) cell will be discarded instead. Lost cells are compensated with the insertion of dummy cells as in the standard algorithm.

3.4.4.3 RTS Extraction and Verification

If the SRTS clock recovery is used for clocking a PDH interface port, RTS values are extracted, verified and written to a PDH port specific FIFO.

The RTS value consists of the four CSI bits of the cells with odd SC values within a cycle of 8 cells. A RTS value is accepted as correct if the following condition is true:

- The SN field is valid
- Four consecutive odd SC values (1, 3, 5 or 7) were received in the previous cycle of 8 cells

Otherwise a dummy RTS-value is written to the FIFO.

When the start of a new cycle is detected, the RTS value of the previous cycle is written to the FIFO.

The RTS Receive FIFO compensates the Cell Delay Variation (CDV). Each RTS Receive FIFO provides space for 8 RTS values. After reaching the initial filling level of 5 RTS values, delay variations of +3 / -5 RTS values can be compensated. This corresponds to a maximum CDV of -4.4 / +7.3 ms (E1) or -5.8 / +9.7 ms (T1).

3.4.4.4 Pointer Field Detection and Verification

If Structured Data Transfer SDT is used, the SAR-PDU payload is supposed to be of the P format under the following conditions:

- The SN field is valid
- Even SC value (0, 2, 4 or 6)
- The CSI field = 1

Only the first cell with CSI bit = 1 in a cycle of 8 cells is supposed to contain a P format SAR-SDU. The other cells with CSI bit = 1 within the same cycle are treated as non-P format. This operation is recommended by ITU-T I.363.1.

In the cells that are supposed to contain a P format SAR-SDU, the pointer field is verified and accepted under the following conditions:

- The parity bit is correct as defined in the ITU-T I.363.1
- The value of the offset field is between 0 and 93 or is the dummy value 127.

If an invalid pointer field (93 < pointer < 127) is detected, its content is replaced by the dummy value (127). The SAR-SDU is processed as if it would have been received with a dummy pointer value. The P format of the SAR-PDU payload is assumed and the first octet of the SAR-PDU payload is not processed as user information.

The verification of the parity bit in the pointer field is always enabled.

Either multiframe or frame based SDT is supported.

3.4.4.5 CAS Conditioning and Freezing in TX Direction

An internal signalling buffer holds the CAS bits. In case of buffer underflow or pointer mismatch the IWE8 provides TX CAS conditioning and freezing according to Bellcore TR-NWT-000170.

The selection between both can be done individually for each logical channel. Values for conditioning can be configured for each timeslot of the logical channel.

The spare and alarm indication bits of the first E1 frame can be programmed by the user as well as the CAS information of idle time slots.

3.4.4.6 Insertion of Dummy Cells at Cell Loss

Upon cell loss detection, the sequence count algorithm will insert dummy cells into the Reassembly Buffer to maintain bit count integrity. The maximum amount of consecutively inserted cells is 6.

These dummy cells are physically inserted when reading the Reassembly Buffer. The Reassembly Buffer itself contains only control field in front of the payload of the next accepted cell, indicating the amount of dummy cells to be inserted.

Inserted dummy cells are not taken into account for the ACM Reassembly Buffer filling level calculation. This means that the buffer filling level is incorrect as long as dummy cells are physically inserted.

The starvation octet used for the dummy cells is selectable by the user.

3.4.4.7 Reassembly Buffer

The purpose of the Reassembly Buffer is to compensate the Cell Delay Variation (CDV) of the ATM network.

It is located in internal RAM providing 512 byte of memory for each timeslot, totalling to 128 KB for 8 PDH ports with 32 timeslots each. The buffer for each timeslot consists of 8 memory blocks with 64 octets:

$$\text{Buffer size} = 8 \text{ Ports} \times 32 \text{ Channels} \times 8 \text{ Blocks} \times 64 \text{ Octets} \quad (2)$$

The number of memory blocks used depends on the bandwidth of the channel (N*64-kbit/s). Thus for structured CES with N*64-kbit/s there are N x 8 memory blocks per connection.

The one-to-one relationship between timeslots and groups of memory blocks allows dynamic re-configuration of a specific logical channel without disturbing other channels of the same PDH port.

3.4.4.8 Handling of Reassembly Buffer Overflow

Overflow is detected when, at the moment of storing an accepted cell, the extra payload of the new cell in the buffer would exceed the logical size of the Reassembly Buffer. The Reassembly Buffer for this logical channel is automatically re-initialized after the overflow event.

3.4.4.9 Handling of Reassembly Buffer Underflow

An underflow period is detected when no octets are available in the Reassembly Buffer to be passed to the framer transmit interface. During the underflow period starvation octets are passed to the framer transmit interface. The underflow is considered to be caused by an extremely late cell. As soon as start of underflow is detected, the Reassembly Buffer is re-initialized automatically.

3.4.4.10 Synchronization of SDT Structure with Port Structure

In normal operation the “ATM start of structure” - start of structure in the reassembly buffer - is synchronized with the “Port start of structure” - start of structure of the PDH interface port. Since this synchronization may get lost, the coincidence of both events is monitored. If they do mismatch, a internal counter is incremented and the Reassembly Buffer is re-initialized

To compensate cell loss the Sequence Count algorithm inserts dummy cells filled with starvation octets. In case the cell filling level is 46 octets or less, the bit count integrity won’t be violated as the length of the AAL-user information within one SAR-SDU is always the same. When operating with a cell-filling of 47 octets, the AAL-user information maybe 47 octet in case of non-P format or 46 octet in case of P format SAR-PDU. As the information on the lost cell’s SAR-PDU format is not available, it is possible that an excess of starvation octets is transmitted. As a result, the “ATM start of structure” might be out of phase with the “Port start of structure”.

The following procedure is implemented for re-synchronization:

- At the end of expanding a burst of dummy cells a flag is set, indicating that a phase shift might occur. The maximum phase shift is 2 octets (e.g. 2 cells with pointers are lost within a sequence of eight cells)

- When an “ATM start of structure” is received and a positive phase shift is detected lower than or equal to 2 octets, an equal number of octets is deleted in the Reassembly Buffer and the flag is reset.
- When the detected phase shift is larger than the allowed value or negative the flag is reset and the Reassembly Buffer is re-initialized.
- When no phase shift is detected the flag is reset.

3.4.4.11 Physical Reassembly Buffer Size

Based on the cell filling level and use of SDT, a memory block is divided into subblocks, where the user data octets of a single cell are stored. The size of the memory subblock per Reassembly Buffer is automatically adapted. Table 21 shows this relationship.

Table 21 Relationship between Cell Filling and Reassembly Buffer Subblock Size

Cell Filling AAL1, no SDT (octets)	Cell Filling AAL1, with SDT (octets)	Octets per block	Cells per block	Octets per cell
32–47	31–47	64	1	64
16–31	15–30	64	2	32
8–15	7–14	64	4	16
4–7	4–6	64	8	8

The physical Reassembly Buffer size used for a N x 64 kbit/s connection is given by:

$$\text{Physical Size(octets)} = N \times 8 \times \text{Cell Filling} \times \text{Cells per Block.} \quad (3)$$

3.4.4.12 Automatic Initialization of the Reassembly Buffer

The Reassembly Buffer is configured automatically to compensate Cell Delay Variation (CDV). User input parameter is the maximum expected |CDV| for the logical channel, |CDV|max.

In order to avoid buffer underflow due to large cell distances the amount of initial starvation octets that are passed to the PDH interface upon arrival of the first cell is set according to |CDV|max. On the other hand this number needs to be as small as possible to avoid excessive delay. The logical Reassembly Buffer size is adjusted according to |CDV|max in order to detect too small cell distances by Reassembly Buffer overflow. The contents of the starvation octets can be defined by the user.

The following sections give an overview on the Reassembly Buffer operation and initialization.

CONFIDENTIAL

Unstructured Data Transfer:

After activation of a channel both SAR Receiver and PDH Transmit Interface start operation. As long as no reassembled cell is available in the Reassembly Buffer it is considered to be in underflow condition and starvation octets are passed to the PDH Transmit Interface.

As soon as the first reassembled cell is available in the Reassembly Buffer the device starts building up the Reassembly Buffer threshold level. This is done by passing an additional amount of starvation octets to the PDH Transmit Interface

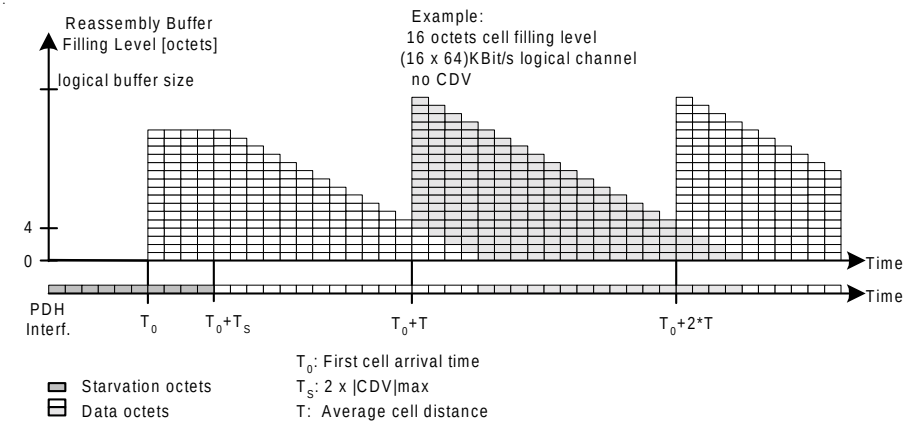


Figure 41 Reassembly Buffer Initialization: No CDV

As the transmission of the reassembled cell stream is delayed by 2 x |CDV|max, there will be octets of the previous cell left in the Reassembly Buffer if the following cell arrives without CDV.

Assuming N octets of data are transmitted within one frame period of 125µs the amount of stravation octects transmitted from T_0 to T_s is:

$$\Delta = 2 \times |CDV|max \times \frac{N}{125\mu s} \quad (4)$$

The worst case for buffer underflow is given if the first cell has maximum positive CDV. In this case the amount of starvation octets inserted after receipt of the first cell has to be bigger than the amount of data transmitted during the expectation interval. Otherwise the Reassembly Buffer will enter underflow condition at any time a cell with maximum positive CDV is followed by a cell with maximum negative CDV.

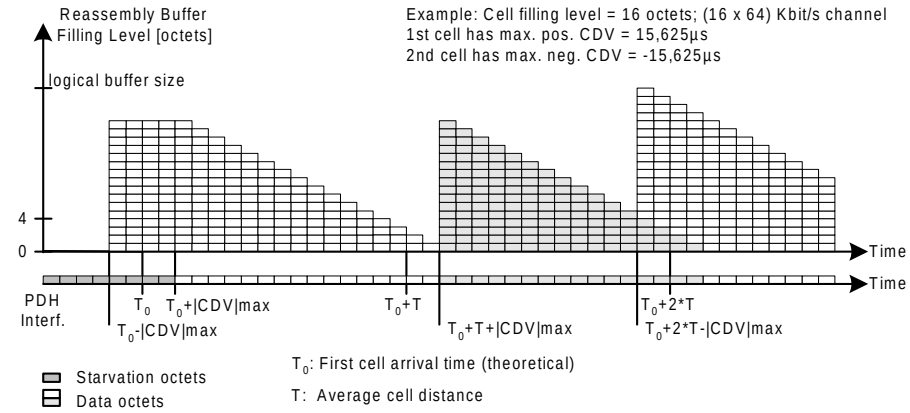


Figure 42 Reassembly Buffer Initialization: positive CDV at Start Up

The worst case for buffer overflow is given if the first cell has maximum negative CDV and then any cell with maximum negative CDV is followed by a cell with maximum positive CDV.

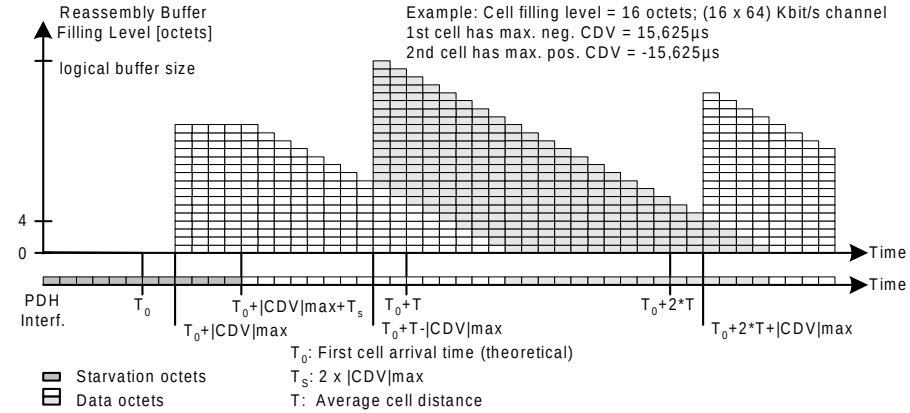


Figure 43 Reassembly Buffer Initialization: Negative CDV at Start Up

If the first cell has maximum negative CDV there will be Δ octets left in the Reassembly Buffer when the following cell arrives with maximum negative CDV. In case the following cell arrives with maximum positive CDV it will be Δ plus the amount of data to be

transmitted in the expectation interval. Just after cell arrival the filling level of the Reassembly Buffer is at its maximum:

$$\text{LogicalBufferSize} = \text{CellFillingLevel} + 4 \times |\text{CDV}|\text{max} \times \frac{N}{125\mu s} \quad [5]$$

The delay introduced by the Reassembly Buffer is:

$$\text{Delay} = 2 \times |\text{CDV}|\text{max} \quad [6]$$

Structured Data Transfer:

After setup of a logical channel both SAR Receiver and PDH Transmit Interface start operation. As long as no reassembled cell in P format is accepted the Reassembly Buffer it is considered to be in underflow condition and starvation octets are passed to the PDH Transmit Interface.

After that, Δ starvation octets are given to the Framer Transmit Interface.

Then, the transmitter reads as many octets from the Reassembly Buffer as indicated by the pointer field. For each octet one starvation octet is given to the PDH Transmit Interface. The next octet to be read from the Reassembly Buffer is the “ATM Start of Structure” (The octet where the AAL1 pointer field points at).

After that, starvation octets are passed to the Framer Transmit Interface until the “Port Start of Structure” is detected. A “Port Start of Structure” occurs when the PDH Transmit Interface requests the first time-slot octet belonging to the channel in the frame or the multiframe.

From that moment on, the “ATM Start of Structure” and “Port Start of Structure” are synchronous and the contents of the Reassembly Buffer are passed to the PDH transmit interface.

The worst case for buffer overflow is given, if the first P format cell has maximum negative CDV, the contents of the pointer field is at its maximum value Pmax and the “Port Start of Structure” occurs right before the receipt of that P format cell. In that case the complete frame needs to be stored in the Reassembly Buffer

If the first cell has maximum negative CDV there will be Δ octets left in the Reassembly Buffer at any time a cell with maximum positive CDV is followed by a cell with maximum negative CDV. the following cell arrives with maximum negative CDV. In case the following cell arrives with maximum positive CDV it will be Δ plus the amount of data to be transmitted in the expectation interval. Just after cell arrival the filling level of the Reassembly Buffer is at its maximum.

CONFIDENTIAL

To allow CDV compensation and SDT structure synchronization, the logical size is automatically programmed to a minimum value given by:

$$\text{LogicalBufferSize} = \text{partfill} + 4 \times |\text{CDV}|\text{max} \times \frac{\text{N}}{125\mu\text{s}} + \text{FR} \times \text{N} + \text{Pmax} \quad [7]$$

with partfill being the cell filling level and with FR being the number of frames in a structure:

FR = 0: when SDT is not used
FR = 1: for frame based SDT
FR = 16: for multi-frame based SDT in E1 mode
FR = 24: for multi-frame based SDT in T1 mode

Pmax is the maximum number of payload octets from the pointer field to the start of structure:

Pmax = N x FR, if N x FR < 2 x Cell filling level
Pmax = 2 x part_fill, if N x FR > 2 x Cell filling level

The delay introduced by the Reassembly Buffer is:

$$2 \times |\text{CDV}|\text{max} \leq \text{Delay} \leq 2 \times |\text{CDV}|\text{max} + \frac{(\text{FR} \times \text{N} + \text{Pmax}) \times 125\mu\text{s}}{\text{N}} \quad [8]$$

3.4.4.13 SRTS Clock Recovery

The transformation of RTS values in a clock is not specified in the SRTS specifications. Basically, the device calculates another RTS value based on the transmit clock. The difference between received RTS values and locally calculated RTS values, drives a DCO. This solution can be described as a PLL with an unusual phase comparator.

Transmit clocks are generated by the internal SRTS PLL. PLL start-up is delayed until 5 RTS values are received. This will take 7.3 ms for E1 and 9.7 ms for T1. During this time the SRTS PLL is free running. A PLL filter is placed behind the SRTS PLL.

PLL-SRTS

PLL-SRTS is used for clock recovery using the SRTS method. It has a cut-off frequency of 20 to 50 Hz. The 19.44 MHz clock at pin NRF is divided to 2.43 MHz. The PLL's use 32 times the outputclock frequency for fulfilling the 0.05 UI jitter requirement (1/32 = 0.03125 UI), in case of E1 32 x 2048 MHz = 65.536 MHz, in case of DS132 x 1.544 MHz = 49.408 MHz.

The phase detector of PLL-SRTS has a linear range which optimized for jitter tolerance requirements. It is defined by a "window" of accepted RTS values. Each time PLL-SRTS detects values, which fall out of the window, or processes invalid values, it is forced in hold over for 1 SRTS period. During start-up of the RTS Receive FIFO, PLL-SRTS is free running.

CONFIDENTIAL

PLL-FILTER

The PLL-FILTER has a very low cut off frequency 0.6 Hz and a tuning range of +/ -240 ppm. It reduces jitter which is generated in, or passed through PLL-SRTS.

3.4.4.14 ACM Clock Recovery

The adaptive clock method does not require information concerning the source clock transferred over the ATM network. The speed of the transmitter is adjusted to the filling level of the receive buffer. If the transmit clock is too slow, the buffer filling level will increase causing the clock recovery circuit to increase the transmit clock frequency. If the transmit clock is too fast the buffer filling level will decrease. In this case the clock recovery circuit will decrease the transmit clock frequency.

Transmit clocks are generated by the internal ACM PLL. Although a PLL-FILTER is placed behind PLL-ACM, it has little or no functionality in case of ACM, as PLL-ACM has a lower cut off frequency.

PLL-ACM

The PLL-ACM is a control system with feedback of 2nd order. Its phase is adjusted according to the filling level of the Reassembly Buffer.

The average buffer filling level AVB is defined automatically by logical channel parameters (bandwidth, |CVD|max) and is subtracted from the current buffer filling level. The result is amplified in order to adjust the cut off frequency and to define the system's damping (number of bytes, needed to drive the DCO over its tuning range. The loop gain is set to factor 4 for best performance regarding jitter and lock in time. The PLL-cut-off frequency is less than 1 Hz. In conjunction with a low pass filter, CDV is very small.

The behaviour of the PLL is characterized by rise time and lock in time. The rise time is the time when the clock output enters the predefined tuning range for the first time. The lock in time is defined as the time after which the clock stays within the accepted deviation.

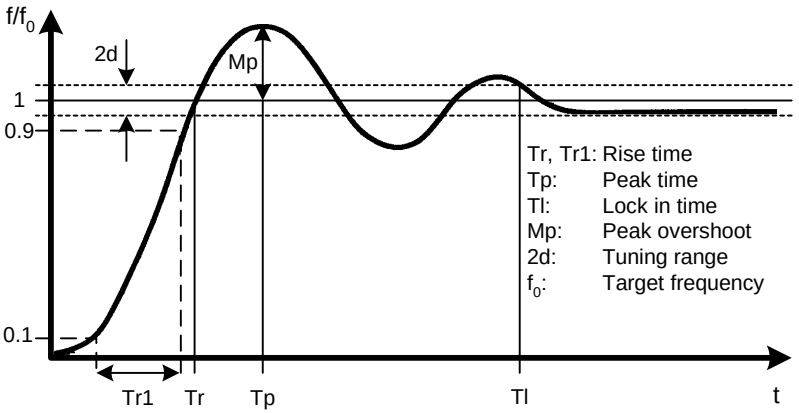


Figure 44 Transient Parameters

The tuning range of the DCO is limited to the value programmed by the user. If the phase detector requests a higher frequency deviation the DCO enters out-of-range condition. In this case the DCO's output will be clipped.

Increasing the loop-gain would reduce the damping of the PLL-ACM. This will reduce the rise time but results in overshoot and long lock-in times.

Reducing the loop-gain would increase the damping. This results in lower cut off frequencies, and prevents overshoot. Thus, CDV is less likely to drive the PLL out of lock. The rise and lock-in time are increased. If the loop-gain is too low, the amount of bytes required to drive the DCO over it's tuning range could cause a data buffer over- or underflow.

Optimized damping allows minimum lock-in time without overshoot. In this case PLL-ACM's frequency is moving asymptotically to the correct value.

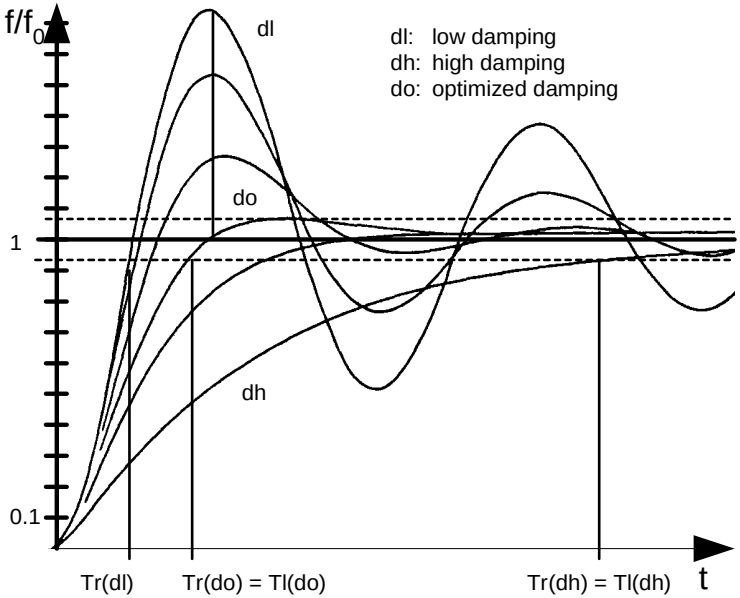


Figure 45 Influence of Damping on Lock in Time

PLL-ACM tries to keep the number of bytes in the Reassembly Buffer at the average buffer filling value AVB. This value is equivalent to the number of bytes stored in the Reassembly Buffer during start-up.

During start-up and restart, PLL-ACM will be free running. During this time the data buffer is filled with an initial number of bytes.

After the initial free run, PLL-ACM will start locking in. The lock in time depends on:

- The difference between the initial number of bytes in the data buffer and the value AVB
- The damping
- The maximum allowed frequency deviation defined by the user
- The required frequency deviation

During this lock-in process, the output frequency might temporarily reach the programmed minimum or maximum value. This strongly depends on the initial difference of the data buffer filling from the value AVB.

As re-initialization of the reassembly buffer is not reported to the internal clock recovery circuit, PLL-ACM will detect a huge difference between data buffer filling and the value AVB. As a result the output frequency will be driven to it's lowest allowed value and stays

there for a relative long period of time. For this reason it is important to program the maximum allowed frequency deviation with the smallest possible value.

Recommendation for maximum allowed frequency deviation:

- +/-50 ppm for E1 networks (as specified in G.703)
- +/-130 ppm for supporting also older T1 equipment (as specified in T1.403-1995)
- +/-32 ppm for new T1 networks

3.5 Address Reduction, Header Translation and External Routing Tag Handling

The device supports up to 256 bi-directional ATM CEP/CPs. 16 selected ATM CEP/CPs can be used as AAL2 paths carrying up to 249 bi-directional AAL2 CEP/CPs each. Address Reduction builds a Connection Handle for each ATM connection by a look-up mechanism.The Connection Handle is attached to the cells transported within IWORX. The internal functional blocks are working on basis of these Connection Handles (e.g. OAM, TM).

Each ATM connection

- May be used as VPC or VCC
- May be used in NNI or uncontrolled UNI mode
- Can be configured to
  AAL1,
  AAL2 over G.804
  ATM/AAL5 over G.804
- Can be routed to the CPU (for future firmware versions)
- Can be traced by the TAD (only for debugging of semiconductor manufacturer in error case)

ATM Connection Handling in Transmit Direction

In transmit direction an ATM connection is identified by its UTOPIA address, the VPI and the VCI bits.

In case of a connection in UNI mode, the GFC bits (VPI[11:8]) may be masked or compared against a fixed value for connection look-up. VPCs are treated as a single ATM connection if the VCI bits are masked for connection look-up. The lower VCI[4:0] bits may be also masked. This allows to map all reserved VCIs on a single ATM connection.

Two different methods of CID handling on AAL2 paths are implemented.

The first assumes that the CID is located at its pre-defined position in the ATM payload of the AAL2U cell, the 6^{th} header octet.

The second assumes that the bit fields of the ATM header are shared for both, the AAL2 path and the CID. In this case a dedicated UTOPIA address needs to be assigned for AAL2 traffic. If a cell is received on that UTOPIA address, the CID is assumed to be located at a user selectable 8-bit field within the ATM header and is copied into the 6^{th} header octet of the AAL2U cell. The user selectable 8-bit field will be ignored for ATM connection look-up. Therefore the associated ATM connection (AAL2 path) will use the same identifier for each of its AAL2 CP/CEPs.

	7			0
1	VPI[11:4]			
2	VPI[3:0]	Reserved		
3	ATM CP[3:0]	CID[7:4]		
4	CID[3:0]	PTI[2:0]	CLP	
5	UDF1[7:0]			

Figure 46 Example Configuration for AAL2 CP/CEPs

The external routing tag is generally discarded.

Once an ATM connection is identified, the IWORX internal routing path is determined depending on the connection context i.e. traffic type (ATM over G.804, AAL1, AAL2 over G.804 or AAL5 over G.804), corresponding PDH port number and additional routing information like “route to CPU” or “trace by TAD”. In parallel the incoming cell’s VPI and VCI are replaced by the values to be used on the transmit connection. For Virtual Path connections the incoming VCI will be transmitted transparently without modification. UNI mode connections may use a user defined GFC field or the one received on UTOPIA to be transmitted.

ATM Connection Handling in Receive Direction

In receive direction an ATM connection is identified by its PDH Port number, the Channel number (first used timeslot of the connection on the PDH port), the VPI and the VCI bits. The amount of cells that do not match existing connections are counted by a unexpected event counter at each PDH Port.

In case of a connection in UNI mode, the GFC bits (VPI[11:8]) may be masked or compared against a fixed value. VPCs are treated as a single ATM connection if the VCI bits are masked for connection look-up. As another means of resource optimization the lower VCI[4:0] bits may be masked. This allows to map all reserved VCIs on a single ATM connection.

Once an ATM connection is identified the IWORX internal routing path is determined depending on the connection context i.e. traffic type (ATM over G.804, AAL1, AAL2 over G.804 or AAL5 over G.804), corresponding UTOPIA port number and additional routing information like “route to CPU” or “trace by TAD”.

For each ATM CEP/CP and AAL2 CEP/CP the incoming cell’s VPI and VCI are replaced by the values to be used on the receive connection and a 10 octet External Routing Tag can be prepended. The first 7 octet of them can be configured by the user. The remaining 3 octets are fixed to zero.

CONFIDENTIAL

In addition a second branch for each AAL2 CEP/CP is implemented in order to allow dual-cast traffic. For each branch of the AAL2 CEP/CP the AAL2U SDU is duplicated, the VPI and VCI fields are replaced and a 7 octet user configurable External Routing Tag can be prepended. However, the CID of receive AAL2U-PDUs is not modified and used for both branches of each AAL2 CP/CEP. Both branches are symmetrical. Therefore they can be individually blocked, unblocked and dynamically re-configured without disturbing each other.

In case of VP connections the received VCI will be forwarded transparently. If a connection is configured to UNI mode the GFC field may be replaced by a user configurable value as well.

Rules for the Connection Handles 255..0

- If AAL2 CPS functionality is applied the Connection Handle Numbers 15..0 have to be used for the AAL2 paths.
- An ATM connection with VCI 31..5,2..1 will not be routed implicitly via the Connection Handle of the VPC, but a dedicated Connection Handle has to be assigned for these connections.
- The Connection Handle Number of virtual channels which are in a virtual path has to be smaller than the Connection Handle Number of the associated virtual path. Hence it’s recommended to allocate connection handle resources from upper numbers to lower numbers and not the other way round.

3.6 Traffic Management

The Traffic Management Unit in IWORX consists of the two major functions Queue Management and Cell Scheduling.

Queue Management is centered around the cell acceptance algorithm. By evaluating a set of thresholds, it provides dynamic buffer sharing and traffic separation while maintaining QoS guarantees.

