ATM Interface and Framer IC for ADSL Chipsets

AD6438

FEATURES

Component in AD20msp918 ADSL Chipset
Suitable for CO or Residence (ATU-R and ATU-C)
Complies with ANSI T1.413 Issue 2, ETSI TR328, ITU G.992.1 and G.992.2 Standards
ATM Operation and Functionality as Defined by Issue 2 ATM Mode and ADSL Forum
Optional Serial Interface to Access the PMD Sublayer, Available on the AD6438-1, Not Available on the AD6438
UTOPIA Level 1 and Level 2 Physical Interface
Byte-Parallel or "Multi-PHY" for Dual Latency
Implements Both Issue 1 and Issue 2 ADSL Framing Structure (Reduced Overhead)
144-Lead LQFP Package
–40°C to +85°C Operating Temperature
3.3 V Operation

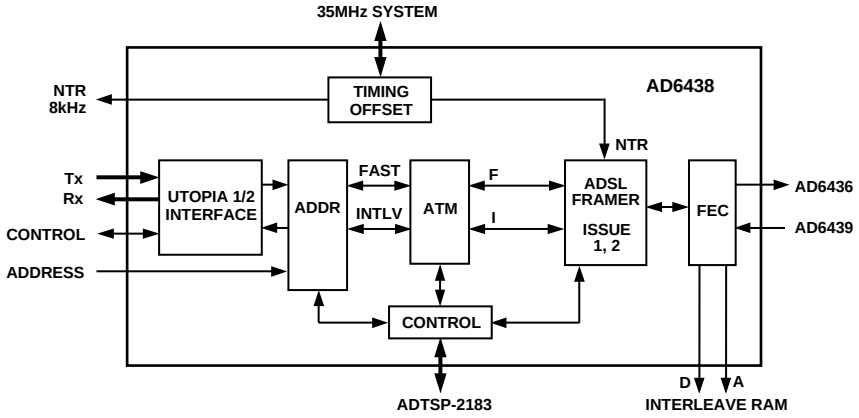
GENERAL DESCRIPTION

The AD6438 Interface and Framer IC is part of Analog Devices' AD20msp918 chipset, a flexible, standards-based solution for creating high performance ATM-based ADSL modems. The AD6438 is interoperable with the AD20msp910 ADSL chipset.

The AD6438 interfaces an ADSL modem to an external system, at either CO (Central Office) or RT (Remote Terminal) modem, using a standard UTOPIA 1 or 2 port (byte wide, cell or byte mode). It inserts ATM cells into an ADSL line, with functions that correspond to the Transmission Convergence (TC) sublayer in the BISDN protocol reference model (per ITU-T I.432 and T1.413 Issue 2). These include cell delineation, HEC generation, and cell rate adaption, as defined by the ATM working group of the ADSL forum and the RBB working group of the ATM Forum.

The AD6438 implements both standard and reduced overhead framing per ANSI T1.413 Issue 2, and includes FEC (forward error correcting code) and interleaving.

FUNCTIONAL BLOCK DIAGRAM



REV. 0

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781/329-4700 World Wide Web Site: http://www.analog.com
Fax: 781/326-8703

AD6438–SPECIFICATIONS

Parameter	Units	Comments
Absolute Maximum Serial Data Rate (Downstream)	8 Mbps	May not be achieved under normal conditions.
Absolute Maximum Serial Data Rate (Upstream)	1 Mbps	Actual performance depends on copper loop.
Supply Voltage (V_{DD})	3.3 V Nominal	
	3.0 V to 3.6 V	
Power Dissipation	600 mW	Typical
Operating Temperature (T_{AMB})	–40°C to +85°C	
Weight	1.3 g	

Parameter	Typ Value	Comments
V_{OH}	V_{DD} – 0.4 V dc	At I_{OH} = –0.5 mA
V_{OL}	0.4 V dc	
V_{IH}	2.0 V dc	
V_{IL}	1.0 V dc	
I_{IH}	±500 nA	$V_{IN} = V_{DD} = 3.6$ V
I_{IL}	±500 nA	$V_{IN} = 0$ V, $V_{DD} = 3.6$ V