Cell Scheduling is based on a hierarchy of mechanisms enforcing absolute and relative priorities and rate guarantees as well as providing optional traffic shaping to connections or connection aggregates.

In IWORX Traffic Management is applied to traffic flows from the UTOPIA towards the PDH interface. A pool of 16 scheduler blocks and 255 queues is provided. Scheduler blocks are typically associated with logical channels and limited terminated VPC while queues are associated with ATM connections. Optional per-VC queueing and PCR shaping is possible. Each queue is also assigned to one out of 16 queue classes providing common queue parameters. All queues dynamically share a buffer of 4095 cells. Buffer sharing and connection isolation is achieved by a proper configuration of thresholds enabling selective cell and packet discard mechanisms controlled by the cell acceptance algorithm.

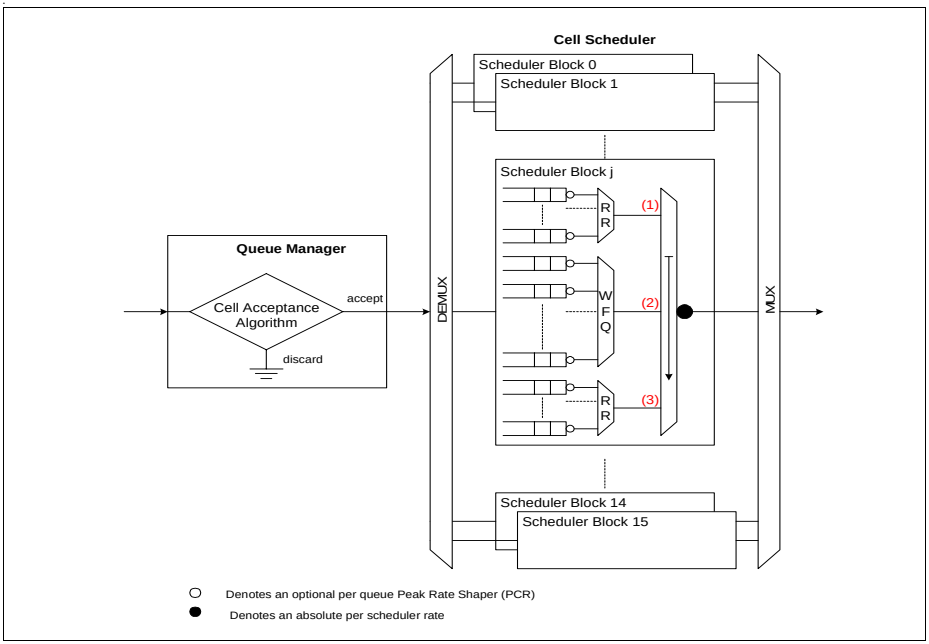


Figure 47 Traffic Management Unit Block Diagram

CONFIDENTIAL

The following description is structured as follows: **Chapter 3.6.1** and **Chapter 3.6.2** provide hardware details of the Queue Manager and Cell Scheduler while **Chapter 3.6.3** describes firmware and higher levels of the control plane.

3.6.1 Queue Manager

3.6.1.1 Functional Overview

The basic function of the Queue Manager (QM) is to decide whether an arriving cell is granted access to the shared buffer or is discarded. This is done by running the Cell Acceptance Algorithm (CAA) (see **Chapter 3.6.1.8**). The queue manager tables accessed by the CAA are summarized in **Figure 48**.

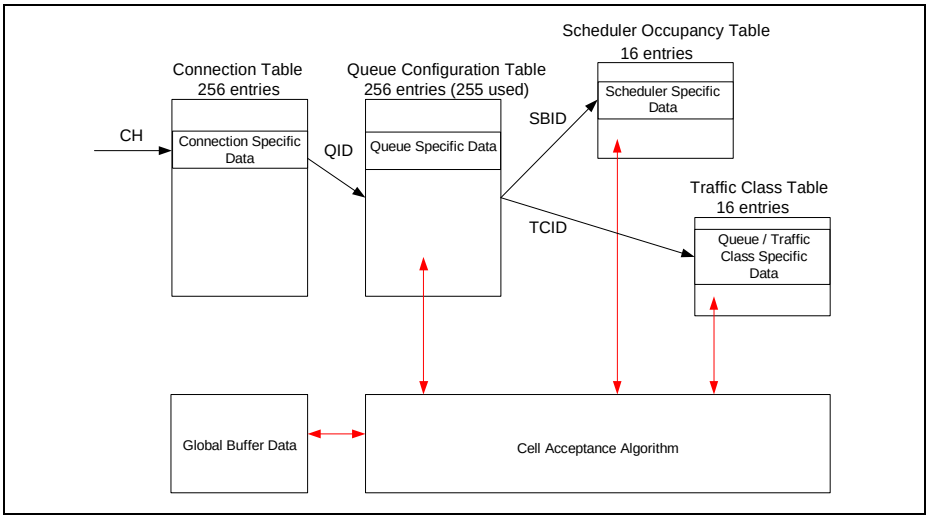


Figure 48 Queue Manager Tables

More generally spoken, the queue manager allocates the buffer resources needed to fulfill the specific service guarantees of individual connections.

In a first step when receiving a cell, the internal connection handle (CH) that was previously assigned by the Header Translation functional block, is mapped to a corresponding Queue Identifier (QID). The QID represents the logical queue the cell will be stored upon acceptance and serves as an index for subsequent table lookups.

With any incoming cell, the CAA can access the current buffer status information containing counters, thresholds and flags. Based on this data, the cell is either discarded or accepted. The respective counters are updated appropriately.

Under normal operation conditions, once a cell is accepted by CAA it will be emitted at a time. The only reason for cell discard after cell acceptance is queue disabling. The cell itself is stored in the internal cell store RAM. The logical queue is simply a linked list of pointers to the cell store providing a FIFO ordering.

3.6.1.2 Global Buffer

A total amount of 4095 cells can be stored in the global cell buffer. Depending on the particular threshold configuration, the global buffer can be exclusively reserved or shared among different queue classes and in consequence to the connections assigned to them.

3.6.1.3 Queue Classes

The concept of queue classes is introduced to provide a logical grouping of queues with common properties, defined by a set of parameters.

The Queue Manager supports up to 16 distinct parameter sets for queue classes in the TCT. Each parameter set includes thresholds as listed in **Chapter 3.6.1.6**.

In addition to thresholds, the following queue class specific flags are defined:

- RTind: Real-Time or non real-Time indicator
- EPDenable: Early-Packet-Discard enable/disable
- PPDenable: Partial-Packet-Discard enable/disable

3.6.1.4 Logical Queues

The concept of logical queues is implemented to provide isolation between connections or groups of connections sharing the global buffer. Strict per-VC queueing is achieved by exclusively assigning connections to logical queues. However, it is also possible to assign more than one connection to a particular logical queue.

A total amount of 255 logical queues is provided, with QIDs ranging from 1 to 255.

Each logical queue is unambiguously assigned to a queue class and inherits the thresholds and flags defined there.

3.6.1.5 Scheduler Blocks

From a queue manager perspective, scheduler blocks (SB) can be conceived as a grouping of logical queues sharing the bandwidth provided by the configured SB rate.

A total amount of 16 scheduler blocks is provided.

Scheduler blocks are usually assigned to ports, logical channels or limited terminated VPCs, providing the necessary rate adaptation.

SB occupancy thresholds are provided for buffer protection in case of SB overload.

3.6.1.6 Occupancy Counter and Threshold Summary

The following table lists the thresholds and related counters grouped into logical buffer entities that are controlled. Refer to the Cell Acceptance Algorithm in **Chapter 3.6.1.8** for

Table 22 Counters and Thresholds

Logical Buffer Entity	Threshold	Scope	Granularity	Range	Related Counter
Global Buffer	ThreshBufMax	global fixed	n.a.	4095	BufferOcc
	ThreshBufNrtEpdClp1	global	16	0 .. 4080	BufferOccNrt
	ThreshBufNrtMax	queue class	16	0 .. 4080	BufferOccNrt
	ThreshBufNrtEpd	queue class	16	0 .. 4080	BufferOccNrt
	ThreshBufNrtCl	global	16	0 .. 4080	BufferOccNrt
Scheduler Block	ThreshSbMaxEpdCi	queue class	16	0 .. 4080	SbOccSB
Queue Class	ThreshQclassMax	queue class	16	0 .. 4080	QClassOcc
Logical Queue	ThreshQueueMax	global fixed	n.a.	2047	LogicalQueueLength
	ThreshQueueMaxEpd	queue class	8	0 .. 2040	LogicalQueueLength
	ThreshQueueCiClp	queue class	8	0 .. 2040	LogicalQueueLength

details on threshold and counter operation. See also **Chapter 3.6.3.1.1**, **Table 25** for a mapping of control plane parameters to thresholds.

3.6.1.7 Discard Mechanisms

3.6.1.7.1 Selective Cell Discard

Selective cell discard based on the CLP marking found in the arriving cells is enabled by the CLP_transparancy flag stored in the connection table. It is used to provide CLP specific service guarantees, e.g. for VBR.2/.3 and GFR.

3.6.1.7.2 Early Packet Discard (EPD)

The Early Packet Discard (EPD) mechanism drops all cells of a packet if the relevant occupancy counter (related to the global buffer, SB, queue class or logical queue) is greater or equal than the corresponding threshold when the first cell of the data packet arrives. By doing so, utilization of buffer and bandwidth resources is improved.

EPD may only be applied to non real-time connections. The mechanism is enabled by the software configurable flag EPDenable, specified per queue class.

3.6.1.7.3 Partial Packet Discard (PPD)

In general, it may happen that an incoming cell is discarded although the EPD algorithm has accepted it. In this case it is meaningful to discard also all following cells of the packet. However, the last cell of a partially discarded packet should be buffered, since the reassemble mechanism at the receiver is triggered by the last cells of user data packets. This mechanism is referred to as Partial Packet Discard (PPD)

PPD may only be applied to non real-time connections. The mechanism is enabled by the software configurable flag PPDenable, specified per queue class.

Note: For both EPD and PPD mechanisms, OAM cells are not discarded.

3.6.1.8 Cell Acceptance Algorithm

The following pseudo-code provides a simplified illustration of the cell acceptance algorithm based on the parameter set listed in **Chapter 3.6.1.6**. It shall illustrate the meaning and compare values of thresholds rather than the complete algorithm. Some thresholds only apply if the corresponding options in the queue class table are selected.

3.6.1.8.4 Threshold Exceed Algorithm

Global Buffer

Algorithm part based on global buffer thresholds and occupancy counters:

```
/*****  Tail drop ****************************************/
IF      [(BufferOcc = ThreshBufMax) OR
        ((BufferOccNrt >= ThreshBufNrtMaxc) AND RTindc = FALSE)]
THEN    CellAcceptedByThreshGlobal = FALSE
ELSE    CellAcceptedByThreshGlobal = TRUE

/*****  EPD support **************************************/
IF      [(BufferOccNrt >= ThreshBufNrtEPDc) AND
        (EPDenablec = TRUE) AND (RTindc = FALSE)]
THEN    ThreshExceedEpdGlobal = TRUE
ELSE    ThreshExceedEpdGlobal = FALSE

IF      [(BufferOccNrt >= ThreshBufNrtEpdClp1) AND
        (RTindc=FALSE) AND
        (CLP=1)AND
        (CLP_transparencyk = FALSE) AND
        (EPDenablec= TRUE)]
THEN    ThreshExceedEpdClp1Global = TRUE
ELSE    ThreshExceedEpdClp1Global = FALSE

/*****  EFCI Support *************************************/
IF      [(BufferOccNrt >= ThreshBufNrtCI) AND
        (RTindc = FALSE)]
```

```
THEN    ThreshExceedCiGlobal = TRUE
ELSE    ThreshExceedCiGlobal = FALSE
```

Logical Queue

Algorithm part based on logical queue thresholds and occupancy counters:

```
/*****  Tail drop ****************************************/
IF      [((LogicalQueueLengthi = ThreshQueueMaxG))OR
        ((LogicalQueueLengthi >= ThreshQueueMaxEpdc) AND
        (EPDenablec = FALSE)) ]
THEN    CellAcceptedByThreshQueue = FALSE
ELSE    CellAcceptedByThreshQueue = TRUE

/*****  Selective Cell Discard ******************************/
IF      [(LogicalQueueLengthi >= ThreshQueueCiClpc) AND
        (CLP=1) AND
        (CLP_transparencyk = FALSE) AND
        (EPDenablec = FALSE)]
THEN    CellAcceptedByThreshQueueClp1 = FALSE
ELSE    CellAcceptedByThreshQueueClp1 = TRUE

/*****  EPD / GFR support ***********************************/
IF      [(LogicalQueueLengthi >= ThreshQueueCiClpc) AND
        (CLP_transparencyk = FALSE) AND
        (EPDenablec = TRUE)]
THEN    ThreshExceedEpdGfrQueue = TRUE
ELSE    ThreshExceedEpdGfrQueue = FALSE

IF      [(LogicalQueueLengthi >= ThreshQueueMaxEpdc) AND
        (EPDenablec = TRUE)]
THEN    ThreshExceedEpdQueue = TRUE
ELSE    ThreshExceedEpdQueue = FALSE

/*****  EFCI Support ****************************************/
IF      [(LogicalQueueLengthi >= ThreshQueueCiClpc)]
THEN    ThreshExceedCiQueue = TRUE
ELSE    ThreshExceedCiQueue = FALSE
```

Queue Class

Algorithm part based on queue class thresholds and occupancy counters:

```
/*****  Queue class limit ***********************************/
IF      (QClassOccc >= ThreshQclassMaxc)
THEN    CellAcceptedByThreshQclass = FALSE
ELSE    CellAcceptedByThreshQclass = TRUE
```

CONFIDENTIAL

Scheduler Block

Algorithm part based on SB thresholds and occupancy counters:

```
/*****  Tail drop ****************************************/
IF      [(SbOccSB >= ThreshSbMaxEpdCic) AND
        (ABRenc =FALSE) AND
        (EPDenablec = FALSE)]
THEN    CellAcceptedByThreshSB = FALSE
ELSE    CellAcceptedByThreshSB = TRUE

/*****  EPD support **************************************/
IF      [(SbOccSB >= ThreshSbMaxEpdCic) AND
        (ABRenc =FALSE) AND
        (EPDenablec = TRUE) ]
THEN    ThreshExceedEpdSB = TRUE
ELSE    ThreshExceedEpdSB = FALSE

/*****  EFCI Support *************************************/
IF      [(SbOccSB >= ThreshSbMaxEpdCic) AND
        (ABRenc = TRUE)]
THEN    ThreshExceedCiSB = TRUE
ELSE    ThreshExceedCiSB = FALSE
```

3.6.1.8.5 EPD Algorithm

Based on the variables set by the EPD support parts of the threshold exceed algorithm and queue specific variables, the EPD algorithm decides upon the acceptance of a packet.

```
/*****  First cell arrival ***********************************/
IF      [(EPDLastCellOfPacketReceivedk = TRUE) AND
        (UserToUserCell = TRUE)]
THEN    IF      [ (ThreshExceedEpdGlobal AND
                (CLP_transparencyk = TRUE) AND
                (LogicalQueueLengthi > 0))OR
                (ThreshExceedEpdGlobal AND ThreshExceedEpdGfrQueue) OR
                ThreshExceedEpdClp1Global OR
                ThreshExceedEpdQueue OR
                (ThreshExceedEpdSB AND (LogicalQueueLengthi > 0)) ]
        THEN EpdDiscardPacketk = TRUE
        ELSE EpdDiscardPacketk = FALSE
ELSE    Do nothing

/*****  Subsequent cell arrival ****************************/
IF      [(EPDenablec = TRUE) AND
        (UserToUserCell = TRUE) AND
        (EPDDiscardPacketk = TRUE)] - see Note 1
```

```
THEN    CellAcceptedByEPD = FALSE
ELSE    CellAcceptedByEPD = TRUE

IF      (UserToUserCell = TRUE)
THEN    IF      (EndOfPacket = TRUE)
        THEN    EPDLastCellOfPacketReceivedk = TRUE
        ELSE    EPDLastCellOfPacketReceivedk = FALSE
ELSE    Do nothing
```

3.6.1.8.6 PPD Algorithm

If the PPD algorithm is applied, the last cell of a corrupted packet must be accepted.

```
/*****  Last cell arrival ***********************************/
IF      [(UserToUserCell = TRUE) AND (EndOfPacket = TRUE)]
THEN    PPDDiscardRestOfPacketk = FALSE

/*****  All but last cell arrival ***************************/
IF      [(PPDenablec = TRUE) AND
        (PPDDiscardRestOfPacketk = TRUE) AND
        (UserToUserCell = TRUE)]
THEN    CellAcceptedByPPD = FALSE
ELSE    CellAcceptedByPPD = TRUE

/*****  PPD enabling ****************************************/
IF      [(discard incoming cell = TRUE) AND
        (UserToUserCell = TRUE) AND
        (EndOfPacket = FALSE)]
THEN    PPDDiscardRestOfPacketk = TRUE
```

3.6.1.8.7 Overall Cell Acceptance Algorithm

The overall decision whether an arriving cell is buffered is based on the results of the previous algorithms. The arriving cell can only be accepted if all algorithms would accept the cell and if buffer space is. To obtain the overall decision whether a correctly received cell is finally buffered the following algorithm applies:

```
IF      [ (CellAcceptedByThreshGlobal = TRUE) AND
        (CellAcceptedByThreshQueue = TRUE) AND
        (CellAcceptedByThreshQueueClp1 = TRUE) AND
        (CellAcceptedByThreshQclass = TRUE) AND
        (CellAcceptedByThreshSB = TRUE) AND
        (CellAcceptedByEPD = TRUE) AND
        (CellAcceptedByPPD = TRUE) ]
THEN    BufferIncomingCell
ELSE    DiscardIncomingCell
```

3.6.2 Cell Scheduling

3.6.2.1 Functional Overview

Basic function of the hierarchical cell scheduler (QSB) is to properly allocate cell transmission slots to scheduler blocks and within those to queues, enabling them to send buffered cells.

Thereby the QSB allocates the bandwidth resources needed to fulfill the specific service guarantees of individual connections.

Internally the QSB functions are implemented by 2 basic building blocks: 16 identical scheduler blocks (SB) and a subsequent round robin scheduler (SBS) as depicted in Figure 49.

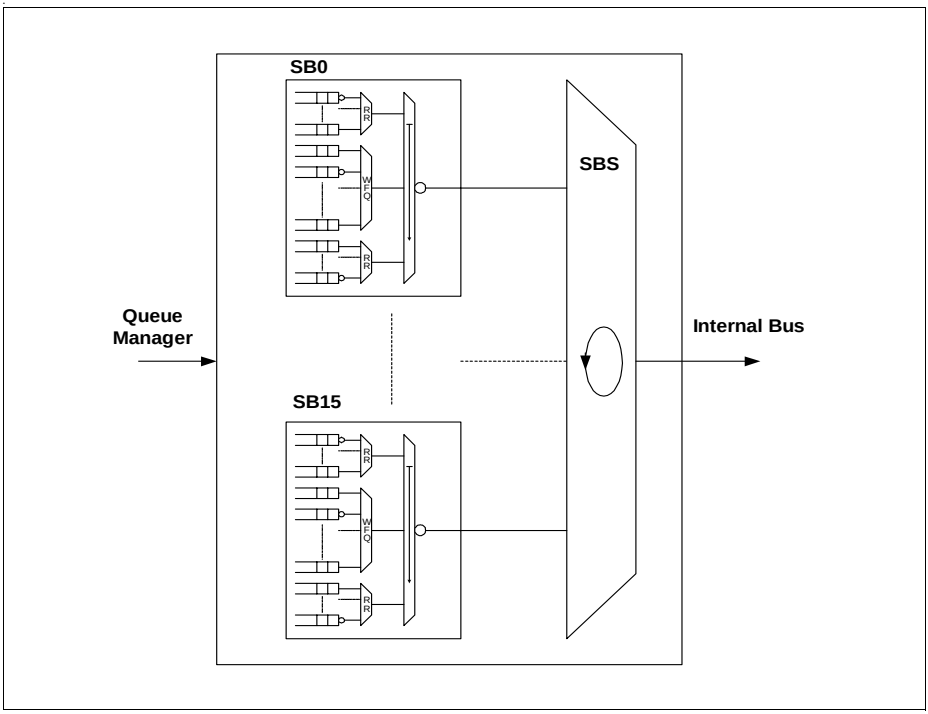


Figure 49 Functional Structure of the Hierarchical Cell Scheduler (QSB)

3.6.2.2 Scheduler Block

Each scheduler block is again a cascade of two scheduling levels, a combination of Weighted Fair Queuing (WFQ) and Round Robin (RR) schedulers in the first stage followed by a priority scheduler in the second stage as shown in **Figure 50**.

An arbitrary number of queues can be assigned to each scheduler input.

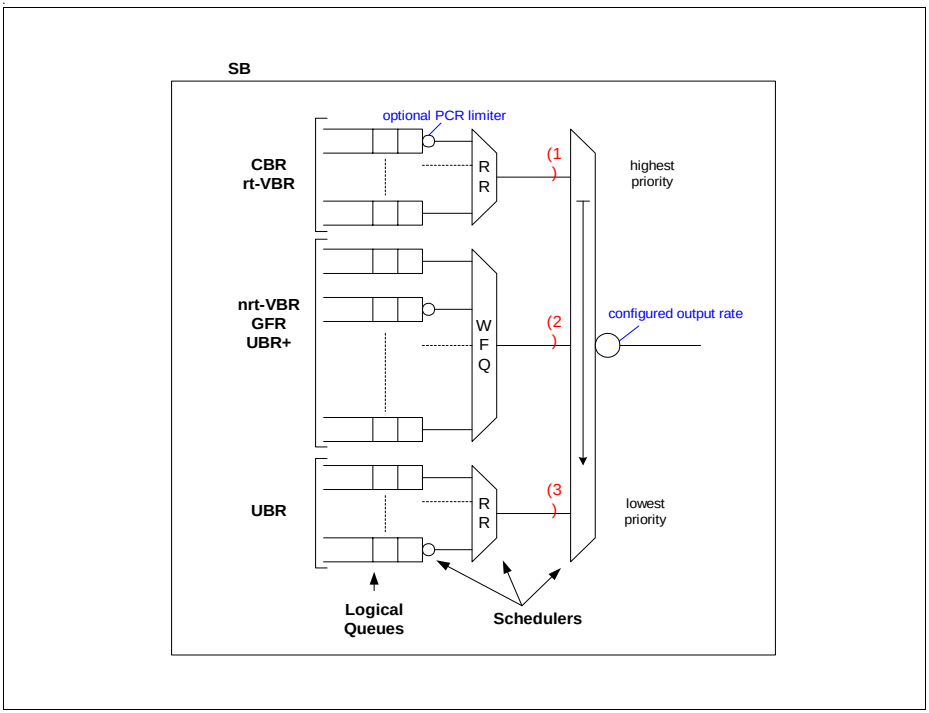


Figure 50 Scheduler Block Structure

One round-robin (RR) scheduler is provided for all real-time (CBR, rt-VBR) connections. It is connected directly to the high priority input of the priority scheduler (marked with the arrow symbol). Thus the real-time traffic is always prioritized. As the output rate of the scheduler block is limited - which is denoted in **Figure 50** with the bubble symbol at the priority scheduler output - the non real-time connections share the remaining bandwidth. Only in the case a cell cycle is not used by either the real-time RR or the WFQ scheduler, low priority traffic from queues connected to the 3rd priority RR scheduler can be served. This 3rd priority can be used for low priority UBR traffic.

3.6.2.2.1 WFQ Scheduler

Rate guarantees for non real-time connections are achieved with the WFQ scheduler. The WFQ scheduler has an arbitrary number of input queues in the range 0 to 255 with a weight factor assigned to each input queue. The absolute values of the weights are irrelevant, only the relative values count.

The WFQ scheduler has the following important properties:

– It is work conserving, i.e. the available bandwidth is always used completely as long as any of the attached queues has cells to send.
– It provides a fair distribution of the available bandwidth in proportion to the assigned weights under any load condition.
– It guarantees minimum rates to queues as long as the sum of the configured minimum rates fits into the available bandwidth.

The properties above make the WFQ scheduler particularly useful for bursty connections with start / stop behavior.

An example for a scheduler with one real-time queue (1) and nine non real-time queues (2..10) is shown in **Figure 51**. Queue 1 is shared by a number of (real-time) connections with different bit rates. For simplicity, the aggregate rate of the real-time traffic is assumed to be constant.

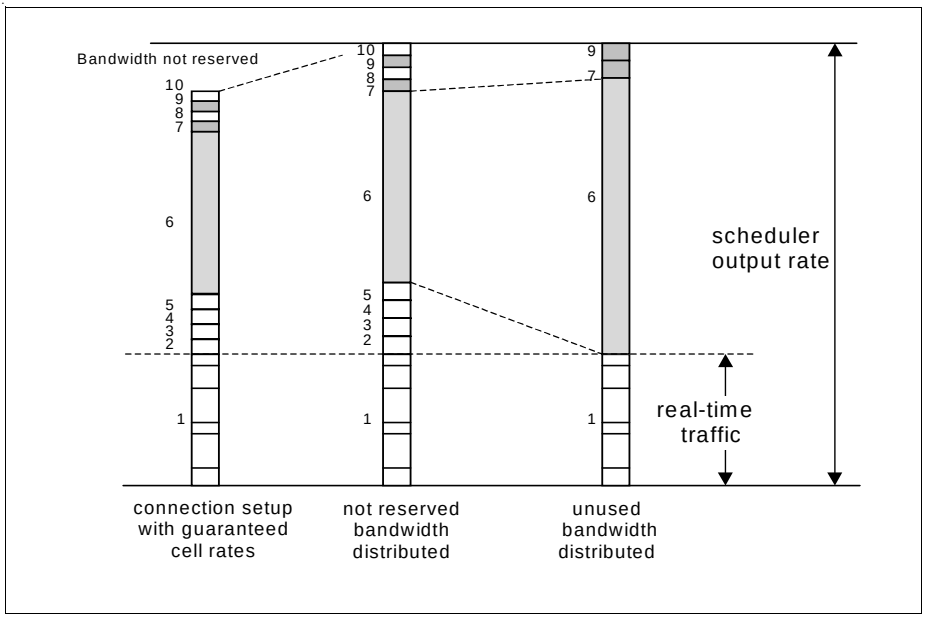


Figure 51 Scheduler Behavior Example

The left column in **Figure 51** shows the load of the scheduler seen from the connection admission control CAC. New connections are accepted as long as their guaranteed rates fit into the spare bandwidth portion of the scheduler.

The center column shows the case that all queues 2..10 are filled, i.e. that all non real-time connections are sending data. The total non real-time bandwidth including the spare bandwidth is then distributed to the 9 queues according to their weight; in this case two weight factors are defined, 1 and 10.

The right column shows the case that only three queues, 6, 7 and 9 are filled, all other connections are not sending data in this moment. Hence the available bandwidth is again distributed fairly among them, still by conserving the 1:10 ratio defined by their weights.

Note that due to it's strict higher priority, real-time connections are not affected by bandwidth re-adjustments in the WFQ scheduler. Conversely the non real-time connections can use spare bandwidth of real-time connections if e.g. a real-time connection stops sending data.

3.6.2.2.2 Logical Queue Assignment and Usage

Assignment of connections to queues is completely arbitrary. In particular, several connections can be assigned to one queue.

For non real-time connections with rate guarantees (nrt-VBR, GFR, UBR+), per VC queueing is recommended, but not mandatory.

The sharing of a logical queue is an option for real-time connections (CBR, rt-VBR) and also for UBR connections.

Note: Rate guarantees always apply to queues. Within a queue shared by multiple connections, the individual connection has no guarantee as the other connections may be unfair.

EPD/PPD functionality is offered by the IWORX on a per-VC basis. Hence, these functions can be supported also for UBR connections sharing one queue.

The assignment of queues to schedulers is also completely arbitrary. One scheduler may have only one queue, another one all the remaining queues.

Optional Peak Rate Shaping

For any queue optional peak rate shaping may be enabled. In case of a peak rate shaping, the queue is first scheduled into a separate time-based calendar. Once emitted by the time-based calendar, the queue is further processed by the associated scheduler in the SB.

3.6.2.2.3 Scheduler Block Assignment and Usage

A scheduler block can be assigned to any logical channel (LC) completely arbitrary. More than one scheduler block can be assigned to one LC.

The scheduler blocks are used to provide rate shaped aggregate cell streams which fit into the bandwidth of the logical channel or limited VPC within the LC.

1. **Using SBs for LC bandwidth limitation**

   In this case, the SB is used to adapt to the LC rate, or eventually to the rate left over by the limited VPCs configured over the LC.

2. **Using SBs for Limited VPC configuration**

   For a limited VPC that contains both real-time and non real-time connections it is recommended to allocate a dedicated SB. This has the advantage that the configured rate of the SB can be shared among the connections while maintaining individual service guarantees. The configured rate is usually constant and will only be modified if the respective VPC rate is modified.