ABSOLUTE MAXIMUM RATINGS

Supply Voltage . . . . . –0.3 V to +4.6 V
Input Voltage . . . . . –0.5 V to V_{DD} + 0.5 V
Output Voltage Swing . . . . . –0.5 V to V_{DD} + 0.5 V
Operating Temperature Range (Ambient) . . . –55°C to +85°C
Storage Temperature Range . . . . . –65°C to +150°C
Lead Temperature (5 sec) LQFP . . . . . +280°C

*Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; functional operation of the device at these or any other conditions above those listed in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

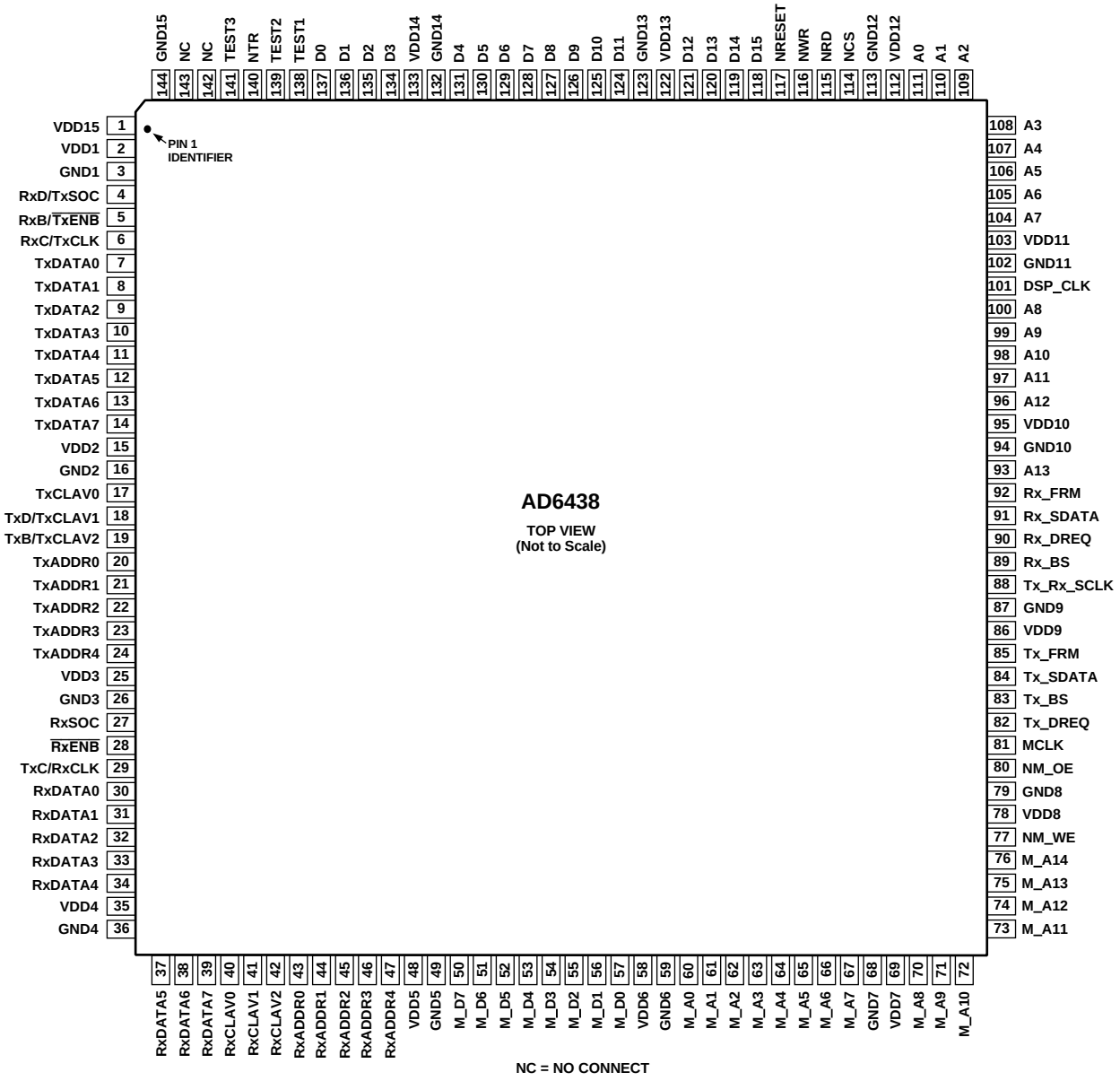
ORDERING GUIDE

Model Number	Temperature Range	Package Description	Package Option
AD6438	–40°C to +85°C	144-Lead LQFP	ST-144

CAUTION

ESD (electrostatic discharge) sensitive device. Electrostatic charges as high as 4000 V readily accumulate on the human body and test equipment and can discharge without detection. Although the AD6438 features proprietary ESD protection circuitry, permanent damage may occur on devices subjected to high energy electrostatic discharges. Therefore, proper ESD precautions are recommended to avoid performance degradation or loss of functionality.

PIN CONFIGURATION



NC = NO CONNECT

PIN FUNCTION DESCRIPTIONS

Pin Number	Pin Name	Description
1, 2, 15, 25, 35, 48, 58, 69, 78, 86, 95, 103, 112, 122, 133	VDD	Supply Power to the AD6438
3, 16, 26, 36, 49, 59, 68, 79, 87, 94, 102, 113, 123, 132, 144	GND	Supply Ground for the AD6438
4	TxSOC/RxD	UTOPIA Tx Start of Cell/Serial Mode Rx Data
5	$\overline{\text{TxENB}}$/RxB	UTOPIA Tx Enable/Serial Mode Rx Byte Strobe