3. **Using Shaped Logical Queues for Limited VPC configuration**

   As a special case, if the limited VPC contains connections which do not require per VC queueing (CBR, rt-VBR, UBR), the VPC can be set up by configuring a PCR limited logical queue. In this case there is no need to allocate a dedicated SB.

4. **Using a dedicated SB for AAL1 connections**

   As AAL1 connections have fixed bandwidth requirements, traffic management can be reduced to assignment of one common, non-shaped real-time queue to all AAL1 connections and assigning a dedicated SB with maximum rate to this queue. Consequently, all AAL1 connections share one dedicated traffic class.

The different mechanisms are shown in **Figure 52**. More details from a resource handling perspective are provided in **Chapter 3.6.3.2**.

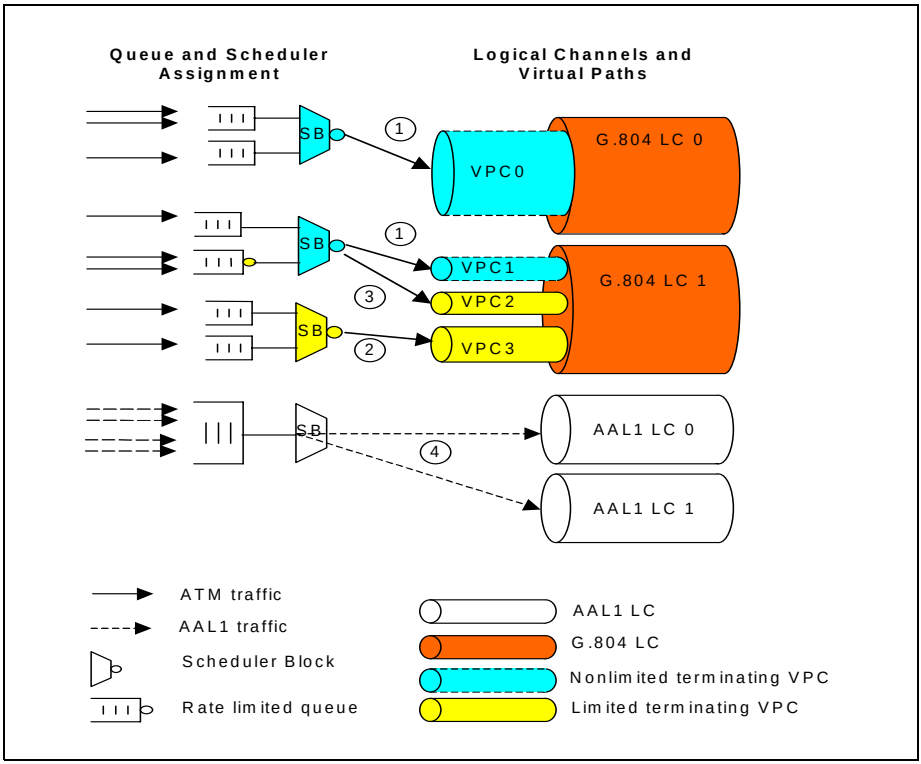


Figure 52 Assignment of Queues and Schedulers

3.6.2.3 Scheduler Block Scheduler (SBS)

The SBS performs a weighted round robin scheduling among the active SB. As long as the sum of the configured SB rates is below the service rate of the SBS, each SB receives bandwidth up to the configured rate, depending on the load in the SB.

The SBS is said to be overbooked if the sum of the configured SB rates is above the service rate of the SBS. This kind of overbooking is not recommended in the IWORX.

An exception to this rule can be made if the in flowing traffic is already limited by other means (e.g. for AAL1 or by PCR limits on all the assigned logical queues).

3.6.2.4 Scheduler Rates and Granularities

3.6.2.4.1 Scheduler Block Scheduler

The aggregate **peak cell rate** of the SBS is calculated as follows:

$$PCR_{SBS} = \frac{SYSCLK}{32} \times \frac{Tstep}{4} \qquad [cells/s]$$

Note: SYSCLK designates the core clock frequency (currently 60 [MHz]). Each cell cycle needs a minimum number of 32 clock cycles.
Tstep is a software configurable parameter with possible values of 1, 2 and 4 allowing to scale QSB rates relative to SYSCLK. A default value of 2 is assumed, allowing a cell emit every 64 clock cycles.

With the core SYSCLK = 60 [MHz] and Tstep = 2 we have PCR_{SBS} = 937500 [cells/s]

This corresponds to 480 [Mbit/s] for 64 byte cells (e.g. on the internal DMA bus) and 397,5 [Mbit/s] for 53 byte cells (towards the PDH interface)

3.6.2.4.2 Scheduler Block

For the **peak cell rate** of an SB we can have $PCR_{SB} = PCR_{SBS}$.

In the following, let LC denote the logical channel assigned to an SB. Recall that a logical channel can subsume the whole PDH port, or an reasonable subdivision.

Let CCR_{SB} denote the **configured cell rate** of an SB (i.e. the desired output cell rate).

$CCR_{SB(LC)} = PCR_{LC}$ must be chosen to match the peak cell rates of the LC as close as possible. Both permanent overload, leading to cell loss in the functional block for ATM cell mapping, and permanent underload, leading to poor channel utilization, must be avoided.

Overall, the following holds

$$\sum_{LC} CCR_{SB(LC)} \leq PCR_{PDH} = 38640 \qquad [cells/s]$$

Note: For short periods of time PCR_{SB} as defined above can occur internally, independent of the particular CCR_{SB}

Deriving Internal Parameters From a Given CCR_{SB}

Internally in the IWORX, the scheduler block output cell rate CCR_{SB} is represented by two parameters:

– $T_{SB(i)}$[14:0], the 15 bit integer division factor
– $T_{SB(f)}$[4:0], the 5 bit fractional division factor

CONFIDENTIAL

These parameters are dimensionless and thus only indirectly represent the output rate. The following formulas shows how to derive the two parameters assuming a given desired output rate CCR_{SB}:

First, a dimensionless floating point number T_{SB} is calculated from CCR_{SB} as follows:

$$T_{SB} = \frac{PCR_{SB}}{CCR_{SB}} \leq 2^{15} - \frac{1}{32} = T_{SBmax}$$

The integer division factor is calculated as:

$$T_{SB(i)} = int(T_{SB})$$

The fractional division factor is calculated as:

$$T_{SB(f)} = ceil\{[T_{SB} - int(T_{SB})] \times 32\}$$

with int(X) designating the integer part of X and ceil(X) designating the next integer greater or equal to X.

The **minimum cell rate** possible in an SB is configured with the maximum value of T_{SB}.

$$MCR_{SB} = \frac{PCR_{SB}}{T_{SBmax}}$$

Examples

In the following the calculation of the integer and fractional divisors is shown for different CCR_{SB} and settings of Tstep.

Table 23 Scheduler Block Limits and Parameter Calculation Examples

Tstep	**1**	**2**	**4**
PCR_{SB} [cells / s]	468750	937500	1875000
MCR_{SB} [cells / s] **MBR_{SB}** [bit / s] (53)	14.3051 6065	28.6110 12130	57.2205 24261
CCR_{SB} = 19320 (IMA 4) [cells/s]			
T_{SB}	24.2624	48.5248	97.0496
$T_{SB(i)}$	24	48	97
$T_{SB(f)}$	9	17	2
CCR_{SB} = 167 [cells/s]			
T_{SB}	2806.8862	5613.7724	11227.5449
$T_{SB(i)}$	2806	5613	11227
$T_{SB(f)}$	29	25	18

3.6.2.4.3 PCR Limited Logical Queues

For each logical queue, an optional peak rate shaper can be programmed. Due to internal scheduling limitations, the peak cell rate values that can be achieved through a PCR limited queue are:

$$PCR_{RSmax} = \frac{SYSCLK}{32} \times \min\left(\frac{Tstep}{4}, \frac{1}{2}\right) \qquad [cells/s]$$

$$PCR_{RSmin} = \frac{PCR_{RSmax}}{16368}$$

Table 24 LQ Rate Shaper Limits

Tstep	1	2	4
PCR_{RSmax} [cells / s]	468750	937500	937500
PCR_{RSmin} [cells / s] PBR_{RSmin} [bit / s]	28.6381 12142	57.2763 24285	57.2763 24285

Recall that all reasonable CCR_{SB} and PCR_{RS} have to be chosen less than the corresponding logical channel rate.
The highest possible LC rate in the IWORX (with IMA 8) is 38640 [cells/s].
The granularity at this rate remains below 1 ‰.

Note: Rate parameters for the per queue PCR shaping are identical to the scheduler output rates. In particular, the value for Tstep is used for both the SBS and the RS queue.

3.6.2.5 Guaranteed Cell Rates and WFQ Weight Factors

The total WFQ scheduler rate is calculated as follows:

$$GCR_{WFQ} = CCR_{SB} - ECR_{RT(SB)}$$

with CCR_{SB} being the configured SB rate as defined above and $ECR_{RT(SB)}$ being the effective cell rate of the real-time bypass in the SB.

CONFIDENTIAL

GCR_{WFQ} is distributed to the queues in proportion to the queue's relative weight factor $1/T_{WFQ}$.

The guaranteed cell rate for connection i is calculated according to:

$$GCR_i = \frac{GCR_{WFQ}}{T_{WFQ(i)}}$$

with T_{WFQ} constrained internally to:

$$T_{WFQ} < 2^{14} - 2^8$$

Therefor T_{WFQmax} = 16127

The minimum guaranteed cell rate at a given GCR_{WFQ} is therefore:

$$GCR_{min} = \frac{GCR_{WFQ}}{T_{WFQmax}}$$

The user must make sure that no overbooking of the WFQ scheduler occurs, i.e.:

$$\sum_i GCR_i \leq GCR_{WFQ}$$

Deriving the Internal Weight Factor From a Given GCR

Assuming a fixed given GCR_{min}, currently set internally to a default of 18 cells/sec, then for any given GCR >= GCR_{min} the corresponding T_{WFQ} is calculated as:

$$T_{WFQ} = \left\lfloor \frac{GCR_{min} \times T_{WFQmax}}{GCR} \right\rfloor$$

The integer function in equation above selects the next smaller value of the integer T_{WFQ}, that is to say, the weight factor is higher than required and, thus, the queue is served slightly faster in order to guarantee the rate.

Two special cases must be considered:

T_{WFQ} = 0 is used to assign the queue to the high priority round robin scheduler.

T_{WFQ} = 16383 is used to assign the queue to the low priority round robin scheduler.

3.6.3 Traffic Management Configuration and Control

This section provides some background needed to handle the commands of the IWORX message catalog related to traffic management.

Figure 53 shows the involved software and firmware layers and subsystems.

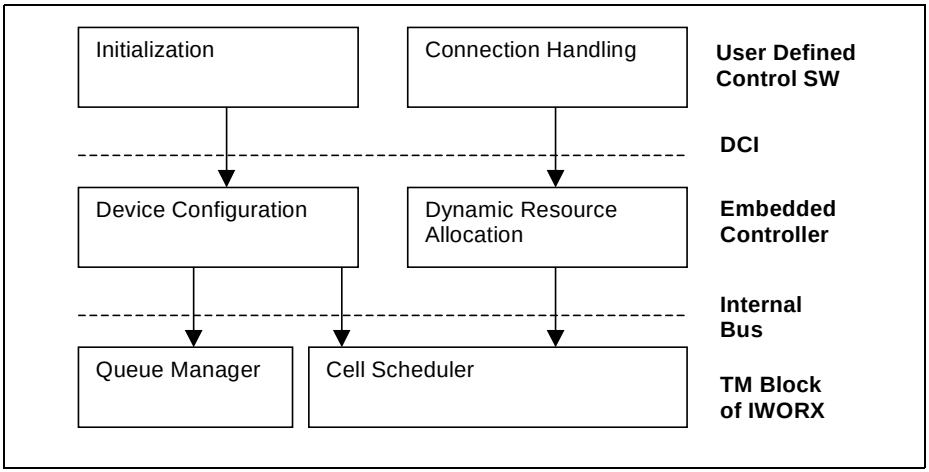


Figure 53 Control Software and Firmware Layers

The DCI provides the interface between the user and the device firmware as described in more detail in the firmware message catalog.

3.6.3.1 Initialization

3.6.3.1.1 Traffic Class Configuration

All QoS guarantees are specified at the level of traffic classes. By mapping a connection to a particular traffic class it is understood that the connection gets access to the predefined resources. Internally, traffic class parameters are mapped to queue class and cell scheduler parameters.

General guidelines

The user, respectively higher level software is responsible for the dimensioning of individual traffic classes. This task may proceed along the following guidelines:

1. Classify the expected traffic at the service level and assume a separate traffic class for each service.
   Possible services are

  - S1:Voice over AAL1
  - S2:Voice over AAL2
  - S3:Data over AAL2 (UMTS terminals are assumed to deliver all traffic over AAL2)
  - S4:Signalling AAL5
2. Select a mapping of the individual service to an appropriate ATM service category.
  - S1:CBR
  - S2:rt-VBR
  - S3:UBR
  - S4:GFR
3. Quantify the traffic parameters and QoS parameters of the ATM connections foreseen to carry the traffic of the individual services.
4. Calculate bandwidth and buffer requirements for the ATM connections.
5. Quantify the number of connections expected to be active simultaneously for each service.
6. Estimate the aggregate bandwidth and buffer requirements for all the connections in one traffic class.
7. Calculate the minimal thresholds for the traffic class such that the QoS requirements are met.
8. Configure the traffic class with the appropriate DCI command.

3.6.3.1.1.1 Traffic Class Parameter Definition

TCMAX:

This parameter controls the maximum number of cells from a particular traffic class that can be accepted into the shared buffer.

The sum of the TCMAX values for all traffic classes should be at least as large as the total buffer space, otherwise the buffer cannot be fully used.

For non real-time traffic classes each individual TCMAX should always be less or equal to BUFNRTMAX, otherwise the threshold is permanently inactive.

TCSCHEDMAX:

This parameter controls the acceptance of cells from a particular traffic class into a particular scheduler block (SB).

Since there are 16 SBs, under homogenous load conditions each SB may hold 1/16 of the cells of the given traffic class. This represents a lower limit for this threshold. To accommodate non-homogenous load, TCSCHEDMAX may be set higher.

TCQUEUEMAX:

This parameter controls the acceptance of cells from a particular traffic class into a particular logical queue (LQ). Depending on the setting of the EPD flag, either whole frames or individual cells are discarded.

Since there are 255 LQs, under homogenous load conditions each LQ may hold 1/255 of the cells of the given traffic class. This represents a lower limit for this threshold. To accommodate non-homogenous load, TCQUEUEMAX may be set higher.

TCQUEUECLP1:

This parameter controls the acceptance of CLP1 cells from a particular non real-time traffic class (except GFR, see below) into a particular logical queue.

The threshold is used to differentiate between CLP transparent and CLP sensitive QoS guarantees.

TCQUEUEEPD:

This parameter is used for GFR traffic classes only to differentiate between the QoS eligible and best effort CLP0 frames in a queue. If this threshold is exceeded together with threshold BUFNRTMAX, CLP0 frames are discarded via EPD.

BUFNRTMAX:

This parameter controls the acceptance of cells from a particular non real-time traffic class into the shared buffer.

See section below for proper setting of this threshold.

BUFNRTEPD:

This parameter controls the acceptance of packets from a particular non real-time traffic class into the shared buffer.

Normally BUFNRTEPD < BUFNRTMAX.

PER_VC

This flag is applicable to traffic classes for the CBR, rt-VBR and UBR service category and enables sharing of logical queues among the connections set up in this traffic class. Per VC queueing implicit for nrt-VBR and GFR.

EPD_EN

PPD_EN

For frame based services, both these flags shall be enabled. This is the default for the GFR traffic class.

The following table shows the mapping between configuration message parameters and internal thresholds.

Table 25 Mapping of Parameters to HW Thresholds

Logical Buffer Entity	Command Parameter	Queue Manager Threshold	Granularity	Range
Global Buffer	EPD_CLP1	ThreshBufNrtEpdClp1	16	0..4080
	BUFNRTMAX	ThreshBufNrtMax	16	0..4080
	BUFNRTEPD	ThreshBufNrtEpd	16	0..4080
Scheduler Block	TCSCHEDMAX	ThreshSbMaxEpdCi	16	0..4080
Queue Class	TCMAX	ThreshQclassMax	16	0..4080
Logical Queue	TCQUEUEMAX	ThreshQueueMaxEpd	8	0..2040
	TCQUEUECLP1 TCQUEUEEPD	ThreshQueueCiClp	8	0..2040

The following table shows the mapping between specific parameters and commands.

Table 26 Threshold Parameter Scope

Parameter	Command / Scope
EPD_CLP1	CMD_TRAFFIC_MANAGEMENT_SET
BUFNRTMAX	CMD_TRAFFIC_CLASS_VBR_NRT_SET CMD_TRAFFIC_CLASS_GFR_SET CMD_TRAFFIC_CLASS_UBR_SET
BUFNRTEPD	CMD_TRAFFIC_CLASS_VBR_NRT_SET CMD_TRAFFIC_CLASS_GFR_SET CMD_TRAFFIC_CLASS_UBR_SET
TCSCHEDMAX	CMD_TRAFFIC_CLASS_*_SET
TCMAX	CMD_TRAFFIC_CLASS_*_SET
TCQUEUEMAX	CMD_TRAFFIC_CLASS_*_SET
TCQUEUECLP1	CMD_TRAFFIC_CLASS_VBR_RT_SET CMD_TRAFFIC_CLASS_VBR_NRT_SET CMD_TRAFFIC_CLASS_UBR_SET
TCQUEUEEPD	CMD_TRAFFIC_CLASS_GFR_SET

3.6.3.1.1.2 Threshold Dimensioning Rules

Buffer threshold dimensioning is governed by the following general rules and formulas.

CONFIDENTIAL

Real-time Traffic

The real-time service categories CBR and rt-VBR have stringent cell loss requirements. In order to guarantee the small cell loss probabilities of these connections, the influence of non real-time traffic is eliminated by the allocation of buffer space exclusively for the real-time queues. Since in general the buffer space needed for this queues is very small, the loss in statistical gain due to the exclusive buffer allocation is negligible.

This kind of buffer reservation is achieved by setting BUFMAX > max(BUFNRTMAX).

Non Real-time Traffic

Non real-time service categories share the remaining buffer space in a statistical manner. It is assumed that buffer space is reserved by CAC on a per connection basis. For nrt-VBR buffer reservation in the order of MBS and a guaranteed throughput according to SCR guarantees negligible cell loss. For UBR no buffer space is reserved.

Exclusive buffer reservation for non real-time traffic can be achieved in IWORX according to the following scheme:

Without loss of generality, assume the ordering among the N non real-time traffic classes to be:

BUFNRTMAX(n) > BUFNRTMAX(n+1) for all n in 1,...,N-1.

The amount of buffer exclusively reserved for class n (not shared with other traffic classes) is called Res_buf(n) and can be calculated as follows:

Res_buf(n) =BUFNRTMAX(n) – BUFNRTMAX(n+1) – TCMAX(0) – ... – TCMAX(n-1)

Note that if Res_buf(n) <= 0, no exclusive buffer reservation is possible for the respective traffic class.

Given a particular exclusive buffer reservation, the following bounds on the available buffer can be derived:

Min_buf_avail(n) =min(max(0, Res_buf(n)), TCMAX(n))

Max_buf_avail(n) =min(BUFNRTMAX(n), TCMAX(n))

The current amount of buffer available for a given traffic class n, Cur_buf_avail(n) is a resource that changes dynamically as connections are established or released and is administered by CAC (see **Chapter 3.6.3.2** below).

3.6.3.1.1.3 Threshold Dimensioning Example

In **Figure 54** below an example threshold dimensioning for four traffic classes is shown.

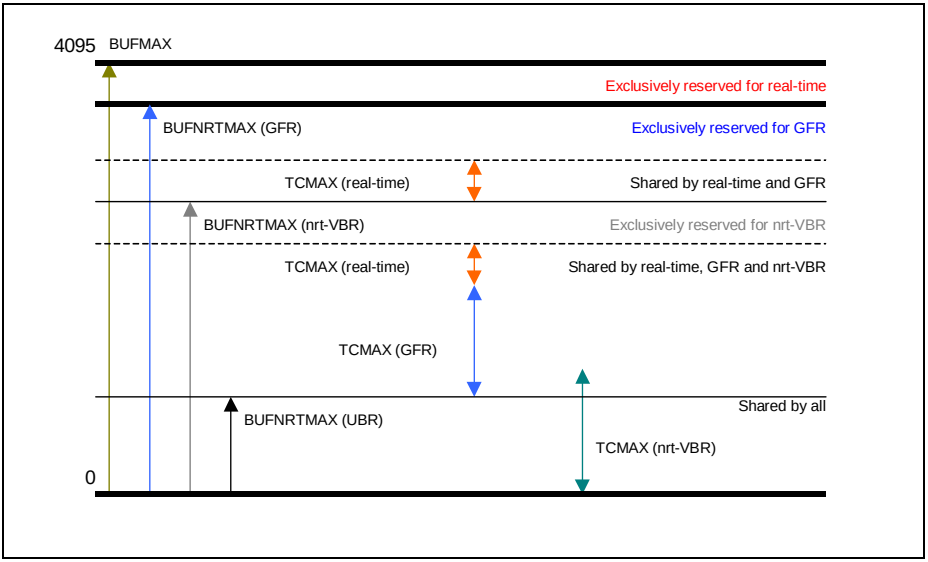


Figure 54 Threshold Dimensioning Principles

It is assumed that all the 16 possible traffic classes are defined at initialization time. The specific parameter settings for individual traffic classes are application dependant.

The following **Table 27** provides a template for N=10 traffic classes.

Table 27 Traffic Class Configuration Example

CMD_ TRAFFIC_ CLASS	TCSET	TCMAX	TCSCHEDMAX	TCQUEUEMAX	TCQUEUECLP1	TCQUEUEEPD	BUFNRTMAX	BUFNRTEPD	EPD_EN	PPD_EN	PER_VC	Comments
CBR	0	256	64	64	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	disable	AAL1
CBR	1	1024	64	64	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	disable	PCR limited
VBR_RT	2	256	64	64	64	n.a.	n.a.	n.a.	n.a.	n.a.	enable	VBR.1
VBR_RT	3	256	64	64	16	n.a.	n.a.	n.a.	n.a.	n.a.	enable	VBR.2/.3
VBR-NRT	4	2048	1024	512	512	n.a.	3824	any	disable	disable	n.a.	VBR.1
VBR-NRT	5	2048	1024	512	32	n.a.	3824	any	disable	disable	n.a.	VBR.2/.3

CMD_ TRAFFIC_ CLASS	TCSET	TCMAX	TCSCHEDMAX	TCQUEUEMAX	TCQUEUECLP1	TCQUEUEEPD	BUFNRTMAX	BUFNRTEPD	EPD_EN	PPD_EN	PER_VC	Comments
GFR	6	3824	1024	1024	n.a.	512	3072	3000	n.a.	n.a.	n.a.	CLP=0 only
GFR	7	3824	1024	1024	n.a.	768	3072	3000	n.a.	n.a.	n.a.	GFR.1/.2
UBR+	8	3824	1024	256	256	n.a.	2048	1900	enable	enable	enable	
UBR	9	3824	1024	256	256	n.a.	2048	1900	enable	enable	disable	sharing one LQ

The GFR traffic classes always have the EPD and PPD flag enabled.

Note that one dedicated traffic class must be defined and reserved for AAL1 connections and all AAL1 connections must be assigned to this particular traffic class at set-up.

3.6.3.2 Connection Handling

3.6.3.2.1 Connection Types

The DCI provides support for the setup, release and where appropriate modify of the following connection types:

Virtual Path Connections (VPC)

– non terminated VPC	CMD_ATM_VPC_XXX
– non limited terminated VPC	CMD_ATM_TERM_VPC_XXX
– limited terminated VPC	CMD_ATM_LIM_TERM_VPC_XXX

Virtual Channel Connections (VCC)

– non terminated VCC	CMD_ATM_VCC_XXX
– terminated VCC	CMD_ATM_TERM_VCC_XXX
– AAL1 channel	CMD_ATM_TERM_VCC_AAL1_XXX
– AAL2 path	CMD_ATM_TERM_VCC_AAL2_XXX

General Assumptions

– Every VCC to be set up is assumed to be assigned to a terminated VPC, set up in advance. There are no VCC set up directly on the LC without relation to an existing terminated VPC.
– For every VPC it is at least known whether it has stringent delay objectives.

Scheduler Block Assignment

The 16 available scheduler blocks are dynamically assigned on demand. The following rules govern the SB assignment:

– Non terminated VPC and non limited terminated VPC are set up directly on a logical channel. They share a common SB denoted SB_{LC}.
– For every limited terminated VPC carrying guaranteed non real-time traffic, a dedicated SB must be assigned, denoted SB_{VPC}.
– For every limited terminated VPC carrying real-time traffic only, a PCR shaped real-time LQ within the common SB can be assigned, denoted LQ_{VPC}.
– For all AAL1 connections a dedicated SB denoted SB_{AAL1} must be assigned.
– The SBs classified above are assigned when the first connection of the particular type is set up and released when the last connection of this type is released.

See also **Chapter 3.6.2.2.3** and **Figure 52**.

Logical Queue Assignment

The available 255 logical queues are dynamically assigned on demand. The following rules govern the LQ assignment:

CONFIDENTIAL

– nrt-VBR, GFR and UBR+ connections are setup in per-VC queuing configurations, that is, an empty queue is reserved for the connection and the queue must be assigned to the WFQ scheduler.
– Several CBR or rt-VBR connections may share a logical queue with a guaranteed rate greater or equal to the sum of the guaranteed rates of the connections.
– Several UBR connections may share a logical queue. For fairness reasons only well behaved UBR connections with similar PCR should be joined.
– Logical queue sharing as described above is selected per traffic class with the parameter PER_VC set to false.

3.6.3.2.2 TCSET Selection

For every connection to be set up, the traffic class has to be determined from both the connections traffic and QoS parameters.

3.6.3.2.3 Connection Admission Control

In **Figure 55** the control flows in CAC are shown.

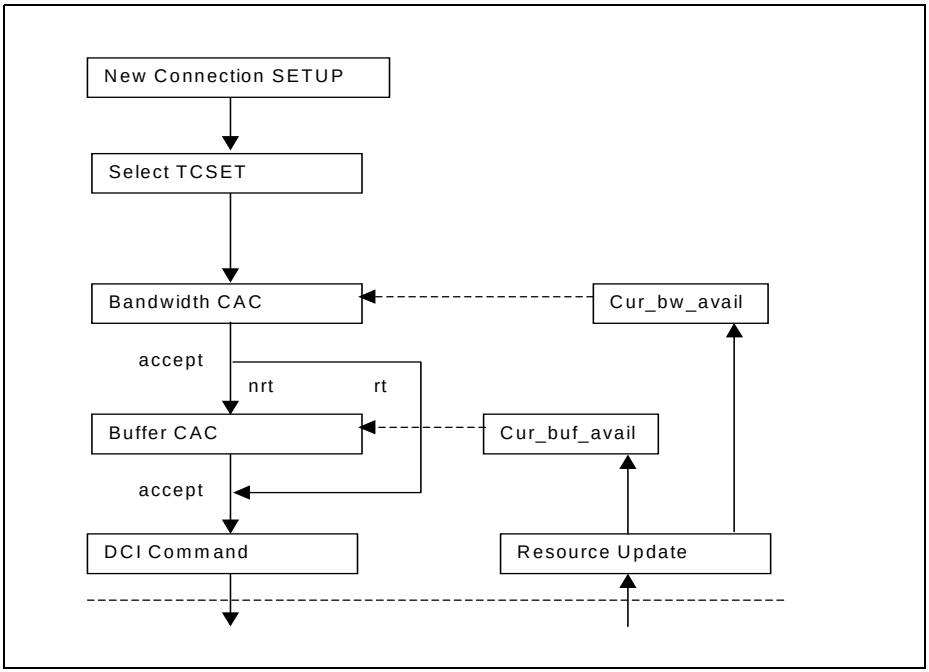


Figure 55 CAC Outline

3.6.3.2.4 Bandwidth CAC

Bandwidth Bottlenecks

– Logical Channel
The usable logical channel rate is controlled by SB_{LC}. The CCR of the SB_{LC} must be adapted to match the bandwidth of the LC, or in case of limited terminated VPC the leftover bandwidth.

– Limited terminated VPC
When setting up a limited terminated VPC, it has to be checked whether it's PCR fits into the available bandwidth of the LC.
When setting up a connection within the limited terminated VPC, it has to be checked wether it's required bandwidth (see below) fits into the available bandwidth of the terminated VPC.

Bandwidth Allocation

The current bandwidth allocation for a particular bandwidth bottleneck B is called Cur_bw_alloc(B). Cur_bw_alloc(B) must be calculated to be sufficient in order to guarantee the QoS of all the individual connections traversing B.

The amount of bandwidth required for a given connection i is equivalent to the guaranteed cell rate GCR(i), which in turn is a function of the connection's traffic parameters and QoS parameters. **Table 28** gives an overview of different bandwidth allocation depending on the ATM service category of the connection.

Table 28 Required Bandwidth Per Connection

Service Category	**GCR**	**Comment**
CBR	PCR	connection parameter
rt_VBR	ECR (effective cell rate)	calculated by CAC
nrt_VBR	SCR	connection parameter
GFR	MCR	connection parameter
UBR+	MCR	connection parameter

The current available bandwidth for a particular bottleneck B, Cur_bw_avail(B) is calculated as follows:

CONFIDENTIAL

– Logical channel

$$\text{Cur\_bw\_avail(LC)} = \text{PCR}_{LC} - \sum_{VPC \in LC} \text{PCR}_{VPC} - \sum_{VCC \in LC} \text{GCR}_{VCC}$$

– Limited terminated VPC

$$\text{Cur\_bw\_avail(VPC)} = \text{PCR}_{VPC} - \sum_{VCC \in VPC} \text{GCR}_{VCC}$$

Then the bandwidth CAC has to check

```
IF      GCR(i) <= Cur_bw_avail(B)
THEN    accept connection at bottleneck B
        update allocated bandwidth
        Cur_bw_alloc(B) = FBW(Cur_bw_alloc(B) , GCR(i))
        -- user defined function FBW
        -- accounting for multiplexing gain where possible
        -- simple addition in most cases
ELSE    reject connection
```

3.6.3.2.5 Buffer CAC

The current buffer allocation for a particular traffic class n is called Cur_buf_alloc(n).

Cur_buf_alloc(n) must be calculated to be sufficient in order to guarantee the QoS of all the individual connections belonging to traffic class n.

The amount of buffer required for a given connection i, also referred to as Guaranteed Cell Buffer GCB(i), is a function of the connection's traffic parameters and QoS parameters. Different functions may be used for different ATM service categories.

Table 29 Required Buffer Per Connection

Service Category	GCB	Comment
CBR	1-3	small constant, F(CDVT)
rt-VBR	MBS	small
nrt-VBR	MBS	connection parameter
GFR	MBS + MFS	connection parameter
UBR+	none	no cell loss objective for UBR+

CONFIDENTIAL

In case of PCR shaped connections, the buffer is needed to temporarily store the cells arriving in excess of the declared peak cell rate. Its size is derived from the PCR and the maximum allowed cell delay variation CDVT.

Assume that the current available buffer for a particular traffic class n Cur_buf_avail(n) is calculated as follows:

$$\text{Cur\_buf\_avail(n)} = \text{Max\_buf\_avail(n)} - \sum_{t \in TC} \text{Cur\_buf\_alloc(t)}$$

with TC denoting the set of traffic classes sharing buffer with traffic class n.

Then the buffer CAC has to check

```
IF      GCB(i) <= Cur_buf_avail(n)
THEN    accept connection into traffic class n
        update allocated buffer
        Cur_buf_alloc(n) = FBUF(Cur_buf_alloc(n), GCB(i))
        -- user defined function FBUF
        -- accounting for multiplexing gain where possible
        -- simple addition in most cases
ELSE    reject connection
```

The following **Table 30** summarizes resource handling for the different connection types.

Table 30 Resource Handling by Connection Type

Connection type	LQ	SB	Bandwidth-CAC	Buffer-CAC
non terminated VPC	dedicated real-time	SB_{LC}	PCR_{VPC}	none
non limited terminated VPC	dedicated	SB_{LC}	none	none
limited terminated VPC (mixed traffic)	none	SB_{VPC}	PCR_{VPC}	none
limited terminated VPC (pure rt or UBR)	dedicated PCR limited	SB_{LC}	PCR_{VPC}	none
non terminated VCC terminated VCC	dedicated	see related VPC	GCR_{VCC}	required
AAL1 channel	shared	SB_{AAL1}	PCR	none
AAL2 path	dedicated	see related VPC	GCR_{VCC}	required

3.7 Operation, Administration and Maintenance

The OAM unit in IWORX covers F4/F5 OAM flow functions according to the ITU I.610 and Bellcore GR-1248 standards covering the functions Fault Management (AIS, RDI, CC, LB) and Performance Monitoring (FPM flow, BR flow, Data Collection). It has STM-4/OC-12 equivalent throughput in RX and TX direction.

The AIS, RDI, CC mechanism can be applied to a range of up to 256 connections. Performance Monitoring can be done for 2 connections simultaneously with each connection selectable from RX or TX direction. Loopback functionality can be applied to 8 selectable connections.

3.7.1 Functional Block Diagram

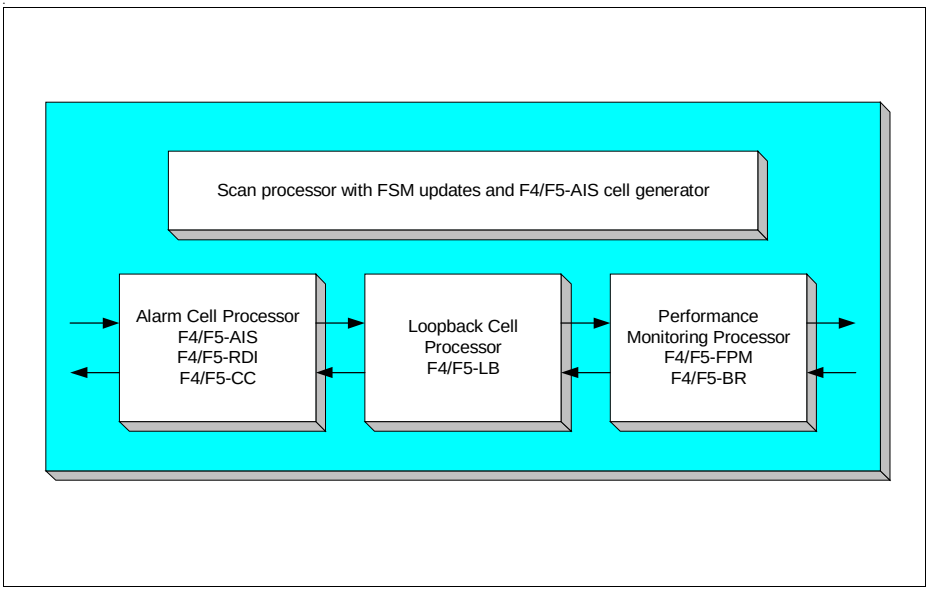


Figure 56 OAM Unit Block Diagram

CONFIDENTIAL

3.7.2 Feature List

Parameters

- 256 connections in both directions (VPC/VCC)
- 2 performance monitoring processors

OAM Functions

- OAM Levels and Flows (F4/F5) according to ITU-T I.610 [11/95] and Bellcore GR-1248
- Generation, discard, extraction and insertion of OAM cells
- Generation of fault management cells at line failure detected by the connected framer device and activated by a message

AIS/RDI Functions

- AIS/RDI function available for all connections
- Programmable guard times and cell insertion intervals
- e-t-e F4/F5 AIS/RDI generation and detection
- segment F4/F5 AIS/RDI generation and detection not supported acc. to 95' I.610
- Non-intrusive monitoring F4/F5 AIS/RDI detection/copy for intermediate points

CC Functions

- CC function available for all connections
- Programmable guard times and cell insertion intervals
- Support of CC activation/deactivation cells
- e-t-e and segment CC detection and generation
  CC cells are sent in the forward direction when no user cells have been sent for a period of nominally 1 second.
- Non-intrusive monitoring CC F4/F5 detection/copy for intermediate points not supported acc. 95' I.610

Loopback

- Loopback functionality available for 8 selectable connections
- e-t-e, segment and intra-domain loopback support per LB ID
- Programmable Port ID for on the fly comparison with Location ID or Source ID of LB cells

Performance Monitoring

- 2 simultaneous PM generation/evaluation processors shared for RX and TX direction
- Evaluation of FPM cells and generation of BR cells

- Support of data collection according to Bellcore GR-1248 by 2 data collection processors, independently direction assignable
- Support of PM activation/deactivation cells
- Support of simultaneous PM flows of F4 and F5 level
- Support of adjacent PM segments within IWORX OAM unit
- Collection of data, e.g. severely errored blocks, total lost cells (see appendix of the Programmer's Reference Manual)

3.7.3 Functional Description

3.7.3.1 Cell Handling

Each cell entering the OAM unit is identified as user cell or as OAM cell. The chip recognizes all standardized OAM cells. Context data stored on a per connection basis determines if a connection is enabled and which layer point is configured. Accordingly the respective function is performed.

- For example an e-t-e F4-AIS cell would be forwarded transparently at a VPC segment endpoint.
- As another example a user cell belonging to a VPC for which end-to-end performance monitoring is enabled is counted and its checksum (BIP-16) added to the checksum in the OAM unit located at the VPC endpoint.

In the respective OAM processing block new status information is calculated, for example alarm indication bits, BIP-16 checksums, cell counts etc.

Whereas user cells are never modified and are always forwarded, OAM cells can be

- generated and inserted into the cell stream in RX or TX direction
- extracted from the cell stream and discarded or dropped to a receive buffer
- forwarded with or without modification
- looped with modification.

For OAM cell generation the OAM unit uses the configuration parameters of the respective connection to determine the OAM cell type: F4 or F5 and segment or end-to-end.

When detecting OAM cells the OAM unit recognizes F4 or F5 OAM cells for end-to-end or segment. According to the configuration the required actions are performed.

3.7.3.2 OAM Functions Overview

There are two groups of applications for OAM functions: alarms and measurements. Alarm functions inform user and network operator about network failures. These include the OAM functions

Alarm Indication Signal AIS

Remote Defect Indication RDI

Continuity Check CC.

AIS and RDI are used to convey transmission line failure information to subscriber and network operator. CC is used to detect ATM layer failures.

Measurements are initiated for diagnosis purpose by the network operator. Therefore these functions need not be permanently active for all connections. The respective OAM functions are:

Loopback LB

Performance Monitoring PM

LB checks the connectivity of a connection by sending a single cell which is looped back at predefined points. LB is used e.g. immediately after connection set-up or periodically to check all permanent connections of a network using end-to-end or segment LB. A network operator could also use intra-domain LB to localize a failed link. Another option for loopback are subscriber initiated loops either end-to-end to the partner or access line LB to the first node in the network.

PM is a more precise tool than LB. It checks not only the connectivity, but the real performance of a connection in terms of bit failures and cell losses. VPCs or permanent VCCs could be monitored if a subscriber pays for this service. Also a network operator would use PM to check the quality of a connection if a subscriber complains it.

3.7.3.3 Alarm OAM Functions (AIS/RDI/CC)

There are two types of failures detected by the alarm functions: transmission line failures and ATM layer failures. Transmission line failures are e.g. line brakes, failures of lasers or failures of reception diodes. Typical ATM layer failures are the misrouting of cells in the switching fabric or a falsified entry in a routing table. All the cells of a connection are then forwarded to a wrong destination.

Transmission Line Failures (AIS/RDI)

Transmission line failures are recognized by the receiving physical layer device, e.g. QuadFALC, and conveyed to the OAM unit by the on-board control processor. It is sufficient to set one single parameter for the respective PHY to initiate the periodic insertion of AIS cells for all affected connections. The OAM unit inserts F4-AIS cells for VPCs and F5-AIS cells for VCCs. **Figure 57** shows that this case occurs at the incoming port of a switch.

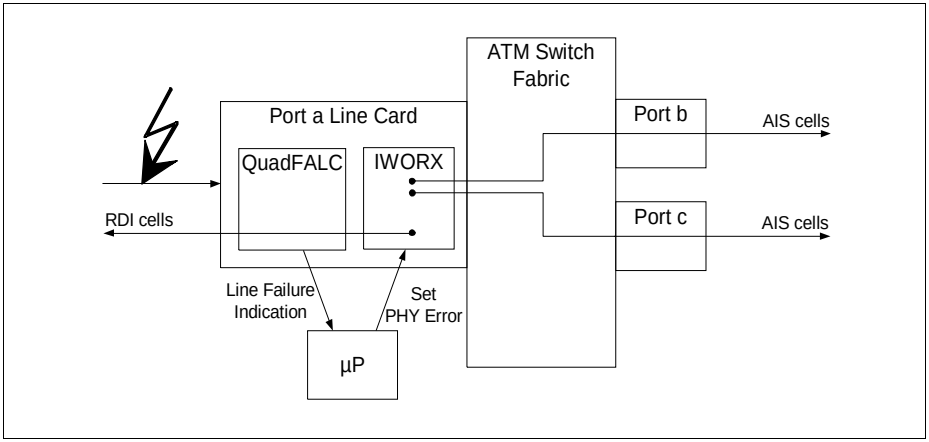


Figure 57 Example for Line Failure Notification via AIS/RDI Cells

The generated F5-AIS cells travel up to the endpoint of the connection, which normally is the user terminal. Thus within a very short time delay - determined by the control processor's response time, the OAM cell insertion delay and the cell transfer time - the user is informed about the failure.

The generated F4-AIS cells travel up to the VPC terminating endpoint which normally is within the network. At the VPC terminating endpoint F5-AIS cells must be generated for all VCCs contained in the VPC. Again, all affected user terminals are informed.

The OAM unit performs the following actions when receiving e-t-e F4-AIS cells at a VPC terminating endpoint:

- Discard of e-t-e F4-AIS cells
- Execution of the AIS state diagram shown in **Figure 58** for the respective VPC.
- Insertion of e-t-e F4-RDI cells in backward direction. This measure informs the originating endpoint of the VPC about the failure. RDI makes sense in the cases where the failure of the line affects only one direction. The RDI generation in backward direction assumes bi-directional connections with the same identifier (connection handle) in both directions.
- Insertion of e-t-e F5-AIS cells in forward direction for each VCC of this VPC. This measure informs all users sharing this VPC about the failure in the network.
- Declaration of AIS/RDI failure states after 3.5 s (standard) persistence of AIS defect state. The cell insertions continue unaffected.

At the originating endpoint of the VPC the following actions are perfomed:

- Discard of e-t-e F4-RDI cells
- Execution of the RDI state diagram shown in **Figure 59**.
- Declaration of RDI failure state after 3.5 s (standard) persistence of RDI defect state.

CONFIDENTIAL

The AIS state diagram executed at the TEP of a connection is shown in **Figure 58**. Note that in accordance with the anomaly-defect-failure mechanism only transitions to and from failure state are notified to the external controller. This avoids unnecessary interrupts.

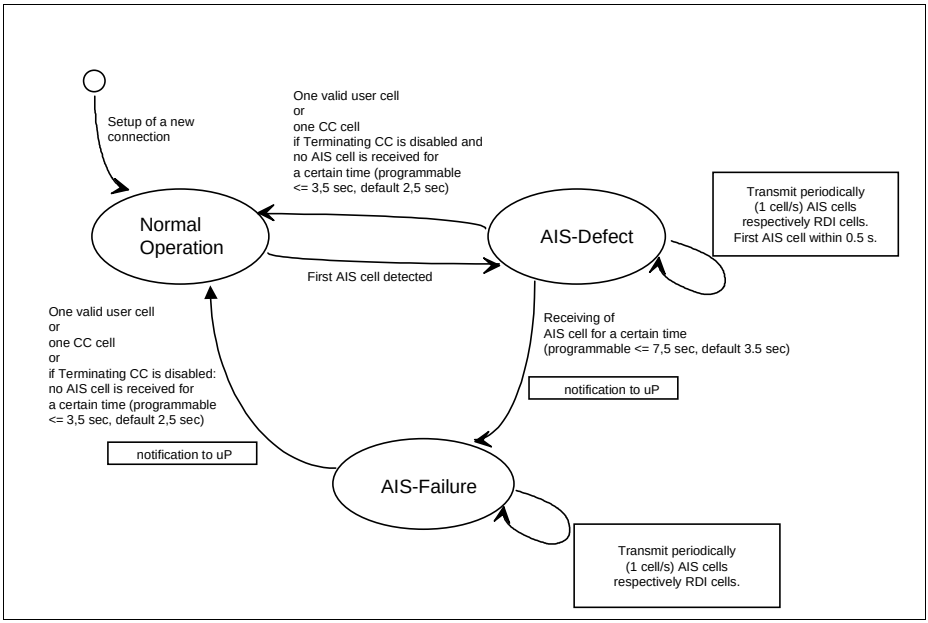


Figure 58 AIS State Diagram

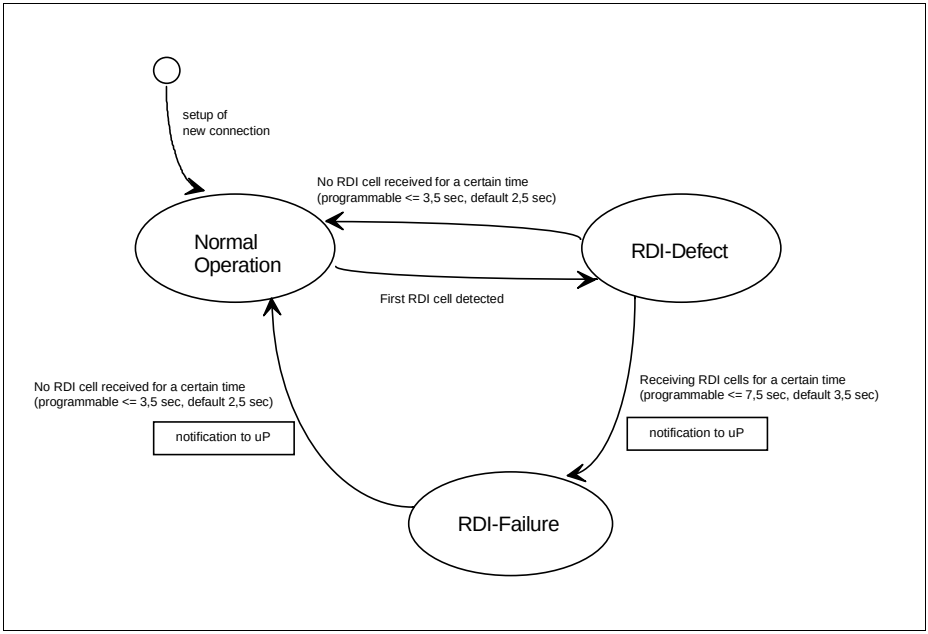


Figure 59 RDI State Diagram

Both forward and backward cell insertions are initiated by the SCAN mechanism (see **Section 3.7.3.7**). All delay times given are default values, recommended by the standard. The OAM unit allows to change these default values in multiples of the 0.5 s.

The 0.5 s SCAN period determines the insertion delay for OAM cells. If the SCAN mechanism has passed a connection entry just before an AIS condition became true the maximum waiting time for the next SCAN access is 0.5 s.

ATM Layer Failures (CC)

The mechanism to detect failures like misrouting is the Continuity Check (CC). Its idea is to insert additional cells in a connection if it is inactive, i.e. if the user is not sending data cells. The additional cells are called CC OAM cells and are inserted at the originating e-t-e/segment point of a connection after a 1-second absence of user cells. The repetition interval is one second. At the terminating segment or e-t-e point the CC cells are discarded. If no user or OAM cells are received within 3.5 seconds the Loss of Continuity (LOC) defect state is declared. Like AIS state LOC causes the insertion of F4-AIS or F5-AIS cells for the affected connections. If LOC is detected at a terminating endpoint RDI cells are generated in backward direction.

CONFIDENTIAL

Figure 60 shows an example for the operation of CC: two VCCs entering a switch at ports a and b should both be forwarded to port c. Due to misrouting within the switching fabric the cells of VCC b are forwarded to an unconnected switch output, where they are lost without being notified. The CC detection function at port c, however, detects the absence of user cells after the 3.5 s time-out and inserts F5-AIS cells for the connection b.

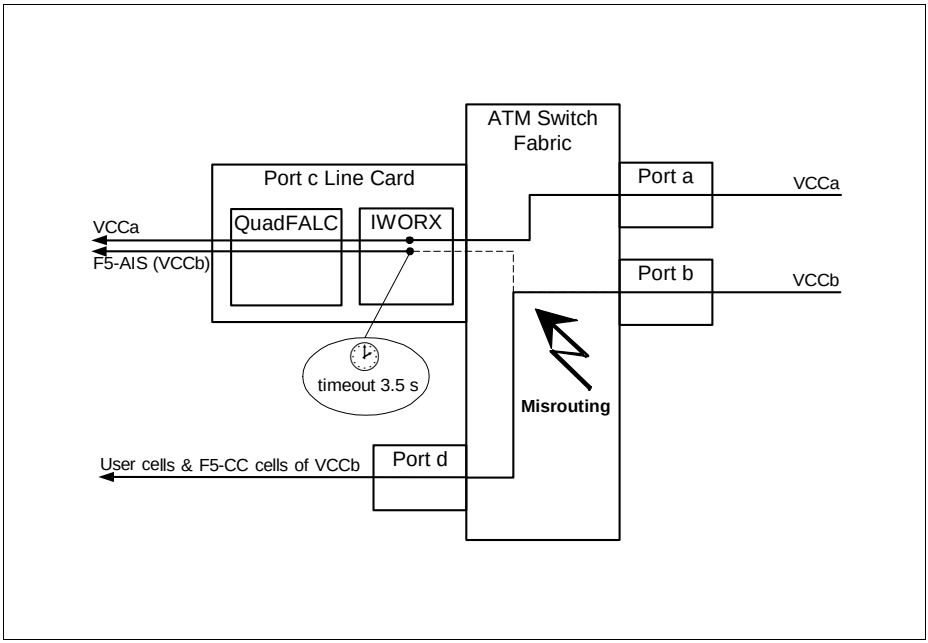


Figure 60 Example for Misrouting Failure Detection

The time values given are values recommended in the standard. The OAM unit allows to change them in a wide range.

The OAM unit supports the CC function for all 256 connections in both RX and TX direction. Setting one connection specific parameter activates the origination or the termination of a CC flow.

Actions at the CC origination point (see **Figure 61**):

- continuous supervision of user cell stream
- periodic insertion of CC cells in 1 s intervals after 1.0 s (standard) time-out

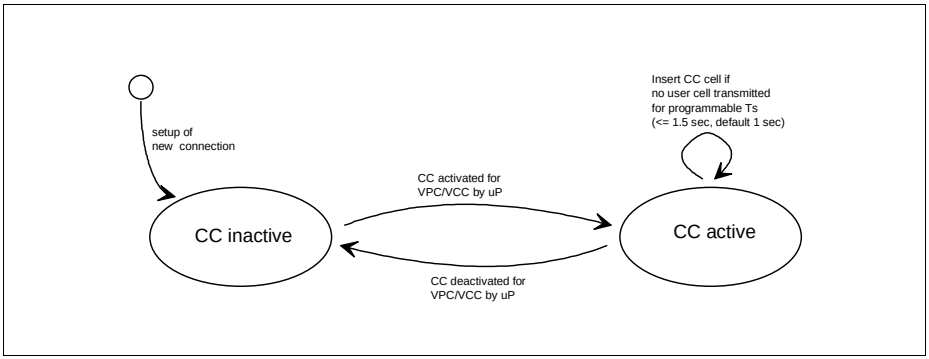


Figure 61 Continuity Check Cell Generation State Diagram

Actions at the CC termination point (see **Figure 62**):

- discard of CC cells
- declaration of LOC defect state and insertion of AIS cells in 1 s intervals after 3.5 s absence of user or OAM cells
- declaration of LOC failure state if LOC defect state persists for 2.5 s.

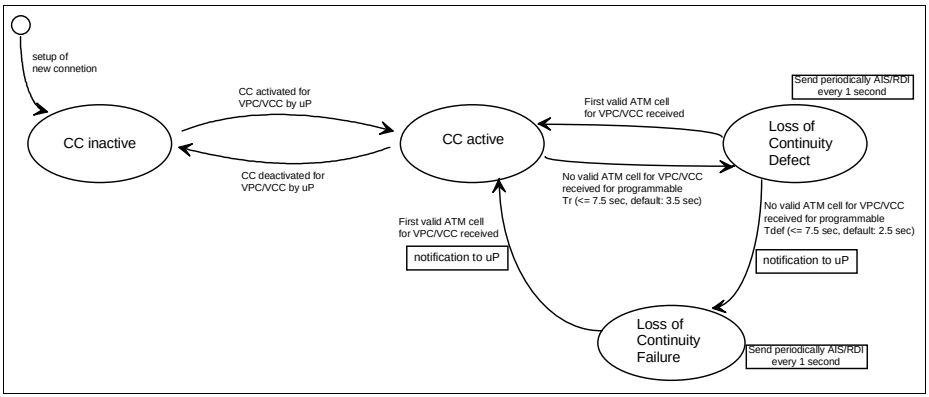


Figure 62 Continuity Check Evaluation State Diagram

The LOC failure state is notified to the control processor, while still AIS cells are generated. This additional filtering according to the standards avoids frequent notifications to the microprocessor due to sporadic errors.

All cell insertions are initiated by the SCAN mechanism (see **Section 3.7.3.7**). The delay times given are default values, recommended by ITU-T I.610. The OAM unit allows to program these values in multiples of 0.5 seconds.

The insertion of AIS cells occurs as in the AIS state, i.e. with periodic insertion of e-t-e F4-AIS or e-t-e F5-AIS cells and e-t-e F4-RDI cells in backward direction. No additional AIS cells are inserted if AIS and LOC state are declared simultaneously.

3.7.3.4 Network Connectivity Check (LB)

General

The loopback (LB) OAM function is intended for checking the connectivity of a virtual connection by sending a single LB cell along the connection. The LB cell is extracted at well defined points of the network and sent back to the source via the backward connection. Note that each ATM connection has an associated connection in backward direction with the same connection identifiers.

There are three possibilities for specifying the loopback point of an LB cell:

1. End-to-end LB cells: these cells are looped at the endpoint of a connection. For a VCC this is normally the sink terminal, but can also be the AAL within the IWORX where the ATM connection ends and the conversion to time slots occurs. For VPCs the endpoint normally is within the ATM network.
2. Segment LB cells: these cells are looped at segment endpoints which are typically at boundaries between network operators. Thus segment LB cells are used to check the connectivity within an operators network. If an operator switches a VPC from one port to another, F4 segment LB cells are used for connectivity check, if a VCC is switched F5 segment LB cells will be used.
3. The third option for LB cells are intra-network LB cells. These are used on F4 and F5 level to check the connectivity within an operators network from node to node. These cells carry the 16-octet location identifier and the 16-octet source identifier. In forward direction the Loopback Location ID of the LB cell is compared with the Port ID programmed into the OAM unit. In backward direction the Source ID of the LB cell is compared with the Port ID. Forward and backward direction LB cells are distinguished by the LB indication bit (LSB of LB indication octet).

LB Support

The OAM unit supports LB in following way:

- Cell insertion and extraction buffers
- Detection and loop of LB cells at loopback point with inversion of LB indication bit
- Detection and extraction of LB cells back at the originating point.

Loop of LB cells with reset of the LB indication bit in the cell is done without microprocessor interaction at the respective segment or connection end points. Note that the loop function assumes identical connection identifiers for both forward and backward connections.

CONFIDENTIAL

3.7.3.5 Connection Quality Measurement (PM)

General

The PM function is split into three different parts:

- PM generation
- PM analysis and loop
- PM data collection.

PM generation includes

- Calculation of total user cell count for all cells and for high priority cells (CLP=0)
- Calculation of BIP-16 checksum over user cell payload
- Generation of FPM cells containing the calculated results.

The FPM cells are coded as F4 or F5 automatically for VPCs and VCCs, respectively, end-end or segment as specified. FPM cell sequence number and CRC-10 checksum are also generated. Block size can be selected between 2 and 65536. The optional time stamp of the FPM cell is not generated.

Forced OAM cell insertion is used for both RX and TX FPM cell insertion. During insertion of FPM cells the user cell stream is stored in the respective buffers.

PM analysis and loop include

- Calculation of total user cell count for all cells and for high priority cells (CLP=0)
- Extraction of FPM cells
- Appending of calculation results at the end of the cell
- Conversion of the cell into a BR cell
- Reinsertion of the BR cell in opposite direction.

PM analysis uses the same PM processor circuits as generation. 2 PM processor circuits are shared by RX and TX direction.

For PM data collection 2 circuits are provided, which are independent of the PM processor circuits. Both performace monitoring PM and data collection DC processors have their respective entries in the internal PM/DC RAMs.

The assignment of PM and DC processors to connections of RX or TX direction is arbitrary. VPCs and VCCs are assigned by pointers in the internal F4 and F5 RAM entries, respectively.

Example

A typical PM scenario is shown in **Figure 63** for the case of VPC e-t-e monitoring. Two nodes are involved, Node a where the VPC_{a-b} is originated and Node b where VPC_{a-b} is terminated. In backward direction the associated VPC_{b-a} is originated in Node b and terminated in Node a. The Originating e-t-e Point OEP of VPC_{a-b} is located in the TX part of the OAM unit in Node a, and the Terminating e-t-e Point TEP of VPC_{a-b} is located in

the RX part of the OAM unit in Node b. For VPC_{b-a} the situation is mirrored according to **Figure 63**.