6	TxCLK/RxC	UTOPIA Tx Clock/Serial Mode Rx Clock
7–14	TxDATA[0:7]	UTOPIA Tx Data
17	TxCLAV0	UTOPIA Tx Cell Available
18	TxCLAV1/TxD	UTOPIA Tx Cell Available 1/Serial Mode Tx Data
19	TxCLAV2/TxB	UTOPIA Tx Cell Available 2/Serial Mode Tx Byte Strobe
20–24	TxADDR[0:4]	UTOPIA Tx Address
27	RxSOC	UTOPIA Rx Start of Cell
28	$\overline{\text{RxENB}}$	UTOPIA Rx ENABLE
29	RxCLK/TxC	UTOPIA Rx Clock/Serial Mode Tx Clock
30–34, 37–39	RxDATA[0:7]	UTOPIA Rx Data
40–42	RxCLAV[0:2]	UTOPIA Rx Cell Available
43–47	RxADDR[0:4]	UTOPIA Rx Address
50–57	M_D[7:0]	Interleave RAM Interface
60–67, 70–76	M_A[0:14]	Interleave RAM Interface
77	NM_WE	Interleave RAM Interface Control
80	NM_OE	Interleave RAM Interface Control
81	MCLK	Master Clock
82	Tx_DREQ	Tx Data Request Provided by AD6439
83	Tx_BS	Tx Byte Strobe Provided by AD6439
84	Tx_SDATA	Serial Data for Tx Port
85	Tx_FRM	Tx Frame Strobe Provided by AD6439
88	Tx_Rx_SCLK	Serial Clock Provided by AD6439
89	Rx_BS	Rx Byte Strobe Provided by AD6439
90	Rx_DREQ	Rx Data Request Provided by AD6439
91	Rx_SDATA	Rx Serial Data Provided by AD6439
92	Rx_FRM	Rx Frame Strobe Provided by AD6439
93, 96–100, 104–111	A[13:0]	Address Bus for DSP Port
101	DSP_CLK	Clock Signal from the DSP Used to Direct Register Ports and for Accessing RAM
114	NCS	DSP Port Bus Control Pin
115	NRD	DSP Port Bus Control Pin
116	NWR	DSP Port Bus Control Pin
117	NRESET	Reset Pin
118–121, 124–131, 134–137	D[15:0]	16-Bit Data Bus from DSP Port
138	TEST1	Test Pin
139	TEST2	Test Pin
140	NTR	Network Timing Reference
141	TEST3	Pull-Down Pin with 10K Transistor
142, 143	NC	