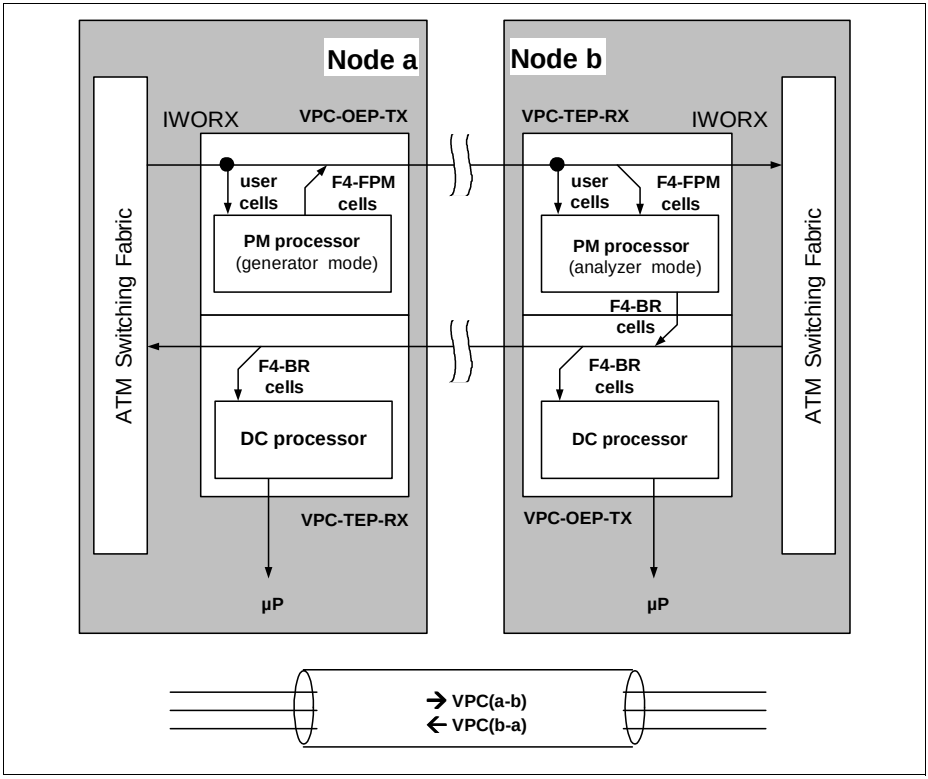


Figure 63 PM Configuration Example

Note that between Nodes a and b a number of intermediate nodes can be located. All chips with an OAM unit on these nodes must be configured either as Originating or Terminating Segment Points (OSP, TSP) or as Intermediate Points (IP).

One PM processors in the OAM unit RX part of Node a is configured in generator mode, i.e. it monitors all user cells of VPC_{a-b}, computes PM data and inserts it after blocks of user cells into the cell stream as Forward Performance Monitoring FPM cells.

At the terminating OAM unit one PM processors is configured in analyzer mode, i.e. it monitors all user cells of VPC_{a-b}, computes PM data and adds it to the PM data contained in the FPM cells. The FPM cells are extracted from the cell stream, converted into Backward Reporting BR cells and reinserted in backward direction. The conversion

into BR cells includes the calculation of the differences between measured PM data and the PM data contained in the FPM cells. The differences are written into the BR cells.

Back at the originating Node a, the BR cells are discarded after evaluation.

Note that the reinsertion of BR cells in backward direction assumes the same identifier (connection handle) of the backward direction connection.

Data Collection

Independent on the FPM/BR cell mechanism is the data collection procedure. It uses one of the two DC processors contained in the OAM unit. Each of them can evaluate the BR data flow from RX or TX direction. Data collection can be done at any node along the way of the BR cells. In the example of **Figure 63** Nodes a or b could be selected for data collection. The OAM unit uses either the BR cells incoming from the UTOPIA interface or the locally looped BR cells for data collection.

Simultaneous PM Flows

The OAM unit contains two PM processors which may be used to generate a PM flow or to terminate a PM flow. Terminating an PM flow means analyzing and loop of the FPM cells as BR cells. During one cell cycle **two** PM processors can be executed for one single cell, arbitrary for F4 and F5 level.

It may happen that a user cell belongs to a VCC for which F5 segment PM is done. E.g. in the example of **Figure 63** node b could be a VCC Originating Segment Point OSP in addition to the VPC TEP. Then the arrival of a VCC user cell triggers two PM processors in the RX part of the OAM unit.

In the case of F4 and F5 segments e.g. the TX part of the OAM unit could be configured as VPC OSP and VCC OSP. In this case a user cell not only triggers two PM processors simultaneously, but might also complete two PM blocks. Then two FPM cells have to be generated simultaneously. In this case the OAM unit first inserts the F4-FPM cell and then the F5-FPM cell.

Adjacent PM Segments

The arbitrary assignment of PM processors to connections also allows e.g. to terminate a Segment PM flow and generate a new Segment PM flow for the same connection within one OAM unit as shown in **Figure 64**.

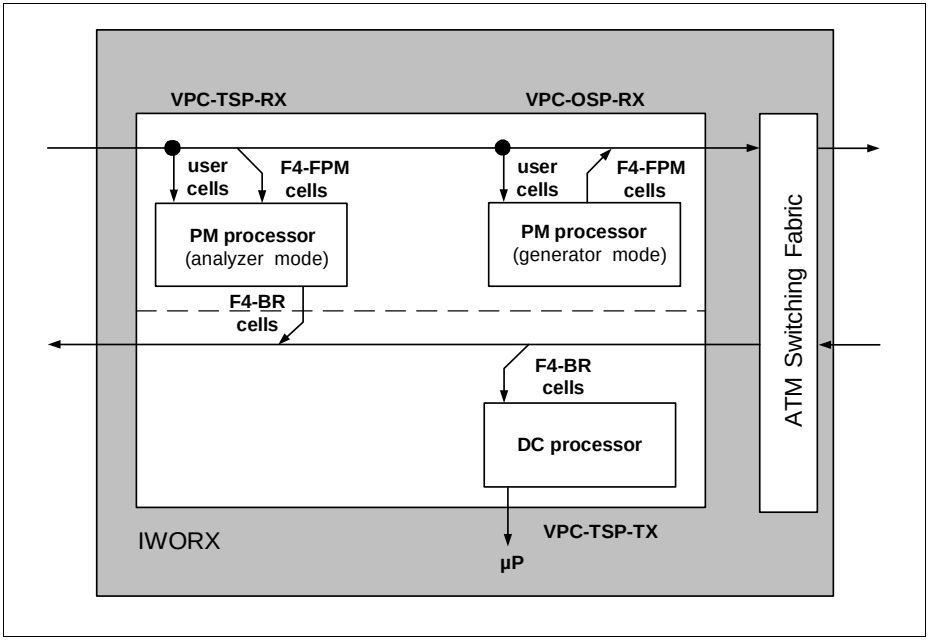


Figure 64 Example for Adjacent PM Segments

3.7.3.6 Activation and Deactivation Cells

These cell types are generated by the external microcontroller and transmitted via the cell insertion function of the OAM unit.

The detection and extraction of activation/deactivation cells is done at the respective segment or end-to-end points if the respective feature of the OAM unit is enabled (which is possible per connection).

3.7.3.7 SCAN Mechanism

The SCAN mechanism is automatically triggered in a 0.5 s time period, as all time-out values are determined based on this time interval. During a SCAN all entries within the specified range of the internal RAMs are accessed sequentially. The SCAN is programmed in such a way that it covers in a little less than 0.5 s all used RAM entries.

With each SCAN trigger OAM functions can be enabled independently for RX and TX direction. OAM functions include all the necessary actions for AIS/RDI/CC processing, i.e. update of counters, check for time-out values and execution of state transitions, OAM cell insertions.

3.7.4 Layer-Point Concept

This concept is introduced to enable the execution of OAM functions by the OAM unit. For each connection, VPC or VCC the layer point is configured at connection setup. Then the OAM functions required for this layer point are executed automatically by the OAM unit. There are 3 different basic layer points:

- End-to-end point EP
- Segment point SP
- Intermediate point IP

EP and SP can be originating or terminating, can belong to a VPC or a VCC and can belong to RX or TX direction. They are referenced as explained in **Chapter 1.6**, e.g. VPC originating end-to-end point in transmit direction VPC-OEP-TX or VCC terminating segment point in receive direction VCC-TSP-RX.

As an example a TSP would just forward arriving e-t-e AIS cells transparently, as e-t-e AIS cells have always the end-to-end identification. The same layer point would loop arriving forward segment LB cells.

In the following scenarios examples for layer point configurations are shown. In these figures a switch with its incoming and outgoing port is represented by the symbol shown in **Figure 65**.

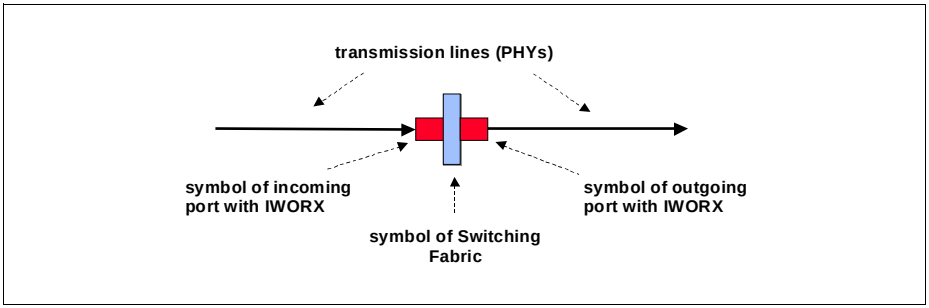


Figure 65 Symbol for Switch with OAM Units

Within a pure ATM network VPCs may be originated or terminated. In addition VP segments can be originated or terminated as shown in **Figure 66**.

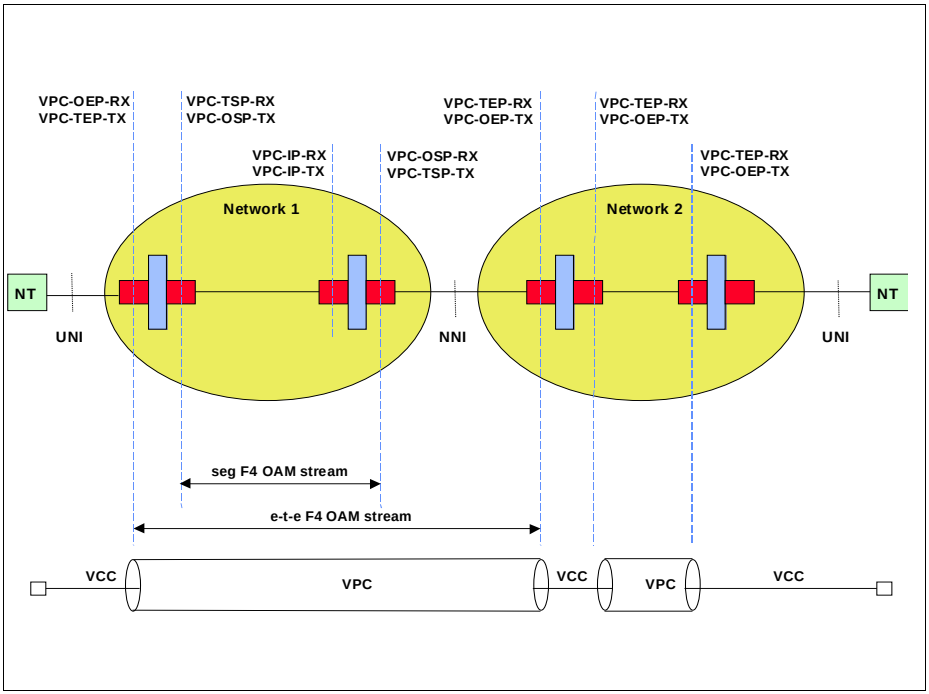


Figure 66 F4 Level OAM Functions

VCCs are not originated or terminated within a pure ATM network, but only VC segments (**Figure 67**).

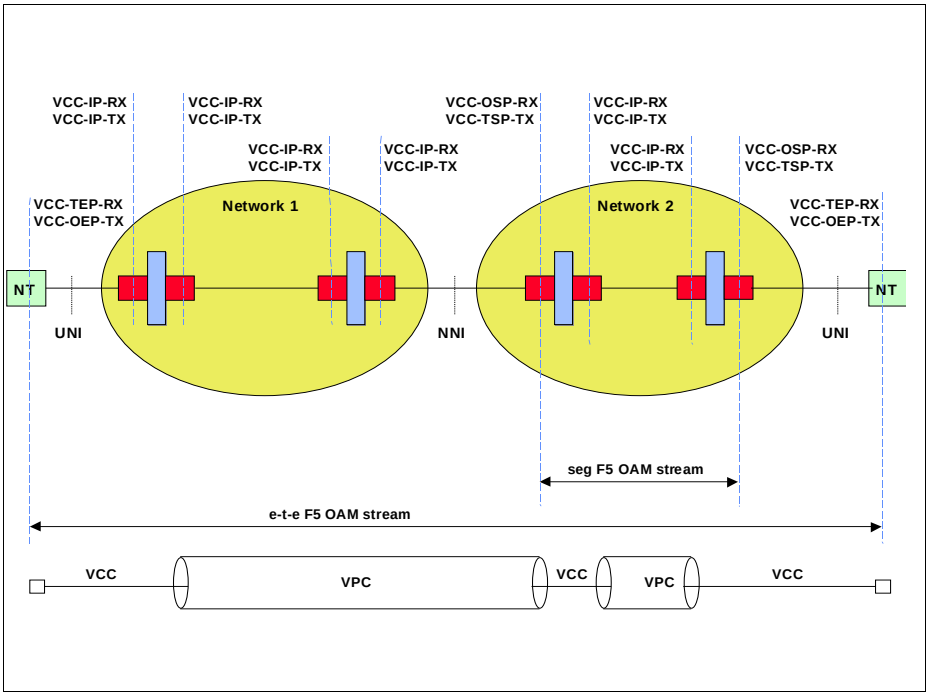


Figure 67 F5 Level OAM Functions

In a heterogeneous network containing ATM and non-ATM interfaces VCC origination or termination occurs at the AAL function as shown in **Figure 68**.

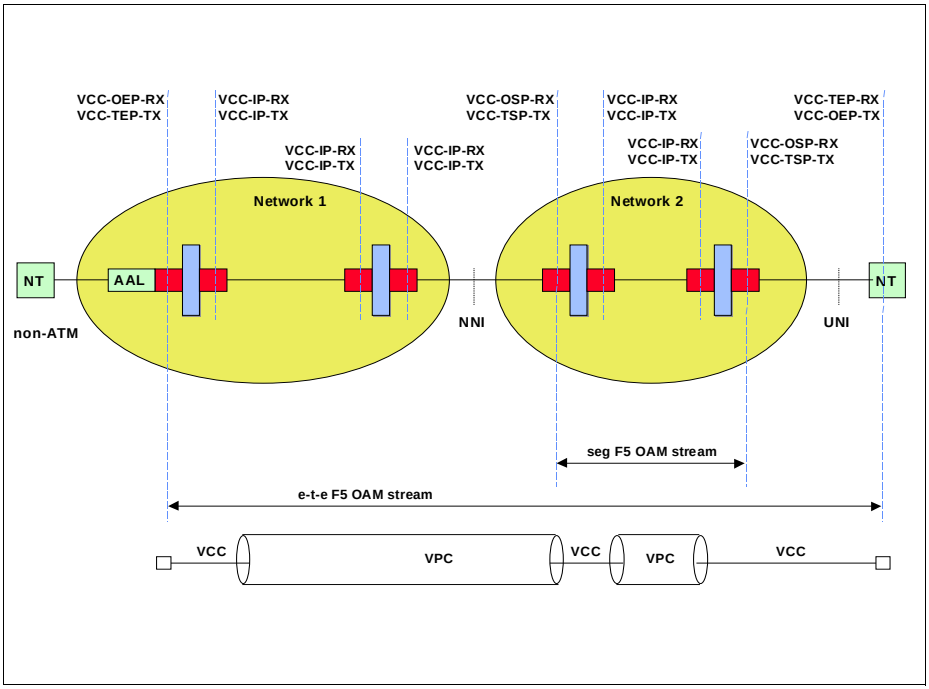


Figure 68 VCC EP Inside the Network

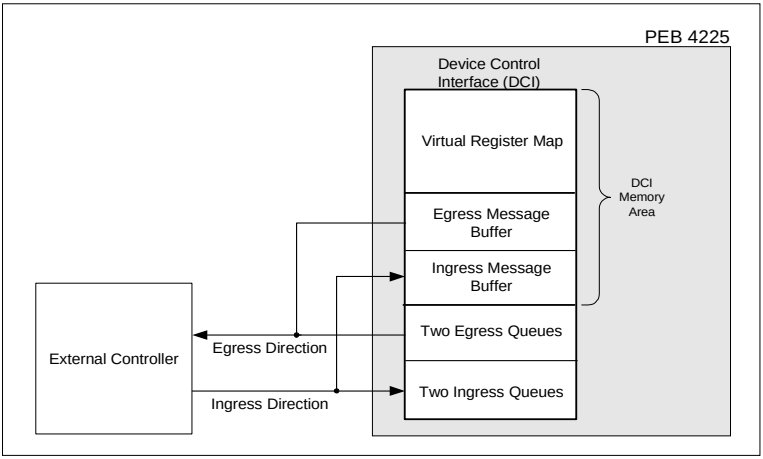
The OAM unit terminates the OAM cell stream in TX direction just before the AAL device which terminates the ATM connections.

4 Operational Description

The Device Control Interface (DCI) is the interface for the host processor to communicate with the device. This communication is message based. The DCI can be seen more or less like a mailbox for the communication between the host processor and the device. After the host processor has written a message into the DCI the device internal processor evaluates it and performs the necessary internal actions. The DCI is a memory and register area which is accessible for the user. The memory area is 32K bytes and the address range is from 8000_H to $FFFF_H$. Furthermore the DCI register area contains interrupt status and mask registers where up to 32 different interrupts are indicated.

Inside the DCI there is a logical structure of message queues that are used for communication between the device and the host processor. There are four message queues for different directions and priorities. Messages from the external controller are written to the ingress queues (see Figure 69). These messages are used for controlling and configuration of the device. The messages to the egress queues are used for acknowledgements, progress indications, notifications and alarms.

Figure 69 Message Flow of the DCI



The characteristics of the different message queues, message formats and message types as well as a detailed overview and description of all existing messages are content of the IWORX document *Programmer's Reference Manual*. Every message and its corresponding information (message id, message parameters) is listed there.

5 Interface Description

5.1 PDH Interface

Dependent on the selected PDH port mode a link can be operated in T1 or E1 mode. Unchannelized operation is provided for unstructured CES.

The PDH interface consists of one receive clock input, a receive frame/multiframe synchronization pulse input and one receive data input for each receive line. For CAS freezing the IWORX provides additionally the signal PRFREEZE. In transmit direction each link consists of a transmit clock, one transmit frame/multiframe synchronization signal and one transmit data output. Synchronization pulses are not supported in unchannelized mode. Additionally one common transmit clock input and common transmit frame synchronization input are provided.

A variety of loop modes is provided to support ingress as well as egress loop testing.

Following signalling formats are supported:

- T1 mode: ANSI T1.403-1995 (bit robbed signalling) in SF (D4) and ESF multiframes
- E1 mode: CAS signalling in time slot 16 according to ITU-T G.704.

In receive direction any time slot may be delayed in the range of 0 to two frames to compensate for external switching delays. To determine the external switching delay the IWORX provides the capability to dump frames of a selected port.

Furthermore the IWORX provides mapping of any external port/time slot combinations to logical channels. A maximum of 256 logical channels is supported.

Functional List of PDH Interface Block

- The serial interface provides eight interface ports and one additional test port.
- Interface ports can be configured for T1 (1.544 MHz) or E1 (2.048 MHz)
- All interface ports provide frame/multiframe synchronization.
- Concatenation of any, not necessarily consecutive, time slots to logical channels on each physical link. A maximum of 256 logical channels is supported.
- Dumping of frames in receive direction is supported. One port at a time can be dumped.
- For each received time slot a variable delay of 0, 1 or two frames can be programmed.
- Serial data/sync/msync sampling/transmitting with rising or falling edge of clocks
- Frame/Multiframe synchronization on rising or falling edge of frame sync pulse
- Programmable PTD/PRD bitshift of +3/-4 bits relative to the sync pulse
- Test port function which allows to monitor and replace a selected link.
- Ingress/Egress Loop per Port

5.1.1 Interface Mode E1 and T1

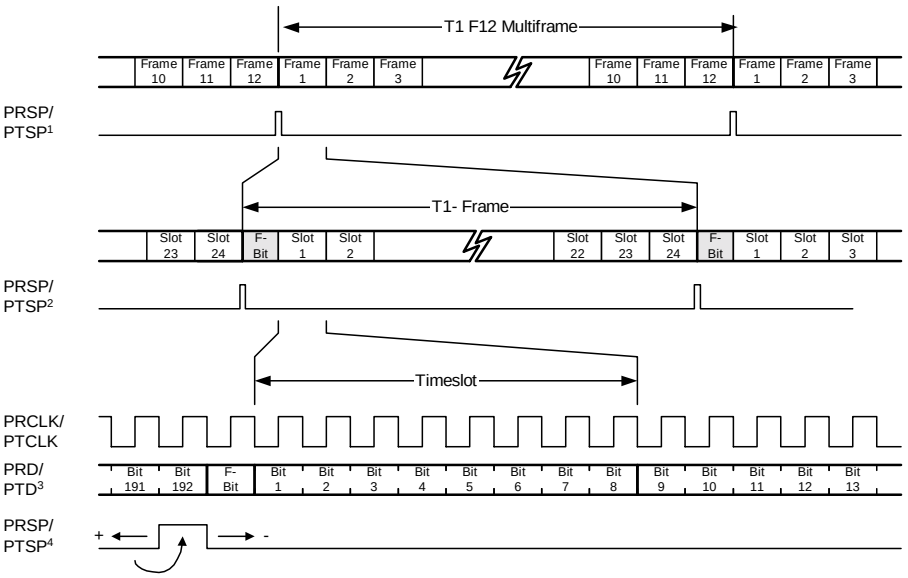
The PDH receive interface is realized by the interface pins PRD/PTD and the frame marker functions PRSP/PTSP. The PDH receive interface is clocked via pin PRCLK(x), while the PDH transmit interface is independently clocked via pin PTCLK(x), independent for each PDH port. The frequency of the working clock, i.e. 1.544/2.048 for the receive and transmit PDH interface is programmable.

The data and sync pulses on the PDH interface can be latched resp. clocked off on the rising or falling clock edge.

The signals on pin PRSP together with the assigned bit shift defines the beginning of a frame on the receive interface.The signal on pin PTSP together with the assigned bitshift define the beginning of a frame on the transmit side. When bit shift is set to zero PTSP marks the last bit of a frame/multiframe while PRSP marks the first bit of a frame/ multiframe.

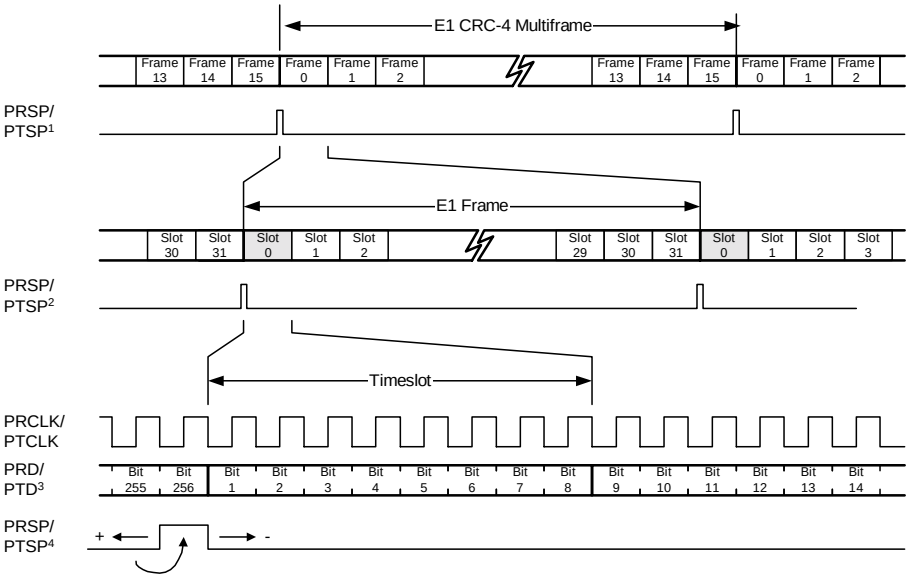
Adjusting the frame/multiframe boundary relative to PTSP/PRSP is possible in the range of -4/+3 bits.

The principal configurations and functions of the PDH interface in DS1 mode are shown in **Figure 70**. PDH interface in E1mode is shown in **Figure 71**.



Note:
1. PRSP/PTSP operated as multiframe synchronization pulse
2. PRSP/PTSP operated as frame synchronization pulse
3. Sampling/Update of PRD/PTD on rising or falling edge of PRCLK/PTCLK is programmable
4. Synchronization pulse can be shifted in the range of -4 to +3 with respect to default position.
   Default position (shift value of 0) is at the F-bit in receive direction/bit 192 in transmit direction.
   PTSP can be operated as input or output.
   PRSP/PTSP can be sampled/updated on rising or falling edge of PRCLK/PTCLK.
   The trigger edge can be programmed as active high or active low.

Figure 70 PDH Interface in T1 Mode



Note:
1. PRSP/PTSP operated as multiframe synchronization pulse
2. PRSP/PTSP operated as frame synchronization pulse
3. Sampling/Update of PRD/PTD on rising or falling edge of PRCLK/PTCLK is programmable
4. Synchronization pulse can be shifted in the range of -4 to +3 with respect to default position.
   Default position (shift value of 0) is at bit 1 in receive direction/bit 256 in transmit direction.
   PTSP can be operated as input or output.
   PRSP/PTSP can be sampled/updated on rising or falling edge of PRCLK/PTCLK.
   The trigger edge can be programmed as active high or active low.

Figure 71 PDH Interface in E1 mode

5.1.2 Connection to a Framer Device

The diagram shows the glueless connection of the device IWORX to the framer device QuadFALC.

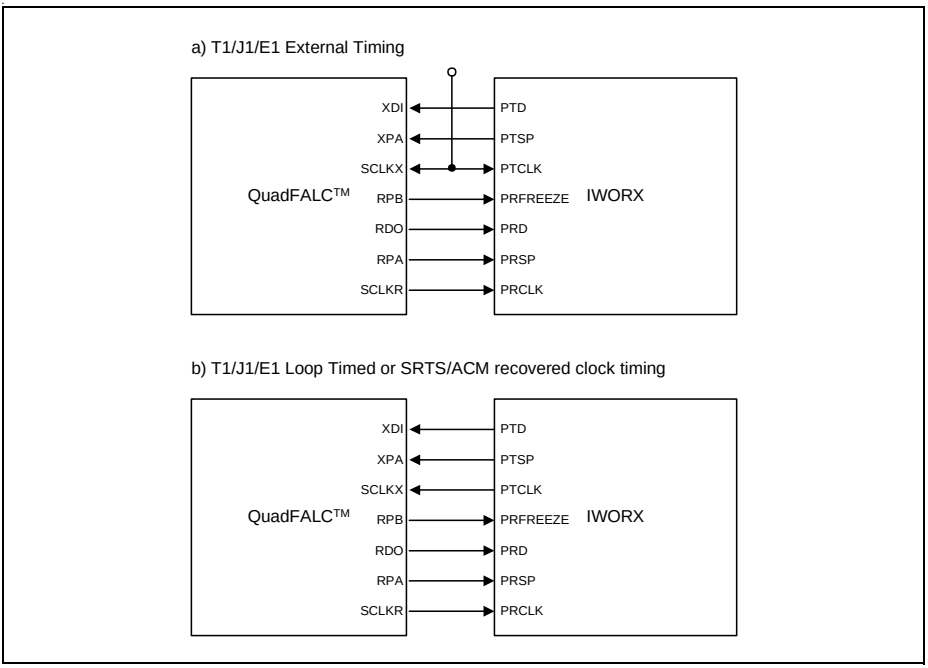


Figure 72 IWORX-QuadFALC Connection for E1 or T1

5.1.3 Clocking Modes

Each receive PDH port of the IWORX can be clocked with the corresponding receive clock PRCLK(x) or with the common transmit clock PCTCLK. The corresponding sync pulse signals are shown in **Figure 74**.

Each transmit clock of the IWORX can be operated in the following modes (for selectable sync pulse signals see **Figure 75**):

Common External Clocking Mode

In this mode the common transmit clock input PCTCLK is used to clock selected transmit ports. In this mode PTCLK(x) is an output.

External Clocking Mode

In external clocking mode the clock input PTCLK(x) is used to operate the corresponding transmit line. In this mode PTCLK(x) is an input.

Internal Clocking Mode

In internal clocking mode the transmit clock PTCLK(x) is a recovered clock according to SRTS or ACM clock recovery method or an internal generated clock phase-locked with the clock at pin XTAL1. The correct frequency - 2.048 MHz for E1 and 1.544 MHz for T1 - is generated automatically and independent from the XTAL1 signal by user selection if the PDH interface shall work in E1 or T1 mode. In this mode PTCLK(x) is an output.

Loop Timed Clocking Mode

In loop timed clocking mode the transmit clock PTCLK(x) is derived from the receive clock PRCLK(x). The looped clock is output on port PTCLK(x).

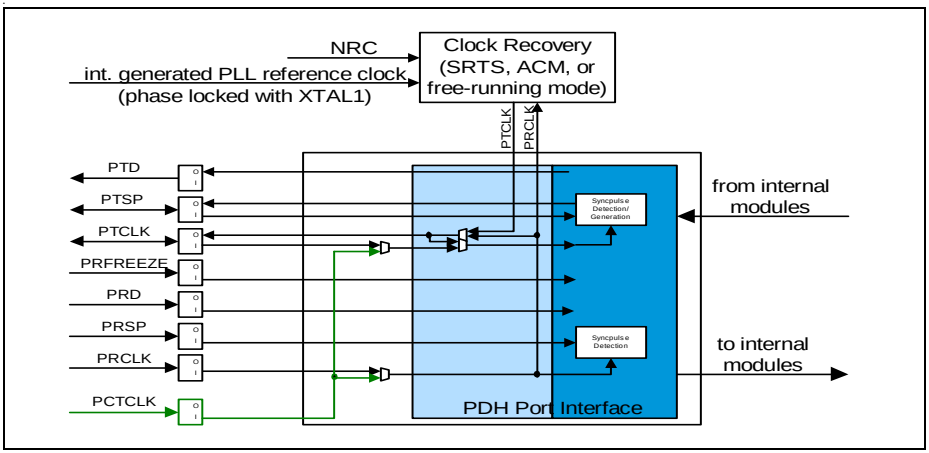


Figure 73 PDH Clocking Modes

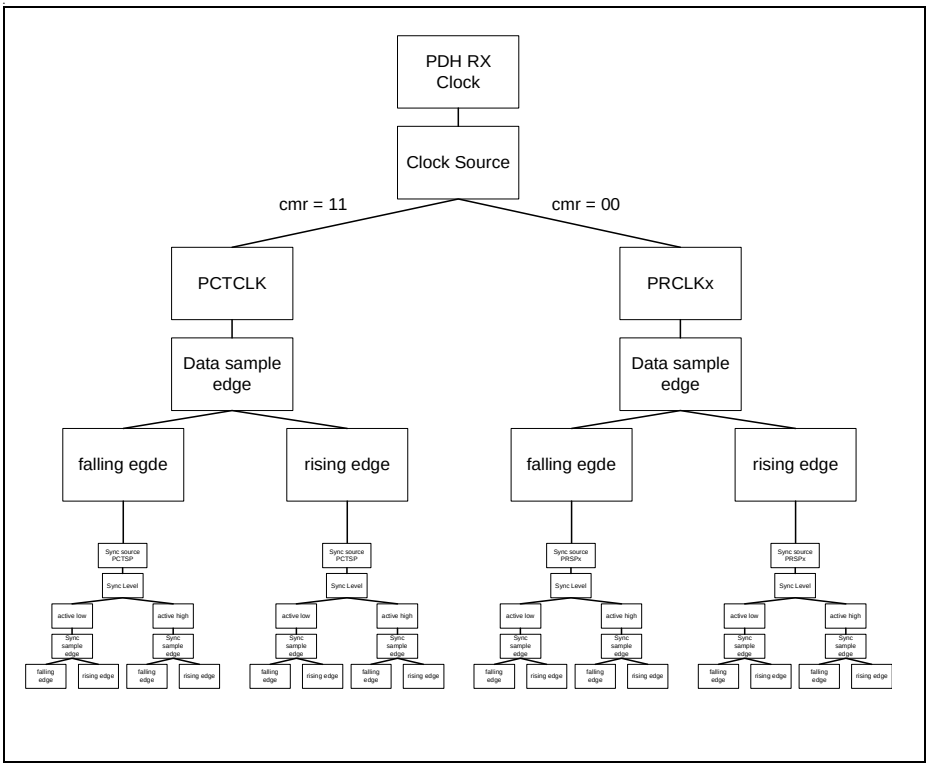


Figure 74 PDH Receive Clocking Configurations

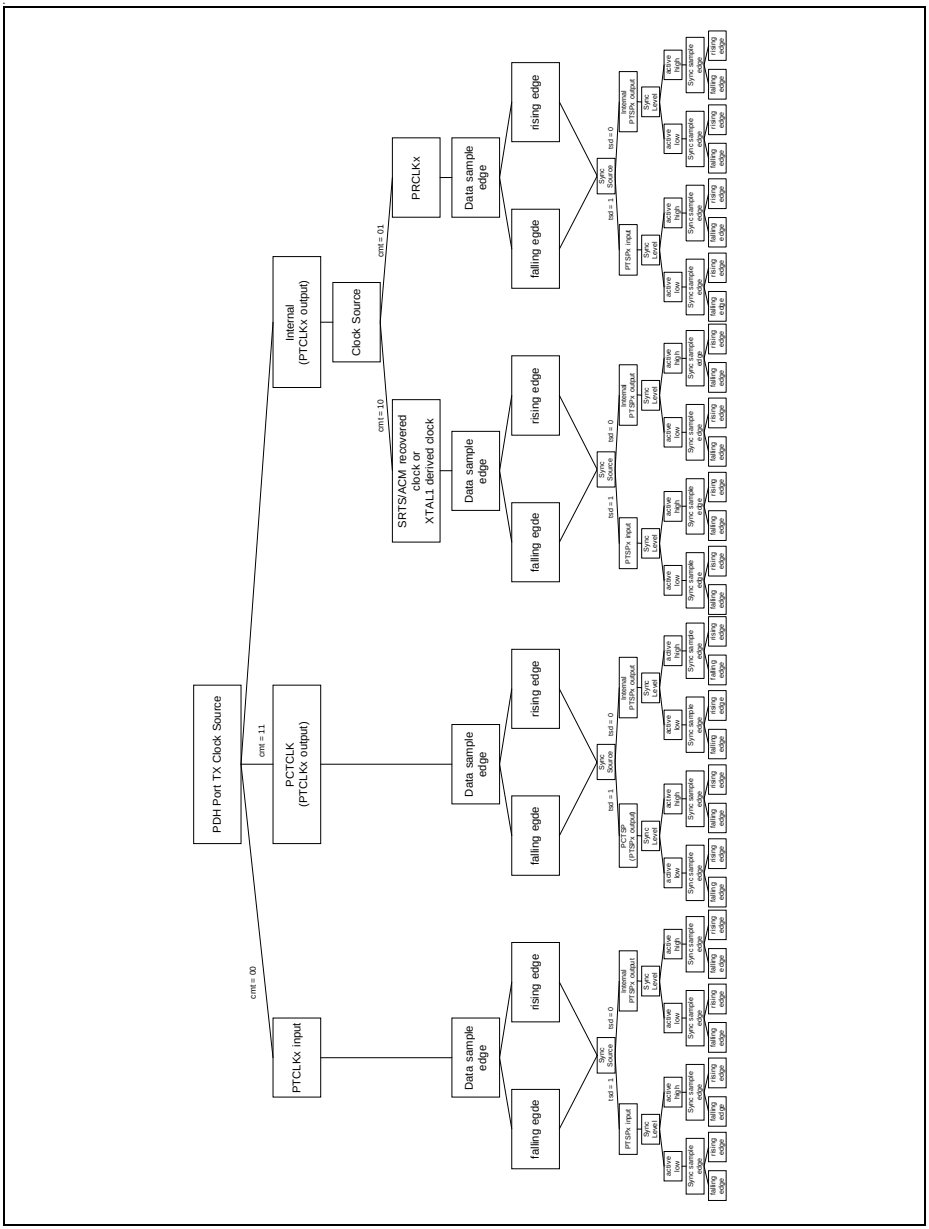


Figure 75 PDH Transmit Clocking Configurations

5.1.4 Loop Modes

5.1.4.1 PDH Port Ingress Loop

The IWORX supports a ingress loop for each of the 8 PDH ports.

The ingress loop mirrors the complete incoming serial data stream including the framing information to the transmit interface. The ingress loop function uses the receive clock PRCLK(x) as the transmit clock. The transmit clock and the transmit synchronization pulse are switched automatically to output mode when the ingress will be enabled.

In ingress loop mode incoming data is forwarded to other internal functional blocks, the ingress loop is transparent.

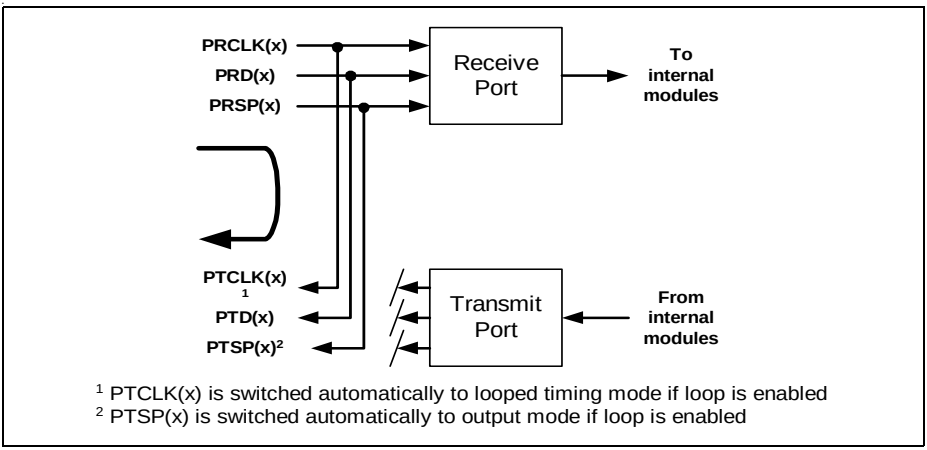


Figure 76 PDH Port Ingress Loop

5.1.4.2 PDH Port Egress Loop

The egress loop is closed in the PDH port interface and mirrors the outgoing bit stream to the receive port. This provides for loops going from TX direction to RX direction. Since the receive and transmit clock have to have the same clock frequency the receiver is operated with the same clock as the transmitter, in other words the receive clock is replaced with the selected transmit clock.

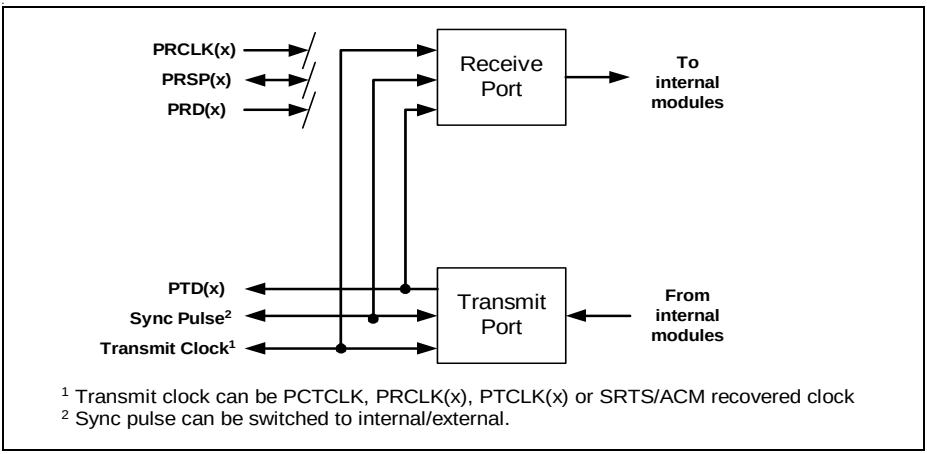


Figure 77 PDH Port Egress Loop

5.1.5 PDH Test Port

The test function provides the capability to multiplex one of the incoming 8 receive links to the outgoing test port, that is the incoming receive clock signal PRCLK(x) is mapped to the test receive clock output PTRCLK, the receive synchronization pulse (the external signal PRSP(x) or the internally generated signal) is mapped to the signal PTRSP and the incoming receive data signal PRD(x) is mapped to the test data output PTRD. In the opposite direction one of the 8 transmit data output signals PTD(x) can be replaced with the test transmit data signal PTTD. Furthermore the corresponding transmit synchronization pulse and the selected transmit clock signal can be monitored on the test port outputs PTTSP and PTTCLK. Each output signal is a buffered version of the corresponding signal source, e.g. the output signal PTD(x) is a buffered version of the incoming signal PTTD.

5.2 UTOPIA Interface

The UTOPIA Interface block supports both the ATM Forum UTOPIA Specification Level 1 Version 2.01 as well as the UTOPIA Specification Level 2 Version 1.0. In addition the cell length can be extended by an External Routing Tag of up to 8 octet totalling to 64 octets.

Receive and transmit path can be configured independently.

5.2.1 Receive Path

The receive path is associated with the Rx pins.

5.2.1.1 Electrical Interface

Independent from the UTOPIA interface type (level 1 or level 2) the interface width may be switched to 8 or 16 bit. The parity over the data bus will be continously generated.

For board testing all outputs are switched to high impedance during start-up and power-on. The outputs of each direction are activated independently from each other after the corresponding UTOPIA configuration message (CMD_UTOPIA_MODE_RX_SET or CMD_UTOPIA_MODE_TX_SET) is sent.

5.2.1.2 Cell Format

Depending on the bus width different cell formats are supported. In both modes the length of the External Routing Tag can be adjusted between 0 and 10 octet. The octets 0 to 6 can be configured by the user independently for each connection. Remaining octets are filled with zero. Additionally an odd parity over the 1st octet may be calculated and inserted.

Table 31 Cell Format in 8-bit Mode

	7	0
0	External Routing Tag	Parity
L		
L+1	ATM Header	
L+4		
L+5	UDF 1	

CONFIDENTIAL

Table 31 Cell Format in 8-bit Mode

L+6	ATM SDU
L+53	

The contents of UDF2 to be inserted between UDF1 and ATM payload in 16-bit mode can be globally configured.

Table 32 Cell Format in 16-bit Mode

	15	8	7	0
0		Parity		
L	External Routing Tag			
L+1	ATM Header			
L+2				
L+3	UDF 1		UDF 2	
L+4	ATM SDU			
L+27				

5.2.1.3 Operation Mode

The UTOPIA module supports cell level handshake with octet level flow control. It can be run in the following configurations:

- UTOPIA Level 2 MPHY with up to 31 PHYs, required mode by standard
- UTOPIA Level 2 MPHY with up to 31 PHYs, optional multiplexed status polling mode by standard
- UTOPIA Level 1 PHY (TXENB may be permanently enabled)

Once the operation mode is selected at start-up individual UTOPIA addresses can be enabled/disabled dynamically in level 2 modes. To prevent head-of-line-blocking cells destined to UTOPIA addresses that are currently not enabled will be discarded internally indicated by a notification.

5.2.1.4 Buffering

The receive UTOPIA datapath provides buffer space for up to 12 cells. The logical size of this buffer can be adjusted between 4 and 12 cells. Upon reaching the threshold a notification will be generated. Cells will not be discarded in case of a reached threshold but in case of an buffer overflow for which also notification will be generated.

5.2.1.5 Notifications

The following events will generate a notification.

- A cell was discarded because the physical size of the receive buffer was exhausted
- The amount of cells in the receive buffer reached the predefined threshold level
- A cell was detected in the receive buffer that belongs to an UTOPIA address which is not enabled

5.2.2 Transmit Path

The transmit path is associated with the Tx pins.

5.2.2.1 Electrical Interface

Independent from the UTOPIA interface type (level 1 or level 2) the interface width may be switched to 8 or 16 bit. Optionally the parity over the data bus can be checked. If paritiy check is enabled received cells with parity error will be discarded and a notification is sent.

For board testing all outputs may be switched to high impedance.

5.2.2.2 Cell Format

Depending on the bus width different cell formats are supported. In both modes the length of the External Routing Tag can be adjusted between 0 and 10 octet. Optionally an odd parity over the 1st octet may be checked. If a cell with parity error is received a notification will be sent. Discarding of cells with errored External Routing Tag parity can be enabled independently.

Table 33 Cell Format in 8-bit Mode

	7	0
0	External Routing Tag	Parity
L		

CONFIDENTIAL

Table 33 Cell Format in 8-bit Mode

L+1 … L+4	ATM Header
L+5	UDF 1
L+6 … L+53	ATM SDU

The contents of UDF2 received in 16-bit mode will be discarded.

Table 34 Cell Format in 16-bit Mode

	15 … 8	7 … 0
0 … L	Parity / External Routing Tag	
L+1 … L+2	ATM Header	
L+3	UDF 1	UDF 2
L+4 … L+27	ATM SDU	

5.2.2.3 Operation Mode

The UTOPIA module supports cell level handshake with octet level flow control. It can be run in the following configurations:

- UTOPIA Level 2 MPHY with up to 31 PHYs, required mode by standard
- UTOPIA Level 2 MPHY with up to 31 PHYs, optional multiplexed status polling mode by standard
- UTOPIA Level 1 PHY (TXENB may be permanently enabled)

Once the operation mode is selected at start-up individual UTOPIA addressables can be enabled/disabled dynamically in level 2 modes.

5.2.2.4 Buffering

The transmit UTOPIA datapath provides buffer space for up to 12 cells. The logical size of this buffer can be adjusted between 4 and 12 cells. Upon reaching the threshold a notification will be generated. Cells will not be discarded in case of a reached threshold. In case of an buffer overflow no additional cells will be accepted from UTOPIA.

5.2.2.5 Notifications

The following events will generate a notification.

- The amount of cells in the transmit buffer reached the predefined threshold level
- A cell with errored External Routing Tag was received
- A parity error occured on the transmit interface. The affected cell is discarded

5.2.3 Loopback Modes

Two kind of loopback modes are available: The ingress loop is at the device interface boundary towards the inner functional blocks, the egress loop is at the device interface boundary towards the interface circuit. If the ingress loop is enabled receiver and transmitter must use the same format with the same UTOPIA addresses enabled.

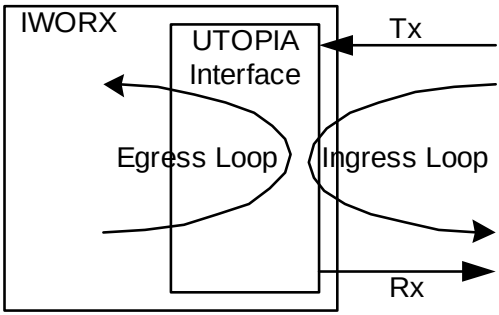


Figure 78 UTOPIA Loopback Modes

5.3 Microprocessor Interface

In the following, the Microprocessor Interface is abbreviated as MPI. The MPI supports direct access to the internal Device Control Interface (DCI) via a 16 or 32 bit wide Microprocessor Data Bus. The DCI area is 32 bits wide.

Features

- acts as a slave in the external microprocessor system
- supports Intel Demux Mode (little endian)
- supports synchronous Motorola Mode (big endian)
- supports asynchronous Motorola Mode (big endian)
- 16/32 bit data bus width
- supports 8/16/32 bit access
- READY/DTACK controlled bus
- optimized for fast access and minimum waitstates
- includes 8k x 32 bit Device Control Interface (DCI) area for message exchange and virtual registers
- queuing of messages in up to four FIFO queues
- all RAM cells are cleared automatically upon reset
- external interrupt generation to support handling of message queues and virtual registers

5.3.1 Mode Selection

The Microprocessor Interface Mode is seleced via the corresponding mode pins at the Microprocessor Interface. **Table 35** gives an overview on mode selection.

Table 35 Microprocessor Interface Mode Selection

MPIM1	MPIM0	Mode
0	0	Prohibited
0	1	Synchronous Motorola Mode
1	0	Intel Demux Mode
1	1	Asynchronous Motorola Mode

5.3.1.1 16 Bit Mode

The signals MPBHE/BLE control the transfer between the internal 32 bit DCI and the external 8 or 16 bit MPD. Refer to **Table 36** for data access in Intel Demux Mode and to **Table 37** for Motorola Modes.

Table 36 Data Access in Intel Demux Mode

MPBHE	MPAD1	MPAD0	Access Type	MPD	DCI
0	0	0	16 bit	MPD[15:0]	DCI[15:0][1)]
0	1	0	16 bit	MPD[15:0]	DCI[31:16][1)]
1	0	0	8 bit	MPD[7:0][2)]	DCI[7:0][1)]
0	0	1	8 bit	MPD[15:8][3)]	DCI[15:8][1)]
1	1	0	8 bit	MPD[7:0][2)]	DCI[23:16][1)]
0	1	1	8 bit	MPD[15:8][3)]	DCI[31:24][1)]
1	X	1	not allowed		

1) all other databus signals are don't care

2) in case of read MPD[15:8] are don't care, in case of write MPD[15:8] are ignored

3) in case of read MPD[7:0] are don't care, in case of write MPD[7:0] are ignored

Table 37 Data Access in Motorola Modes

MPBLE	MPAD1	MPAD0	Access Type	MPD	DCI
0	0	0	16 bit	MPD[15:0][1)]	DCI[31:16][2)]
0	1	0	16 bit	MPD[15:0][1)]	DCI[15:0][2)]
1	0	0	8 bit	MPD[15:8][1)]	DCI[31:24][2)]
0	0	1	8 bit	MPD[7:0][1)]	DCI[23:16][2)]
1	1	0	8 bit	MPD[15:8][1)]	DCI[15:8][2)]
0	1	1	8 bit	MPD[7:0][1)]	DCI[7:0][2)]
1	X	1	not allowed		

1) in case of read all other data bus signals are don't care, in case of write all other databus signals are are ignored

2) all other databus signals are don't care

5.3.1.2 32 Bit Mode

The signals $\overline{BE0}$ to $\overline{BE3}$ / TSIZ[1:0]) and AD[1:0]) (Intel/Motorola) control transfer between the internal 32 bit bus and external 32/16/8 bit accesses. Refer to **Table 38** for data rearrangement in Intel Demux Mode and to **Table 39** for Motorola mode.

After reset the MPI is be initiated in 16 bit mode. The 16/32 bit mode selection is done via a register and described in the Programmer's Reference Manual of the device. The

signals $\overline{BE0}$ to $\overline{BE2}$ are mapped to the signals MPAD0, MPAD1 and $\overline{BHE}$/$\overline{BLE}$. A 32 bit system is able to switch from the 16 bit to the 32 bit mode after reset.

Table 38 Data Access in Intel Demux Mode

$\overline{MPBE3}$	$\overline{MPBE2}$	$\overline{MPBE1}$	$\overline{MPBE0}$	Access Type	MPD	DCI
0	0	0	0	32 bit	MPD[31:0]	DCI[31:0]
1	1	0	0	16 bit	MPD[15:0][1)]	DCI[15:0][2)]
0	0	1	1	16 bit	MPD[31:16][1)]	DCI[31:16][2)]
1	1	1	0	8 bit	MPD[7:0][1)]	DCI[7:0][2)]
1	1	0	1	8 bit	MPD[15:8][1)]	DCI[15:8][2)]
1	0	1	1	8 bit	MPD[23:16][1)]	DCI[23:16][2)]
0	1	1	1	8 bit	MPD[31:24][1)]	DCI[31:24][2)]
other combinations				not allowed		

1) in case of read all other data bus signals are don't care, in case of write all other databus signals are are ignored

2) all other databus signals are don't care

Table 39 Data Access in Motorola Modes

MPTSIZ 1	MPTSIZ 0	MPAD1	MPAD0	Access Type	MPD	DCI
0	0	0	0	32 bit	MPD[31:0]	DCI[31:0]
1	0	0	0	16 bit	MPD[31:16][1)]	DCI[31:16][2)]
1	0	1	0	16 bit	MPD[15:0][1)]	DCI[15:0][2)]
0	1	0	0	8 bit	MPD[31:24][1)]	DCI[31:24][2)]
0	1	0	1	8 bit	MPD[23:16][1)]	DCI[23:16][2)]
0	1	1	0	8 bit	MPD[15:8][1)]	DCI[15:8][2)]
0	1	1	1	8 bit	MPD[7:0][1)]	DCI[7:0][2)]
other combinations				not allowed		

1) in case of read all other data bus signals are don't care, in case of write all other databus signals are are ignored

2) all other databus signals are don't care

5.3.2 Device Configuration Interface

This is the interface area between the internal bus and the external bus. The RAM data width is 32bit. The depth is 8192 words, resulting in an overall size of 32 kBytes.

The FIFO controller manages up to 4 FIFOs inside the Dual Port RAM area. These FIFOs are accessible from the internal and from the external bus for read and write, with the external access having the higher priority. The width of the FIFO words are 32bit.

The read and write pointer of the FIFO points always to word addresses. One byte of the word is configured as an "indicator" byte. Writing to the indicator byte of a FIFO word increments the write pointer, reading the indicator byte increments the read pointer. While reading or writing other bytes of a word the pointers don't move.

A level counter indicates how many words are stored inside the FIFO. The counter is incremented with every indicator byte write and is decremented with every indicator byte read. This counter is stored in a read only register and this register is accessible from both sides. In case of reading an empty FIFO the FIFO controller returns the value 0. In case of writing to a full FIFO the written word is discarded.

The FIFO controller can trigger an external interrupt (pin MPINT) to notify the host processor that messages in outgoing queues are ready to be read. The interrupt pin MPINT can be programmed to be open drain or push/pull active high or active low.

Interrupt status register and interrupt mask register are provided. To allow 16 or 8 bit accesses a shadow buffer is implemented. One byte of the interrupt status register bytes is configured as the indicator byte. This byte must be accessed first. At this moment the interrupt status register is copied to the shadow register and then cleared. All other read accesses to the interrupt status register not accessing the indicator byte are served by the shadow register.

5.4 JTAG Interface

The boundary scan functionality is implemented according to IEEE 1149.1, using a 5-pin test access port.

5.5 OCDS Interface

External debug equipment is connected to IWORX's on-chip debugging system (OCDS) via the JTAG interface (control) and the OCDS interface, which provides status, breakpoint and program counter information. The OCDS interface is intended only for use by the maufacturer.

5.6 Tracing Interface

The tracing interface with the pins FREEZEI and FREEZEO has been implemented for debug purposes. The implemented HW blocks of the IWORX can be set in the freeze mode by an internal trigger signal in case of errors or by asserting the pin FREEZEI. All internal register and RAM entries are frozen in the freeze mode. The pin FREEZEO pin indicates by an "1" that freeze mode is active. Both pins should be accessible on the board but stimulated only on demand of the manufacturer when errors occur.

5.7 Control Interface

Power-On Reset and Host Reset

Setting the external input pin $\overline{\text{POR}}$ to 0 resets the complete chip with new image download. This is an asynchronous reset. The pin $\overline{\text{HRST}}$ is intended to be used only by the manufacturer for e.g. applying a synchronous reset.

Non-Maskable Interrupt

The pin $\overline{\text{NMI}}$ supports the possibility to transfer an interrupt to the implemented embedded controller. This pin is intended only for use by the maufacturer.

Boot Mode Configuration

The 3 pins CFG[2..0] exist for configuration of the boot mode and should be set to "001", CFG0 to high level, CFG1 to low level and CFG2 to low level for normal boot operation via the microprocessor interface. Any other setting are for debug purpose or future use and should be applied only by the manufacturer.

Firmware Release Pin

The pin FWR outputs continuously a 32 bit entry without gaps.

31	30	29	28	27	26	25	24
0	0	0	0	0	0	0	0

23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	1

15	14	13	12	11	10	9	8
FW functionality ID				FW functionality subset ID			

7	6	5	4	3	2	1	0
FW development status ID							

Bit 31 is the first bit sent out. Bit 31..16 contains the start pattern 0x0001, bit 15..12 the FW functionality ID, bit 11..8 the FW functionality subset ID and bit 7..0 the FW develpoment status ID (see also **Chapter 1.4**). The pin output for FW release 1.1-1.2.1 will look like as shown in **Figure 79**.

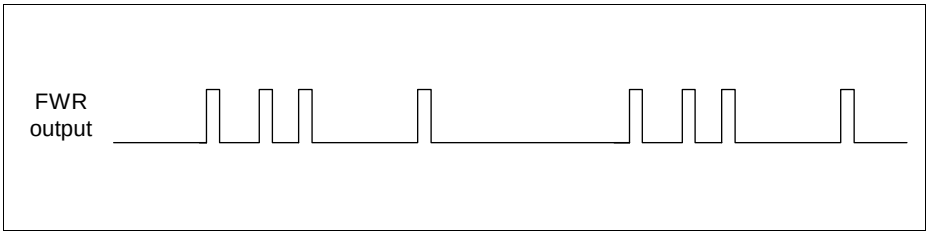


Figure 79 FWR Output with FW Release 1.1-1.2.1

5.8 UART Interface

The UART interface will be used by the manufacturer as a debug terminal for firmware development and isn't intended for use by the IWORX customer. The UART interface does not implement a receive FiFo. Autobauding is not supported. The maximum baud rate is 3,5 Mbaud.