INTRODUCTION

This data sheet describes functionality and interfacing of the AD6438 Interface and Framer IC. The AD6438 is the interface chip in the AD20msp918 ADSL chipset, connecting core transceiver functions to the external system. It also handles ADSL framing, FEC (forward error correcting code), and interleave functions.

The other components of the AD20msp918 chipset are:

- AD6439 Core DMT Signal Processor
- AD6437 Analog Front-End IC
- AD8016 Driver/Receiver
- ADTSP2183 System Control Processor

Figure 1 illustrates the basic interconnection between system components.

An object code license for all modem software is supplied with the AD20msp918 chipset.

When used as part of the AD20msp918 ADSL chipset, internal functionality of the AD6438 is under control of the firmware supplied with the ADTSP2183 and its MP (Messaging Protocol). This protocol supplies a hardware-neutral method of controlling operation of the ADSL chipset that is compatible with various hardware implementations.

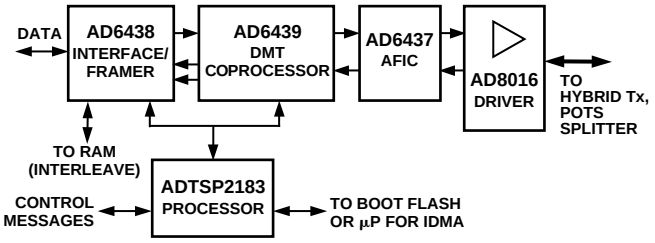


Figure 1. Block Diagram AD20msp918 Chipset

FEC BLOCK

The FEC Block of the AD6438 implements the digital manipulation and error correction portions of the ADSL system, as defined in ANSI T1.413.

The FEC Block contains two mostly independent data streams. The Tx path, used for transmitted data, receives data from the interface block and performs various operations on it, including CRC calculation, bit scrambling, Reed-Solomon checkbyte encoding and convolutional interleaving before passing it to the AD6439 signal processor. The Rx path receives serial data from the DMT and performs the reverse operations of the Tx path, including convolutional de-interleaving, Reed-Solomon error detection and correction, bit descrambling and CRC calculation and comparison. Data is then sent to the interface block.

The FEC Block supports the following code word cases:

- One code word per frame in the fast and/or interleaved data portion of a frame.
- Multiple code words per frame in the fast and/or interleaved data portion, providing the code word length evenly divides into the output (DMT) frame length.
- Multiple frames per code word on the interleaved portion of the frame only, up to 20 frames per code word. The number of checkbytes must be an integer multiple of the number of frames in the code word.
- Code words may span superframes.

The FEC Block is expected to be operated by starting out of the reset condition with the first superframe and is not meant to be online-adaptable. Communications parameters are expected to be set during initialization and not changed thereafter without resetting; any such attempt is at the discretion of the user and not guaranteed for any particular application.

Functionality

After reset, the ADTSP2183 initializes the FEC Block, loading registers to implement the following functions:

- Set up frame sizes. The frames define the amounts of raw data entering or leaving.
- Set up code word sizes that define the number of checkbytes to be appended a set of data bytes.
- Automatically or manually strobe superframes.
- Determine if wait states are needed for the Interleave RAM or RS decoder.
- Determine if multiframe code words are being used.

Once initialized, the FEC Block is activated by frame pulses, either from the DMT or via internal strobing bits. It reacts to events in every frame (e.g., reading status and acting on AOC) and every superframe (e.g., checking CRC bits and indicator bits) under the control of the ADTSP2183.