5.9 Clock Interface

IWORX uses a three-stage clock generation scheme:

- Crystal Oscillator
- Analog PLL
- DCO

5.9.1 Crystal Oscillator

An integrated crystal oscillator generates a 'local clock' at moderate frequency. Use shunt capacitances of 10 pF between XTAL1 and VSS and between XTAL2 and VSS.

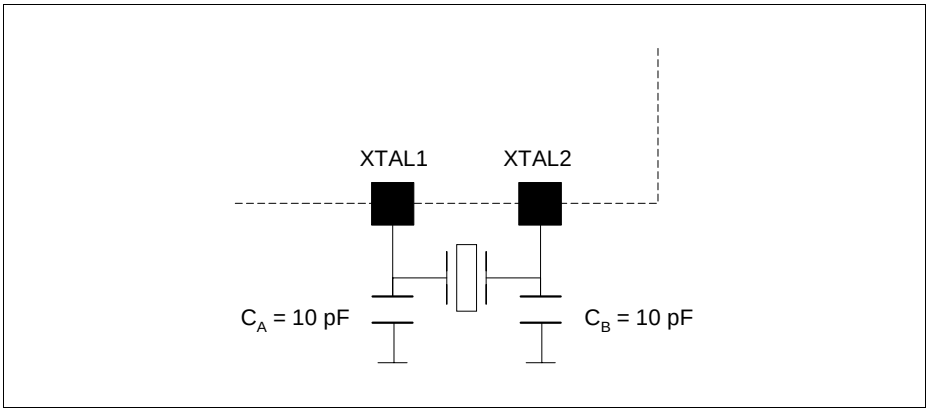


Figure 80 Operation in Crystal Mode

Alternatively the oscillator input ball XTAL1 can be used to feed in an LVTTL clock signal. Frequency range is 10 to 32 MHz.

5.9.1.1 Calculations for Crystal Mode

Crystal Data (from crystal datasheet):

- f_0 crystal frequency
- R_1 series resistance
- C_0 static capacitance
- C_1 series capacitance
- P_{max} max. power dissipation in crystal

Circuit Data:

- C_{AT} total capacitance XTAL1 to VCC (C_A + $C_{Aparasitic}$)
- C_{BT} total capacitance XTAL2 to VCC (C_B + $C_{Bparasitic}$)

CONFIDENTIAL

- C_{IO} total parasitic capacitance between crystal pins (IC and crystal package, PCB), NOT including C_0, approximately 2 pF

Oscillator Data:

- g_m oscillator circuit gain (transconductance) = 8.3 mS
- I_{BIAS} Bias current of the oscillator amplifier = 492 µA

Load Capacitance

Variation of C_L changes the oscillator frequency slightly. Large values for C_A and C_B minimize the influence of the parasitics, while on the other hand they require a high g_m, which means high bias current

$$C_L = \frac{C_{AT} \bullet C_{BT}}{C_{AT} + C_{BT}} + C_{IO}$$

Startup Condition

- R_L effective load resistance
- R_{out} negative output resistance of oscillator circuit

The oscillator will start safely, when the negative output resistance of the oscillator circuit is greater than the effective load resistance..

$$R_L = R_1 \bullet \left(1 + \frac{C_0}{C_L}\right)^2$$

$$R_{out} = -\frac{g_m}{4 \bullet (2 \bullet \pi \bullet f_0)^2 \bullet (C_L + C_0)^2}$$

$$\left|R_{out}\right| / 2 > R_L$$

For stable oscillation following equation has to be fulfilled.

$$g_m < \left(g_m(max) = 2 \bullet \pi \bullet f_0 \bullet \left(C_{AT} + C_{BT} + \frac{C_{AT} \bullet C_{BT}}{C_0 + C_{IO}}\right)\right)$$

CONFIDENTIAL

Amplitude V_1

The amplitude V_1 at XTAL1 should exceed 200 mV for good noise immunity.

$$V_1 \approx \frac{5}{12} \bullet \frac{I_{BIAS}}{R_1 \bullet (2 \bullet \pi \bullet f_0)^2 \bullet (C_L + C_0)^2}$$

Power Dissipation P in the Crystal

$$P = \frac{R_1 \bullet (2 \bullet \pi \bullet f_0 \bullet C_A \bullet V_1)^2}{2} < P_{max}$$

Startup Time $T_{startup}$

$$T_{startup} \approx 150 \bullet \frac{(C_L + C_0)^2}{g_m \bullet C_1}$$

Calculation Example

The crystal and circuit data are:

- f_0 crystal frequency = 32 MHz
- R_1 series resistance = 30 Ohm
- C_0 static capacitance = 7 pF
- C_1 series capacitance = 25 fF
- P_{max} max. power dissipation in crystal = 1mW
- $C_{Aparasitic} = C_{Bparasitic}$ = 7 pF
- $C_A = C_B$ = 10 pF
- $C_{AT} = C_{BT}$ = 17 pF
- C_{IO} = 2 pF

The equation results are:

- C_L = 10.5 pF, R_L = 83.3 Ohm, $|R_{out}|/2$ = 83.9 Ohm ==> safe start
- g_m(max) = 13.3 mS > 8.3 mS ==> stable oscillation
- V_1 = 0.55 V > 0.2 V ==> good noise immunity
- P = 0.02 mW < 1 mW ==> power dissipation is low enough
- $T_{startup}$ = 0.2 ms

5.9.2 Analog PLL

The Analog PLL operates on the local clock frequency. Target PLL frequency is 240 MHz. The analog PLL generates a 120 MHz (nom.) clock used by the IWORX processor

subsystem, and, after division, as a 60 MHz system clock. Furthermore the analog PLL supplies a multiphase clock output at 240 MHz.

After reset, a default value of 32 MHz is assumed for the local clock frequency. This ensures that the chip can be safely booted. Later on the PLL settings can be corrected using a message which defines the actual local clock frequency.

5.9.3 DCO

The DCO synthesizes the E1 and T1 PLL reference frequencies of 65.536 and 49.408 MHz for the internal clock recovery circuit. It operates on the PLL's 240 MHz multiphase clock output.

6 Electrical Characteristics

6.1 Absolute Maximum Ratings

Parameter	Symbol	Limit Values	Unit
Ambient temperature under bias	T_A	-40 to +85	°C
Storage temperature	T_{stg}	-65 to +125	°C
Core supply voltage	V_{DD}	-0.4 to 1.89	V
I/O supply voltage	V_{DDP}	-0.4 to 3.6	V
Voltage on any pin with respect to ground	V_S	-0.4 to 3.6	V
ESD robustness[1)] HBM: 1.5 kΩ, 100 pF	$V_{ESD,HBM}$	2000[2)]	V

1) According to EIA / JESD22-A114-B

2) 1000 V for PLL pins

Note: Stresses above those listed here may cause permanent damage to the device. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

6.2 Operating Range

Parameter	Symbol	Limit Values		Unit	Test Condition
		min.	max.		
Ambient temperature	T_A	-40	85	°C	
Core supply voltage	V_{DD}	1.71	1.89	V	
I/O supply voltage	V_{DDP}	3.0	3.6	V	

Note: In the operating range, the functions given in the circuit description are fulfilled.

6.3 Thermal Package Characteristics

Parameter	Symbol	Limit Values	Unit	Test conditions
Thermal package resistance junction to ambient without airflow	$R_{JA(0,25)}$	18	°C/W	T_A=25°C

6.4 DC Characteristics

Parameter	Symbol	Limit Values		Unit	Notes
		min.	max.		
Input low voltage	V_{IL}	-0.4	0.8[1)]	V	
Input high voltage	V_{IH}	2	3.6	V	
Output low voltage	V_{OL}		0.4	V	$I_{OL, DriverD}$ = 5 mA, $I_{OL, DriverC}$ = 9 mA
Output high voltage	V_{OH}	2.4		V	$I_{OH, DriverD}$ = -5 mA, $I_{OH, DriverC}$ = -9 mA
Avg. V_{DD} power supply current	I_{CC}		1000[2)]	mA	
Avg. V_{DDP} power supply current	I_{CCP}		50[3)]	mA	
Input leakage current	I_{IL}		1.0	µA	Test condition 3.6V
Output leakage current	I_{OL}		1.0	µA	Test condition 3.6V
Pull Up Current	I_{PUB}	11	34	µA	Test condition 0 V
	I_{PUA}	58	170	µA	
Pull Down Current	I_{PDB}	9	42	µA	Test condition 3.3V
	I_{PDA}	48	190	µA	

1) valid for all pins except for pin XTAL1; max. value for XTAL1 is 0.6 V

2) not tested in production

3) not tested in production

Note: The listed characteristics are ensured over the operating range of the integrated circuit. Typical characteristics specify mean values expected over the production spread. If not otherwise specified, typical characteristics apply at T_A = 25 °C and the given supply voltage.

6.5 AC Characteristics

T_A = -40 to 85 °C, V_{DDP} = 3.3 V ± 10%, V_{DD} = 1.8 V ± 5%, V_{SS} = 0 V

All inputs are driven to V_{IH} = 2.4 V for a logical “1” and to V_{IL} = 0.4 V for a logical “0”

All outputs are measured at V_{TH} = 2.0 V for a logical “1”and at V_{TL} = 0.8 V for a logical “0”

The AC testing input/output waveforms are shown below.

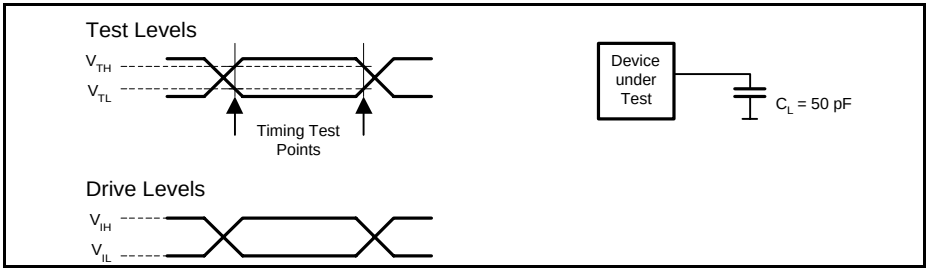


Figure 81 Input/Output Waveforms for AC Measurements

6.5.1 PDH Interface

PDH Receive Interface

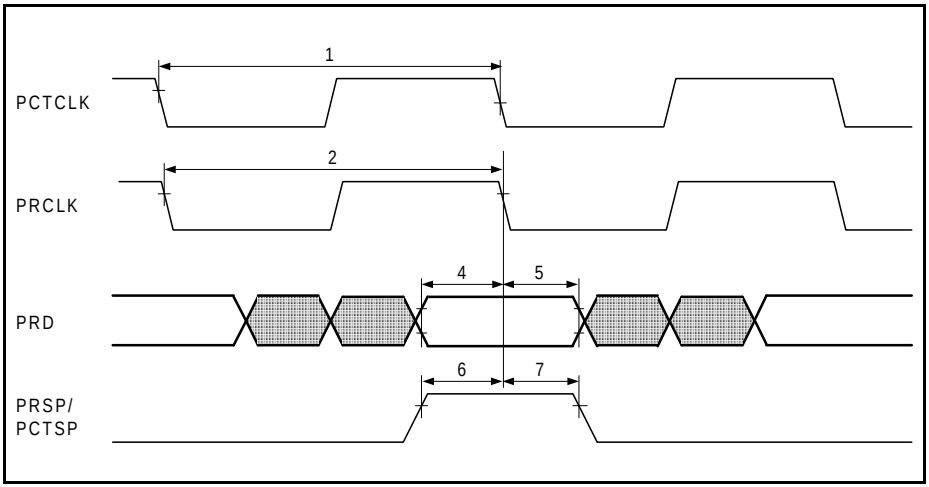


Figure 82 PDH Receive Interface Timing

Table 40 PDH Receive Interface Timing

No.	Parameter	Limit Values			Unit
		Min	Typ	Max	
1	T_{PCTCLK}: Period PCTCLK				
	E1:		488		ns
	T1:		647		ns
1A	F_{PCTCLK}: Frequency PCTCLK				
	E1:		2,048		MHz
	T1:		1,544		MHz
2	T_{PRCLK}: Period PRCLK				
	E1:		488		ns
	T1:		647		ns
2A	F_{PRCLK}: Frequency PRCLK				
	E1:		2,048		MHz
	T1:		1,544		MHz

Table 40 PDH Receive Interface Timing (cont'd)

No.	Parameter	Limit Values			Unit
		Min	Typ	Max	
4	Setup time PRD before PRCLK/ PCTCLK falling/rising (center of bit period)	25			ns
5	Hold time PRD after PRCLK/PCTCLK falling/rising (center of bit period)	25			ns
6	Setup time PRSP/PCTSP before PRCLK/PCTCLK falling/rising (center of bit period)	25			ns
7	Hold time PRSP/PCTSP after PRCLK/ PCTCLK falling/rising (center of bit period)	25			ns

PDH Transmit Interface

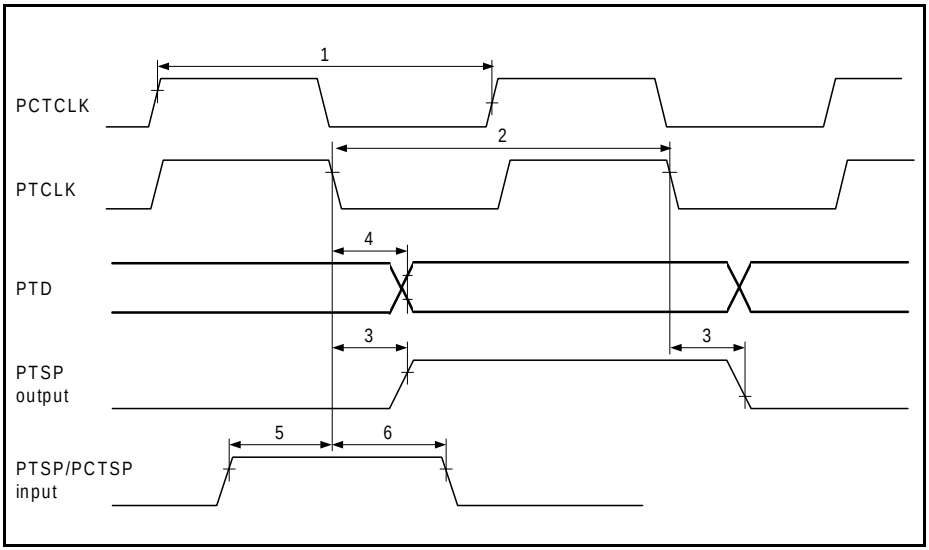


Figure 83 PDH Transmit Interface Timing

Table 41 PDH Transmit Interface Timing

No.	Parameter	Limit Values			Unit
		Min	Typ	Max	
1	T_{PCTCLK}: Period PCTCLK				
	E1:		488		ns
	T1:		647		ns
1A	F_{PCTCLK}: Frequency PCTCLK				
	E1:		2,048		MHz
	T1:		1,544		MHz
2	T_{PTCLK}: Period PTCLK				
	E1:		488		ns
	T1:		647		ns
2A	F_{PTCLK}: Frequency PTCLK				
	E1:		2,048		MHz
	T1:		1,544		MHz
3	Delay PTCLK/PCTCLK in falling to PTSP	3		32	ns
	Delay PTCLK out falling to PTSP	-7		32	ns
4	Delay PTCLK/PCTCLK in falling to PTD	3		32	ns
	Delay PTCLK out falling to PTD	-7		32	ns
5	Setup time PTSP/PCTSP before PTCLK/PCTCLK falling	25			ns
6	Hold time PTSP/PCTSP after PTCLK/ PCTCLK falling	25			ns

6.5.2 UTOPIA Interface

The AC characteristics of the UTOPIA interface fulfills the ATM Forum “UTOPIA level 2 Specification, Version 1.0" as defined for the interface running at 50 MHz.

The AC characteristics are based on the timing specification for the receiver side of a signal.

The setup and the hold times are defined with regard to a positive clock edge, see **Figure 84**.

Taking the actual used clock frequency into account (e.g. up to the max. frequency), the corresponding (min. and max.) transmit side “clock to output” propagation delay specifications can be derived. The timing references (tT5 to tT12) are according to**Table 42** to **Table 45**.

In the following tables, A>P (column DIR, Direction) defines a signal from the ATM layer (transmitter, driver) to the PHY layer (receiver), A<P defines a signal from the PHY layer (transmitter, driver) to the ATM layer (receiver).

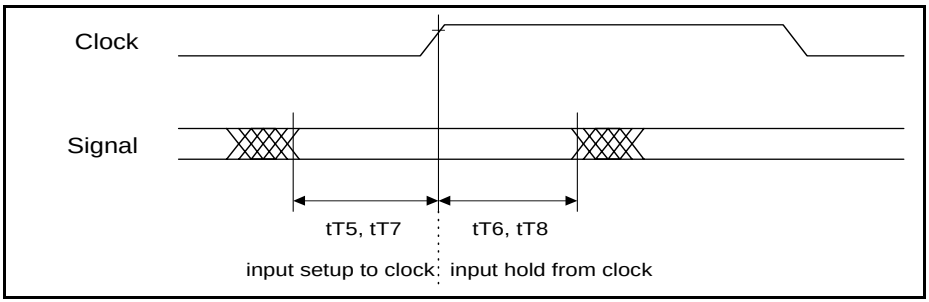


Figure 84 Setup and Hold Time Definition (single- and multi PHY)

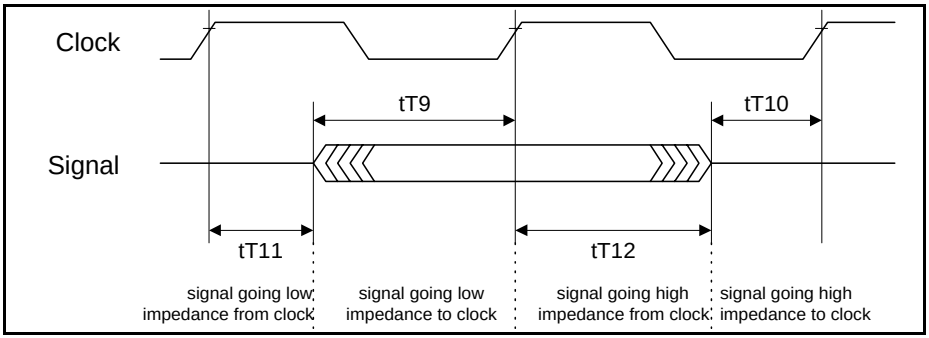


Figure 85 Tri-state Timing (multi-PHY, multiple devices only)

Table 42 Transmit Timing (16-Bit Data Bus, 50 MHz at Cell Interface, Single PHY)

No.	Signal Name	DIR	Description	Limit Values		Unit
				Min	Max	
t1	UATMCLK[1)]	A>P	UATMCLK frequency (nominal)	0	50	MHz
tT2			UATMCLK duty cycle	40	60	%
tT3			UATMCLK peak-to-peak jitter	-	5	%
tT4			UATMCLK rise/fall time	-	2	ns
tT5	UTXDATA, UTXPRTY, UTXSOC, UTXENB	A>P	Input setup to UATMCLK	4	-	ns
tT6			Input hold from UATMCLK	1	-	ns
tT7	UTXCLAV	A<P	Input setup to UATMCLK	4	-	ns
tT8			Input hold from UATMCLK	1	-	ns

1) The frequency should be equal or smaller than 52 MHz

Table 43 Receive Timing (16-Bit Data Bus, 50 MHz at Cell Interface, Single PHY)

No.	Signal Name	DIR	Description	Limit Values		Unit
				Min	Max	
t1	UATMCLK[1)]	A>P	UATMCLK frequency (nominal)	0	50	MHz
tT2			UATMCLK duty cycle	40	60	%
tT3			UATMCLK peak-to-peak jitter	-	5	%
tT4			UATMCLK rise/fall time	-	2	ns
tT5	URXENB	A>P	Input setup to UATMCLK	4	-	ns
tT6			Input hold from UATMCLK	1	-	ns
tT7	URXDATA URXPRTY, URXSOC, URXCLAV	A<P	Input setup to UATMCLK	4	-	ns
tT8			Input hold from UATMCLK	1	-	ns

1) The frequency should be equal or smaller than 52 MHz

Table 44 Transmit Timing (16-Bit Data Bus, 50 MHz at Cell Interface, Multi-PHY)

No.	Signal Name	DIR	Description	Limit Values		Unit
				Min	Max	
t1	UATMCLK[1)]	A>P	UATMCLK frequency (nominal)	0	50	MHz
tT2			UATMCLK duty cycle	40	60	%
tT3			UATMCLK peak-to-peak jitter	-	5	%
tT4			UATMCLK rise/fall time	-	2	ns
tT5	UTXDATA UTXPRTY, UTXSOC, UTXENB, UTXADDR	A>P	Input setup to UATMCLK	4	-	ns
tT6			Input hold from UATMCLK	1	-	ns
tT7	UTXCLAV	A<P	Input setup to UATMCLK	4	-	ns
tT8			Input hold from UATMCLK	1	-	ns
tT9			Signal going low impedance to UATMCLK	4	-	ns
tT10			Signal going high impedance to UATMCLK	0	-	ns
tT11			Signal going low impedance from UATMCLK	1	-	ns
tT12			Signal going high impedance from UATMCLK	1	-	ns

1) The frequency should be equal or smaller than 52 MHz

Table 45 Receive Timing (16-Bit Data Bus, 50 MHz at Cell Interface, Multi-PHY)

No.	Signal Name	DIR	Description	Limit Values		Unit
				Min	Max	
t1	UATMCLK[1)]	A>P	UATMCLK frequency (nominal)	0	50	MHz
tT2			UATMCLK duty cycle	40	60	%
tT3			UATMCLK peak-to-peak jitter	-	5	%
tT4			UATMCLK rise/fall time	-	2	ns
tT5	$\overline{\text{URXENB}}$, URXADDR	A>P	Input setup to UATMCLK	4	-	ns
tT6			Input hold from UATMCLK	1	-	ns
tT7	URXDATA URXPRTY, URXSOC, URXCLAV	A<P	Input setup to UATMCLK	4	-	ns
tT8			Input hold from UATMCLK	1	-	ns
tT9			Signal going low impedance to UATMCLK	4	-	ns
tT10			Signal going high impedance to UATMCLK	0	-	ns
tT11			Signal going low impedance from UATMCLK	1	-	ns
tT12			Signal going high impedance from UATMCLK	1	-	ns

1) The frequency should be equal or smaller than 52 MHz

6.5.3 Microprocessor Interface

6.5.3.1 Intel Demux Mode

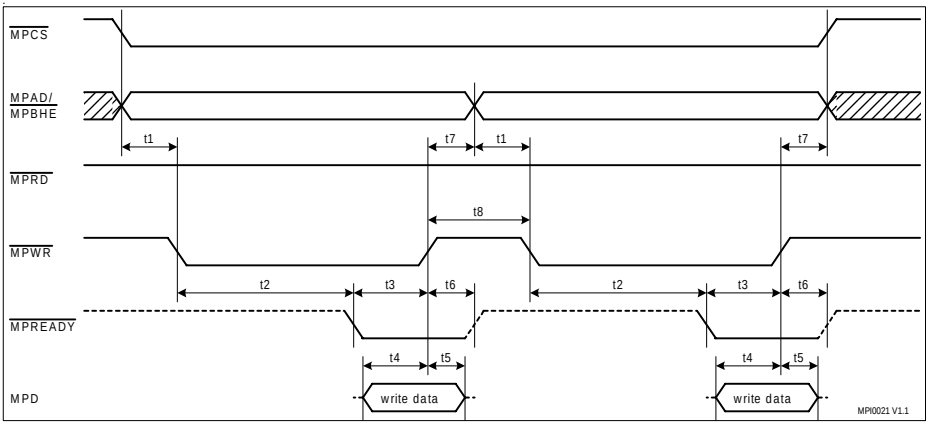


Figure 86 Intel Demux Mode Write Timing

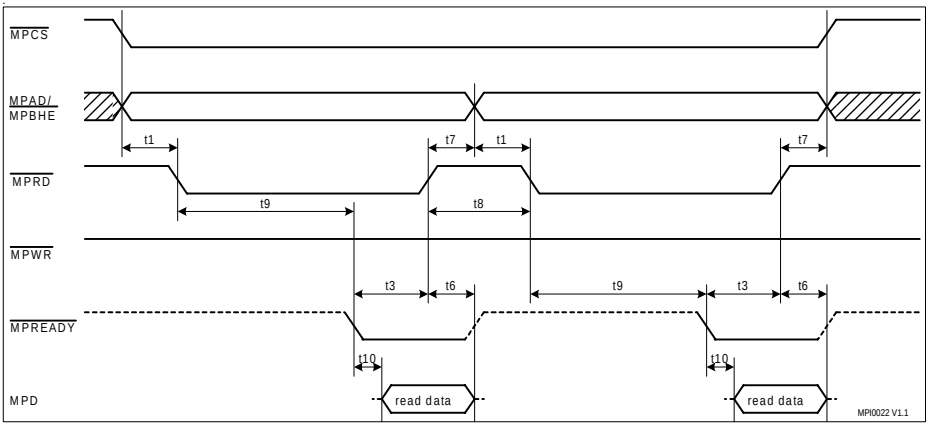


Figure 87 Intel Demux Mode Read Timing

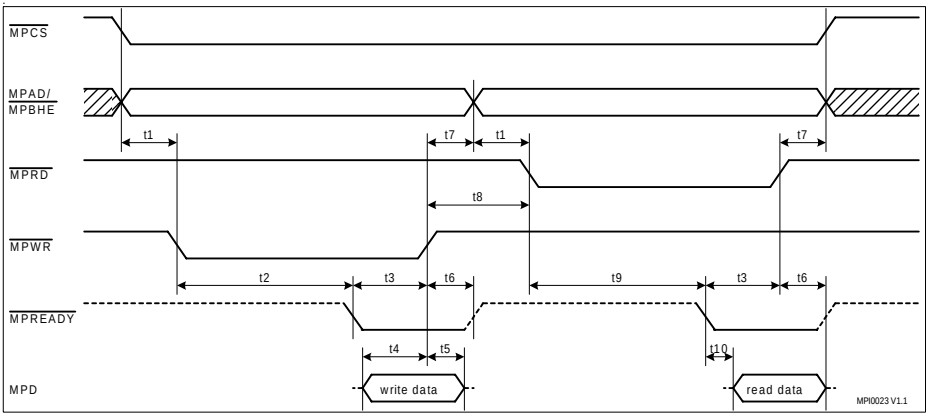


Figure 88 Intel Demux Mode Write/Read Timing

Table 46 Timings for Intel Demux Mode

Timing	Description	min	typ	max	unit
t1	$\overline{\text{MPCS}}$, MPAD/$\overline{\text{MPBHE}}$ setup to $\overline{\text{MPWR}}$/ $\overline{\text{MPRD}}$	0			ns
t2	$\overline{\text{MPWR}}$ to $\overline{\text{MPREADY}}$ delay	25	50		ns
t3	$\overline{\text{MPWR}}$/$\overline{\text{MPRD}}$ delay after $\overline{\text{MPREADY}}$	0			ns
t4	MPD setup to $\overline{\text{MPWR}}$	10			ns
t5	MPD hold from $\overline{\text{MPWR}}$	10			ns
t6	$\overline{\text{MPREADY}}$ and read MPD hold/inactive after $\overline{\text{MPWR}}$/$\overline{\text{MPRD}}$	5		15	ns
t7	$\overline{\text{MPCS}}$, MPAD/$\overline{\text{MPBHE}}$ hold from $\overline{\text{MPWR}}$/ $\overline{\text{MPRD}}$	0			ns
t8	$\overline{\text{MPWR}}$ to $\overline{\text{MPWR}}$, $\overline{\text{MPRD}}$ to $\overline{\text{MPRD}}$, $\overline{\text{MPWR}}$ to $\overline{\text{MPRD}}$ and $\overline{\text{MPRD}}$ to $\overline{\text{MPWR}}$ inactive time	21			ns
t9	$\overline{\text{MPRD}}$ to $\overline{\text{MPREADY}}$ delay	85	150		ns
t10	delay between $\overline{\text{MPREADY}}$ and read MPD driven and stable	-17		0	ns

6.5.3.2 Asynchronous Motorola Mode

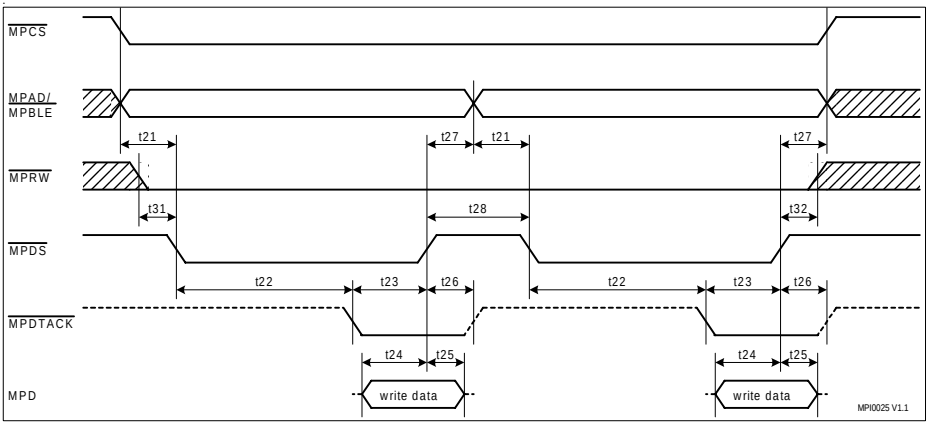


Figure 89 Asynchronous Motorola Mode Write Timing

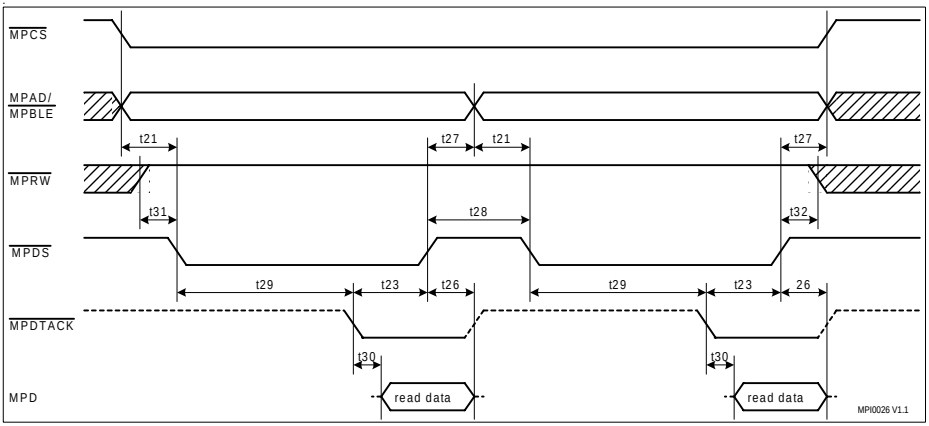


Figure 90 Asynchronous Motorola Mode Read Timing

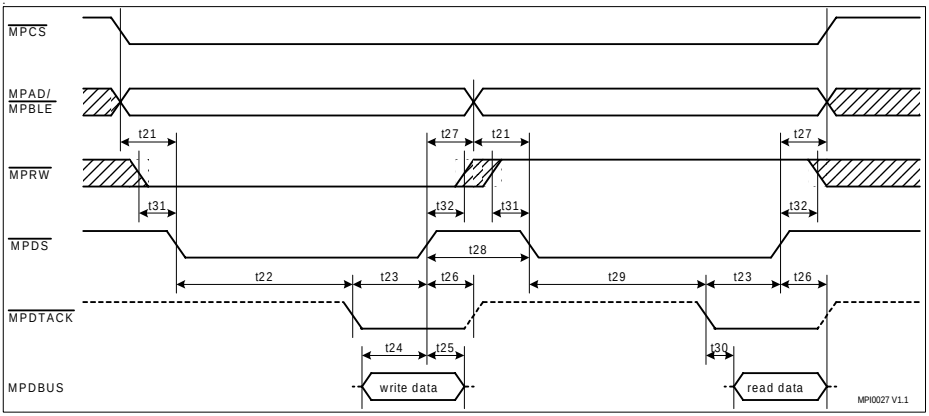


Figure 91 Asynchronous Motorola Mode Write/Read Timing

Table 47 Timings for Asynchronous Motorola Mode

Timing	Description	min	typ	max	unit
t21	$\overline{\text{MPCS}}$, MPAD/$\overline{\text{MPBLE}}$ setup to $\overline{\text{MPDS}}$	0			ns
t22	$\overline{\text{MPDS}}$ to $\overline{\text{MPDATCK}}$ delay in case of write	25	50		ns
t23	$\overline{\text{MPDS}}$ delay after $\overline{\text{MPDTACK}}$	0			ns
t24	write MPD setup to $\overline{\text{MPDS}}$	10			ns
t25	write MPD hold from $\overline{\text{MPDS}}$	10			ns
t26	$\overline{\text{MPDTACK}}$ and read MPD hold/inactive after $\overline{\text{MPDS}}$	5		15	ns
t27	$\overline{\text{MPCS}}$, MPAD/$\overline{\text{MPBLE}}$ hold from $\overline{\text{MPDS}}$	0			ns
t28	$\overline{\text{MPDS}}$ inactive time	21			ns
t29	$\overline{\text{MPDS}}$ to $\overline{\text{MPDTACK}}$ delay in case of read	85	150		ns
t30	delay between $\overline{\text{MPDTACK}}$ and read MPD driven and stable	-17		0	ns
t31	$\overline{\text{MPRW}}$ setup to $\overline{\text{MPDS}}$	10			ns
t32	$\overline{\text{MPRW}}$ hold from $\overline{\text{MPDS}}$	10			ns

6.5.3.3 Synchronous Motorola Mode

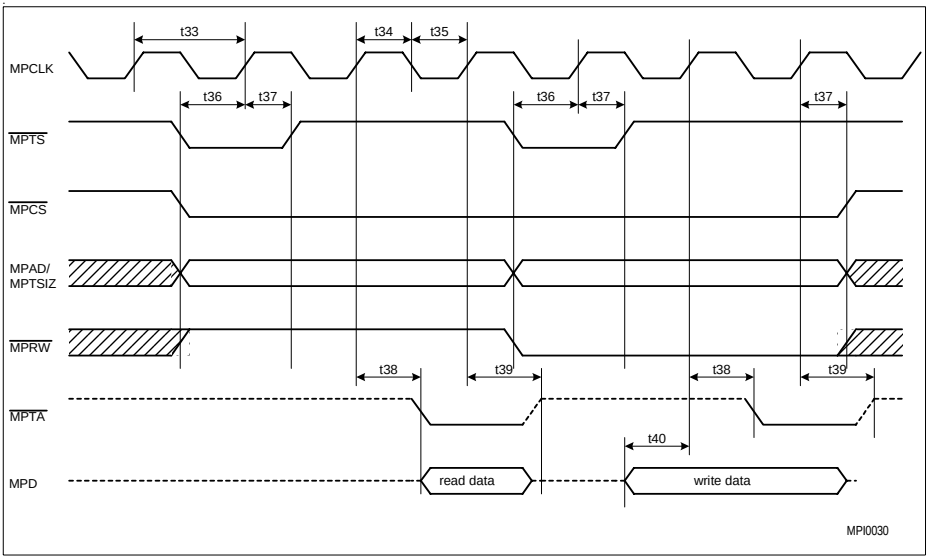


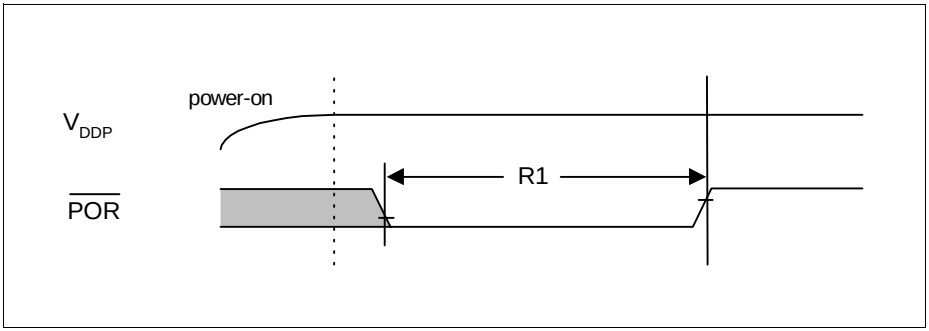
Figure 92 Synchronous Motorola Mode Write/Read Timing

Please refer to the MPC860 specification for the synchronous Motorola mode timing. All given timings fit directly to the MPC860 timing specification.

Table 48 Timings for Synchronous Motorola Mode

Timing	Description	min	typ	max	unit
t33	MPCLK period	20			ns
t34	MPCLK high period	8			ns
t35	MPCLK low period	8			ns
t36	$\overline{\text{MPTS}}$, $\overline{\text{MPCS}}$, MPAD/MPTSIZ, $\overline{\text{MPRW}}$ setup time	8.25			ns
t37	$\overline{\text{MPTS}}$, $\overline{\text{MPCS}}$, MPAD/MPTSIZ, $\overline{\text{MPRW}}$ hold time	5			ns
t38	$\overline{\text{MPTA}}$, MPD delay	1		10.25	ns
t39	$\overline{\text{MPTA}}$, MPD deasserted to tristate	1		10.25	ns
t40	write MPD setup time	8.25			ns

6.5.4 Control Interface



Number	Parameter	Limit Values			Unit
		min.	typ.	max.	
R1	$\overline{POR}$ pulse width	500			ns

6.5.5 Clock Interface

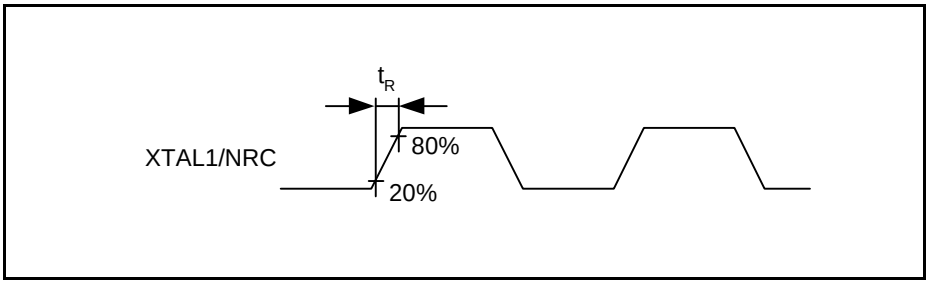


Figure 93 Clock Interface Timing Diagram

Table 49 Clock Interface AC Timing Characteristics

No.	Parameter	Limit Values			Unit
		Min	Typ	Max	
t_R	t_R: Rise time at XTAL1 and NRC			4	ns

6.5.6 JTAG Interface

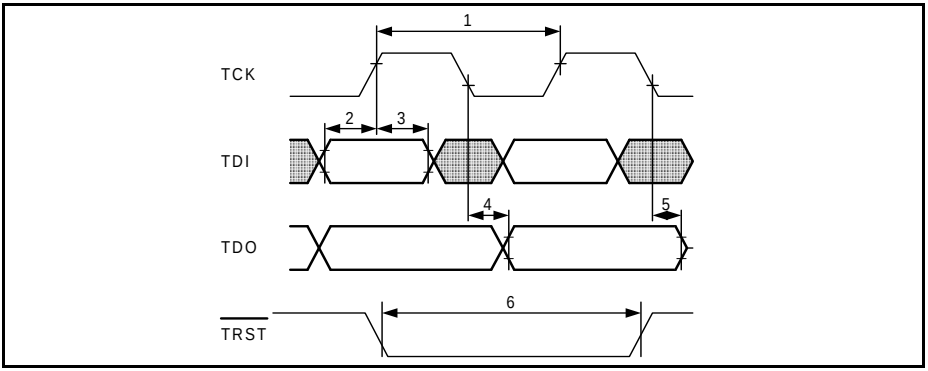


Figure 94 Boundary-Scan Test Interface Timing Diagram

Table 50 Boundary-Scan Test Interface AC Timing Characteristics

No.	Parameter	Limit Values			Unit
		Min	Typ	Max	
1	T_{TCK}: Period TCK	160			ns
1A	F_{TCK}: Frequency TCK			6,25	MHz
2	Setup time TMS, TDI before TCK rising	10			ns
3	Hold time TMS, TDI after TCK rising	10			ns
4	Delay TCK falling to TDO valid	0		30	ns
5	Delay TCK falling to TDO high impedance	0		30	ns
6	Pulse width $\overline{TRST}$ low	2 x T_{TCK}			ns

6.6 Capacitances

Parameter	Symbol	Limit Values		Unit	Notes
		min.	max.		
Input capacitance	C_{IN}		10	pF	
Output capacitance	C_{OUT}		10	pF	

7 Package Outlines

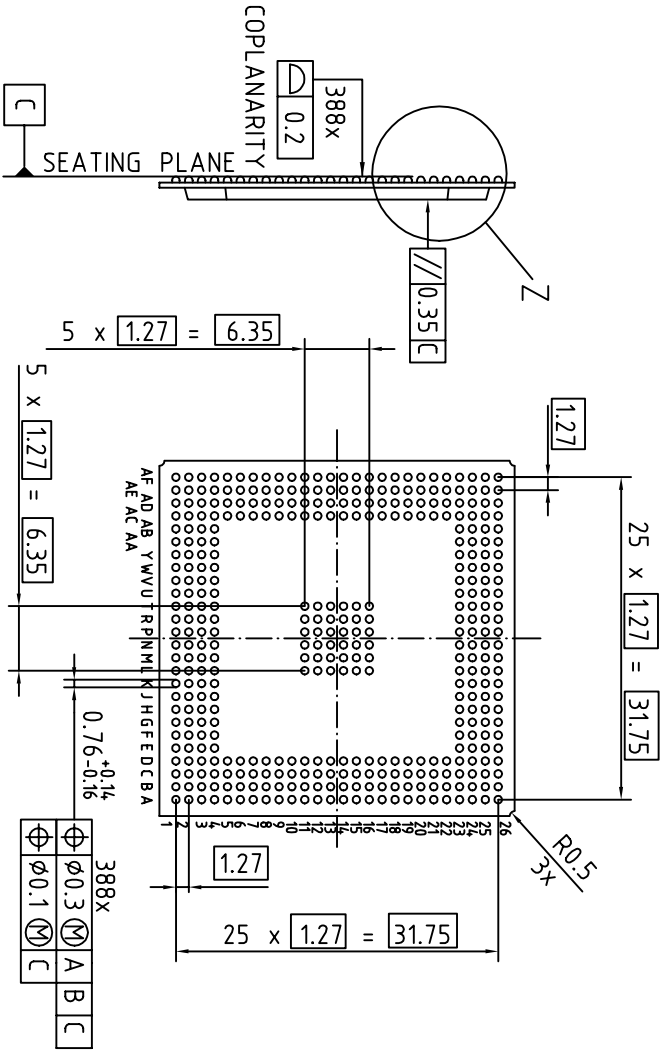


Figure 95 Package Outline, Side and Bottom View

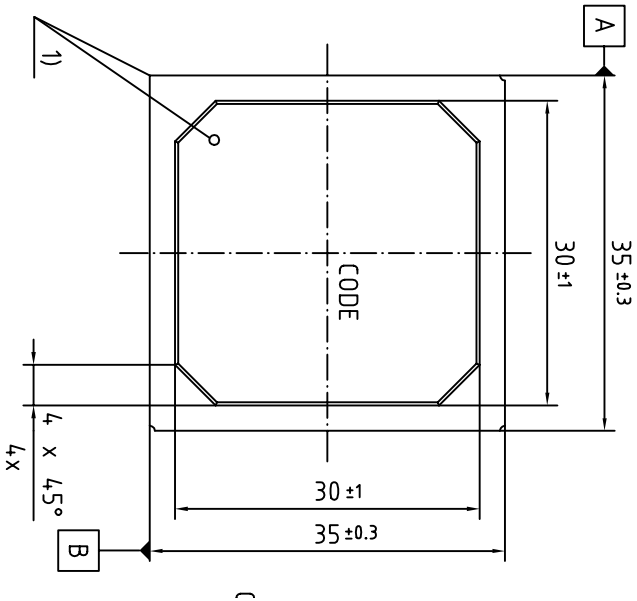


Figure 96 Package Outline, Top View

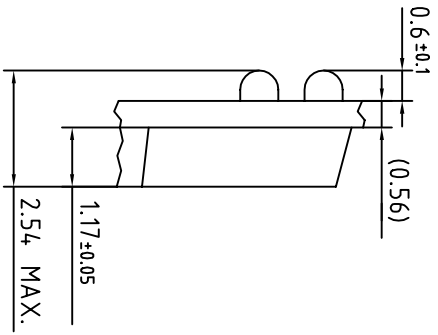


Figure 97 Package Outline, Side Detail

8 Contacts for SRTS Patent Fee

When using the device with SRTS support a patent fee for the SRTS clock recovery needs to be paid to Telcordia Technologies, Inc.:

BRUCE P. SIDRAN

Vice President & General Manager
Intellectual Capital Products

Telcordia Technologies, Inc.

331 Newman Springs Road
NVC-3Z375
Red Bank, NJ 07701-5699

Voice: (732) 758-4646
Fax: (732) 758-5418
Email: bsidran@telcordia.com

VERNON A. ANTHONY

Executive Director
Intellectual Capital Products

Telcordia Technologies, Inc.

331 Newman Springs Road
NVC-3Y389
Red Bank, NJ 07701-5699

Voice: (732) 758-5263
Fax: (732) 758-5418
Email: vanthony@telcordia.com

9 Glossary

AAL	ATM Adaptation Layer
AAL2U	AAL type 2 User
AC	Alternating Current
ACM	Adaptive Clock Method
ABR	Available Bit Rate
AF	Adaption Function
AI	Air Interface
AIS	Alarm Indication Signal
ATM	Asynchronous Transfer Mode
ATMF	ATM Forum
AVB	Average Buffer level
BGA	Ball Grid Array
BIP	Bit Interleaved Parity
B-ISDN	Broadband - Integrated Services Digital Network
BR	Backward Reporting
BSC	Base Station Controller
BTS	Base Transceiver Station
CAC	Connection Admission Control
CAS	Channel Associated Signalling
CBR	Constant Bit Rate
CC	Continuity Check
CCS	Common Channel Signalling
CDMA	Code Division Multiple Access
CDV	Cell Delay Variation
CEP	Connection End Point
CES	Circuit Emulation Service
CI	Congestion Indication
CID	Channel Identifier (voice channel identifier in AAL2 cell)
CLP	Cell Loss Priority
CODEC	Coder/Decoder

CP	Connection Point
CPE	Customer Premises Equipment
CPS	Common Part Sublayer
CPU	Central Processing Unit
CR	Clock Recovery
CRC	Cyclic Redundancy Check
CS	Convergence Sublayer
CSI	Convergence Sublayer Indication
DC	Data Collection or Direct Current
DCI	Device Control Interface
DCO	Digitally Controlled Oscillator
DEMUX	Demultiplexer
DHO	Diversity Handover
DMA	Direct Memory Access
DS1	Digital Signal 1 (1.544 Mbit/s) (=T1)
DSP	Digital Signal Processor
EP	End Point
EPD	Early Packet Discard
ESD	Electrostatic Discharge
ESF	Extended Super Frame
e-t-e	end-to-end
F4	OAM Flow on virtual path level
F5	OAM Flow on virtual channel level
FALC	Framer And Line Interface Component
FIFO	First In, First Out
FM	Forward Monitoring
FPM	Forward Performance Monitoring
FR	Frame Relay
FS/DL	Frame Sync/Data Link
FSM	Finite State Machine
FW	Firmware
GFC	Generic Flow Control

GFR	Guaranteed Frame Rate
HEC	Header Error Control
HTTP	Hypertext Transfer Protocol
HW	Hardware
ID	Identifier
IEEE	Institute of Electrical & Electronic Engineers
IMA	Invers Multiplexing over ATM
I/O	Input/Output
IP	Internet Protocol or Intermediate Point
ITU	International Telecommunications Union
ITU-T	International Telecommunications Union - Telecommunications Standardization Sector
IWU	Interworking Unit
JTAG	Joint Test Action Group
LAN	Local Area Network
LB	LoopBack
LCD	Loss of Cell Delineation
LI	Length Indication
LIC	Line Interface Card or Line Interface Circuit
LIFO	Last In First Out
LLC	Logical Link Control
LLID	Loopback Location IDentifier
LM	Layer Management
LOC	Loss Of Continuity
LOS	Loss of Signal
LSB	Least Significant Bit
LU	Line Unit
LVCMOS	Low Voltage Complementary Metal Oxide Semiconductor
LVDS	Low Voltage Differential Signal
LVTTL	Low Voltage Transistor Transistor Logic
MAC	Medium Access Control
MCSN	Monitoring Cell Sequence Number

MIB	Management Information Base
µP	Microprocessor
MPHY	Multi-PHY
MPI	MicroProcessor Interface
MSB	Most Significant Bit
MSC	Mobile Switching Center
MT	Mobile Terminal
MUX	Multiplexer
NIC	Network Interface Controller or Card
NIU	Network Interface Unit
NNI	Network-to-Network Interface
NT	Network Termination
OAM	Operation, Administration, and Maintenance
OC	Optical Carrier
OCD	Out of Cell Delineation
OCDS	On Chip Debugging System
OEP	Originating End-to-End Point
OSF	Offset Field
OSP	Originating Segment Point
P	Parity or Pointer
P-BGA	Plastic BGA
PAD	Padding
PBX	Private Branch Exchange
PCM	Pulse Code Modulation
PDH	Plesiochronous Digital Hierarchy
PDU	Protocol Data Unit
PH	Packet Header
PHY	Physical Layer Device
POTS	Plain Old Telephone Service
PP	Packet Payload
PPD	Partial Packet Discard
PPP	Point-to-Point Protocol

PT	Payload Type
PTI	Payload Type Identifier
PCR	Peak Cell Rate
PLL	Phase Locked Loop
PM	Performance Monitoring
POI	Point Of Interface
ppm	parts per million
PSTN	Public Switched Telephone Network
QID	Queue Identifier
QoS	Quality of Service
RAA	Reduced ATM Address after address reduction
RAM	Random Access Memory
RDI	Remote Defect Indication
RM	Resource Management
RNC	Radio Network Controller
RR	Round Robin
RTS	Residual Time Stamp
RX	Receive
SAR	Segmentation And Reassembly
SC	Sequence Count
SCR	Sustainable Cell Rate
SDT	Structured Data Transfer
SDU	Service Data Unit
SF	Super Frame
SN	Sequence Number
SNP	Sequence Number Protection
SP	Segment Point
SRTS	Synchronous Residual Time Stamp
SSCS	Service Specific Convergence Sublayer
STF	Start Field
STM	Synchronous Transport Module
SW	Software

TAD	Test And Debug
TAP	Test Access Port
TBD	To Be Defined
TC	Transmission Convergence
TCP	Transport Communication Protocol
TDM	Time Division Multiplexing
TEP	Terminating End-to-End Point
Timer_CU	Combined Use Timer
TM	Traffic Management
TSP	Terminating Segment Point
TUC	Total User Cell (number)
TX	Transmit
UART	Universal Asynchronous Receiver/Transmit
UBR	Unspecified Bit Rate
UDF	User Defined Field
UDP	User Datagram Protocol
UDT	Unstructured Data Transfer
UI	Unit Interval
UMTS	Universal Mobile Telecommunications System
UNI	User-to-Network Interface
UTOPIA	Universal Test and Operations Physical Interface for ATM
UUI	User-to-User Indication
VBR	Variable Bit Rate
VBR-nrt	VBR non-real time
VBR-rt	VBR real time
VC	Virtual Channel
VCC	Virtual Channel Connection
VCI	Virtual Channel Identifier
VP	Virtual Path
VPC	Virtual Path Connection
VPI	Virtual Path Identifier
WAN	Wide Area Network

WFQ	Weighted Fair Queueing
WWW	World Wide Web

10 Bibliography

1. ANSI, American National Standard for Telecommunications. Digital Hierarchy Formats Specification, T1.107-1995.
2. ANSI, B-ISDN Customer Installation Interfaces: Physical Layer Specification, Draft American National Standard for Telecommunications, T1E1.2/93 020R2.
3. ANSI, B-ISDN Network Node Interfaces and Inter-Network Interfaces: Rates and Formats Specification, Draft American National Standard for Telecommunications T1S1.5/94 001R1.
4. ATM Forum, DS1 Physical Layer Specification, Version 1.0, af-phy-0016, September 1994
5. ATM Forum: UTOPIA Specification Level 1, Version 2.01, af-phy-0017.000, March 1994
6. ATM Forum: UTOPIA Level 2 Specification, Version 1.0, ATM Forum Contribution af-phy-0039.000, June 1995.
7. ATM Forum, “E1 Physical Interface Specification”, af-phy-0064.000, September, 1996
8. ATM Forum, Inverse Multiplexing for ATM (IMA Specification, Version 1.1, af-phy-0086.001, February, 1999
9. ATM Forum, “ATM on Fractional E1/T1”, str-phy-fn64-01.00, July, 1999
10. ATM Forum, Circuit Emulation Service Interoperability Specification Version 2.0, af-vtoa-0078.000, January, 1997.
11. ATM Forum, “User-Network Interface Specification”, Version 3.1, 1994
12. ATM FORUM, “Traffic Management Specification”, Version 4.1, AF-TM-0121.000, March 1999
13. Bellcore, Generic requirement, ATM and AAL protocols, GR-1113-CORE, Issue 1, July 1994
14. Bellcore, Asynchronous Transfer Mode (ATM) and ATM Adaptation Layer (AAL) Protocols Generic Requirements, GR-1113-CORE, Issue 1, July 1994.
15. Bellcore, “Digital Cross-Connect System - Generic Requirements and Objectives”, TR-NWT-000170, Issue 2, January, 1993
16. Bellcore, B-ISDN UNI and NNI Physical Layer Generic Criteria, TR-NWT-001112, Issue 1, June 1993
17. Bellcore, Transport Systems Generic Requirements, TR-TSV-000499, Issue 4, December 1991
18. ETSI, B-ISDN ATM Adaptation Layer (AAL) Specification Type 1, prI-ETS 300353, December 1994
19. ETSI, Transmission and Multiplexing (TM); Generic Frame Structures for the Transport of Various Signals (including ATM cells) at the CCITT Recommendation. G.702 Hierarchical Rates of 2048-kbit/s, 34368-kbit/s and 139264-kbit/s; pr-ETS 300337, February 1993
20. IEEE Std 1149.1-1990, IEEE Standard Test Access Port and Boundary-Scan Architecture
21. Infineon, Data sheet: Frame and Line Interface Component, PEB 2254 (FALC)

22.ITU-T, Recommendation G.703, Physical/Electrical Characteristics of Hierarchical Digital Interfaces, Geneva 1991
23.ITU-T, Recommendation G.704, Synchronous Frame Structures used at 1544, 6312, 2048, 8488 and 44736 kbit/s Hierarchical Levels, 07/95
24.ITU-T, Recommendation G.775, Loss of Signal and Alarm Indication Signal Defect Detection Criteria for Equipment Interfaces described in Recommendation G.703 and Operating at Bit Rates described in Recommendation G.702, COM 15-R 9-E, October 1993
25.ITU-T, Recommendation G.804, “ATM Cell Mapping into Plesiochronous Digital Hierarchy (PDH)”, February 1998
26.ITU-T, Recommendation G.823, The Control of Jitter and Wander within Digital Networks which are based on the 2048-kbit/s Hierarchy, March 1993
27.ITU-T, Recommendation G.824, The Control of Jitter and Wander within Digital Networks which are based on the 1544-kbit/s Hierarchy, March 1993
28.ITU-T Recommendation I.231.10, “Circuit-mode Multiple-rate Unrestricted 8 kHz Structured Bearer Service Category”
29.ITU-T, Recommendation I.361, B-ISDN ATM Layer Specification, 11/95
30.ITU-T, Recommendation I.363.1, B-ISDN ATM Adaptation Layer specification: Type 1 AAL, 08/96
31.ITU-T, Recommendation I.363.2, B-ISDN ATM Adaptation layer specification Type 2 AAL, 09/97
32.ITU-T, Recommendation I.363.5, B-ISDN ATM Adaptation Layer specification: Type 5 AAL, 08/96
33.ITU-T Recommendation I.432 “B-ISDN user-network interface - Physical layer specification”, March, 1993
34.ITU-T Recommendation I.432.1 “B-ISDN user-network interface - Physical layer specification: General characteristics”, August, 1996
35.ITU-T Recommendation I.432.3 “B-ISDN user-network interface - Physical layer specification: 1544 kbit/s and 2048 kbit/s operation”, August, 1996
36.ITU-T Recommendation I.610 “B-ISDN operation and maintenance principles and functions“, 02/99

Infineon goes for Business Excellence

“Business excellence means intelligent approaches and clearly defined processes, which are both constantly under review and ultimately lead to good operating results.
Better operating results and business excellence mean less idleness and wastefulness for all of us, more professional success, more accurate information, a better overview and, thereby, less frustration and more satisfaction.”

Dr. Ulrich Schumacher

http://www.infineon.com

Published by Infineon Technologies AG