Data is transmitted and received in packets of data called frames. A frame consists of two distinct portions—fast data and interleaved data. The fast data precedes the interleaved data in the frame. Although the ADSL spec requires there to be at least one byte in each portion of the frame, the FEC Block will accept one or the other having no bytes.

The ADSL concept of a superframe, which is a specific grouping of frames, is also supported. The ADSL spec defines a superframe as 68 frames of data plus one additional sync frame, however the FEC Block does not see the sync frame, so for the purposes of this document, an ADSL superframe is 68 frames long, numbered from 0 to 67. The length of the superframe is not enforced; superframes up to 128 frames are supported.

ADSL Operations Channels (AOC)

Data is inserted and extracted from the interleaved portion of the data according to the rules in the ADSL spec:

- If there is only one byte of interleaved data (the sync byte), AOC data is inserted every frame except frame 0 in the sync byte.
- If there is more than one byte of interleaved data, the last byte of the interleaved data must be the LEX byte. For frame 0, AOC data is inserted in the LEX byte. For all other frames AOC data is inserted in the LEX byte only if the LSB of the sync byte is 1.

Indicator Bit Insertion and Extraction

Indicator bits (IB) are inserted and extracted from the fast portion of the data according to ADSL spec:

- Bits 7–0 of the indicators bits are inserted in the fast sync byte of frame 1.
- Bits 15–8 of the indicators bits are inserted in the fast sync byte of frame 34.
- Bits 23–16 of the indicators bits are inserted in the fast sync byte of frame 35.

All indicator bits except Bits 11–8 are taken from the IB_OVERRIDE_LO and IB_OVERRIDE_HI registers. Bits 11-8 are derived from certain events on the Rx channel, as reflected in the DIA_STATUS register Bits 11-8. These events indicate whether a CRC error occurred on the previous superframe, or whether there were any code words corrected during the previous superframe. Note that these events are captured during byte 0 of frame 0 of the fast data on the Rx path. The FEC and CRC indicator events for the interleaved data therefore reflect those portions of the interleaved data which overlap the fast data in the superframe. Since the interleaved data may be delayed from the fast data, this may actually represent portions of two different superframes.

All insertion and extraction of indicator bits is local to the data pump and transparent to the management entity. Bits 8–11 are handled automatically in silicon while the remaining bits are handled by the data pump software.

CRC Generation and Comparison

A Cylindrical Redundancy Check (CRC) byte is calculated independently for the fast and interleaved data on the Tx path. The CRC byte covers an entire superframe, not including byte 0 of frame 0, which is where the CRC byte for the previous superframe is inserted. The CRC calculation uses the polynomial described in the ADSL T1E1 spec, and is performed on the data after the AOC and IB bytes are inserted.

On the Rx path, a CRC byte is generated using an identical algorithm. As byte 0 of frame 0 for each portion (CRC of the fast/interleaved portions) passes through, that byte is compared to the CRC calculated on the incoming data. If these values do not compare, then the fast and/or interleaved CRC error bits are set. This also sets the CRC indicator bits, which will then be transmitted in the Tx data indicator bits. These bits may be examined and cleared. They are automatically cleared at the start of an Rx superframe (fast or interleaved as appropriate), and therefore reflect the state of the CRC comparison for the current frame. Because of the possible interleaved data delay, the CRC status for the interleaved data may lag that of the fast data.

Data Scrambling/Descrambling

Data scrambling is applied independently to the fast and interleaved data streams on the Tx path after the AOC, IB and CRC operations, but before the RS encoding. This procedure XORs each bit with the previous 18th and 23rd bit, which breaks up long strings of 0s or 1s.

Error Detection and Correction

The FEC Block supports forward error correction (FEC) using the Reed-Solomon code defined in the ADSL T1E1 spec. On the Tx path, checkbytes are calculated and appended to a defined number of bytes of data to form a code word. A code word therefore consists of a number of data bytes followed by a number of checkbytes. A code word is not intrinsically related to a frame in any way, though for the ADSL configurations a code word has an integer relationship to the frame lengths.

Code word lengths are specified separately for the fast and interleaved data, and the number of checkbytes is included in the code word length, but is also specified separately. The number of checkbytes supported by the FEC Block are even numbers from 0 to 20. The basis of the algorithm is such that the number of correctable bytes in a code word is 1/2 the number of check bytes. The maximum length of a code word is 255 bytes, so the data portion of a code word can run from 255 down to 235, depending on the number of checkbytes.

Code words for the fast data may be less than or equal to the length of a frame at the DMT side. Code words longer than a frame (multiframe code words) are not supported in the fast data. Multiframe code words are supported for the interleaved data, but there are special considerations which are covered in the next section.

On the Rx path, incoming code words are processed through the RS decoder, where the checkbytes are stripped off and used to determine if an error has occurred during transmission. This requires buffering an entire code word, since if there is a transmission error, the data may be correctable. This buffering inherently makes the Rx path slower than the Tx path in terms of overall data throughput. In addition, the RS decoding uses an algorithm which has to iterate over the entire code word using roughly the same number of computational iterations as there are bytes in the code word.

Code words with errors may be correctable if the number of errors is less than or equal to half the number of checkbytes. These corrections are made as the code word is unloaded to the ADSL framing operations stage (descrambler, etc.). Correctable code words also set a bit in the DIA_STATUS register, and for each superframe these bits are saved as part of the indicator bits which will be transmitted back on the Tx path. Code words that have too many errors are considered uncorrectable, and no correction is attempted. These set a Noncorrectable Code word (NCC) error bit in the DIA_STATUS which remains set until cleared via the RESET register. This is logged and controlled by the supervisory system (and then the MP interface). It does not affect the indicator bits.

The associated FEC_CTRL register contains two sets of fields. The first allows the number of errors considered correctable to be reduced. The second tailors the number of clock cycles the RS decoder engine waits during each iteration. These should be changed only if it is suspected that the RS decoder results are bad due to the engine not having enough time to compute each iteration. Additional wait states decrease overall throughput on the Rx path.

Special Considerations for Multiframe Code Words

If multiframe code words are to be used, there are a number of special considerations:

- Multiframe code words are only valid on the interleaved portion of the frame.
- The number of checkbytes must be a multiple of the number of frames in the code word. This limits a multiframe code word to 16 frames.

Convolutional Interleaving/Deinterleaving

Convolutional interleaving in the AD6438 is the process of spreading data from one code word in the transmitted stream across a number of code words. This enhances the likelihood of the correction of a large error burst during transmission, since the error will, after deinterleaving on the Rx path, be fragmented across multiple code words, each of which has a better likelihood of correcting the smaller error fragment.

Interleaving is provided only for the interleaved portion of the data, and is performed on the data as it leaves the RS encoder/MFC buffering stages. Interleaving is achieved by writing data in one pattern to a block of memory in an external RAM, and reading the data back in a different pattern.

Framing

The 918 chipset implements full and reduced overhead framing modes as defined in the ANSI T1.413 specification, Issue 2. The ME selects the desired framing mode using Bits 24 and 25 of the OPTN.utopia CMV.

To maintain backward compatibility with the 910 chipset (which operates in framing mode 0), the 918 chipset will receive and decode stuffed and robbed data when configured to operate in framing mode 1. The ME selects the path to be used (fast or interleaved) during link initiation. If a rate adaptive link is established, the interleaved path is used on both downstream and upstream. If a fixed rate link is established, the ME can select either fast or interleaved path for both downstream and upstream. The AD6438 supports framing modes 1, 2 and 3 whether it is operating in ATM or serial mode.

Finally, the ME uses Bit 26 of the OPTN.utopia CMV to select cell or octet mode. These modes of ATM data transfer are defined in the UTOPIA Level 1 and Level 2 specifications.

Network Timing Reference (NTR)

The network timing reference is an 8 kHz signal input at the ATU-C, transmitted over the ADSL physical layer, recovered at the ATU-R, and output as an 8 kHz phase locked output signal at the ATU-R (see Figure 2).

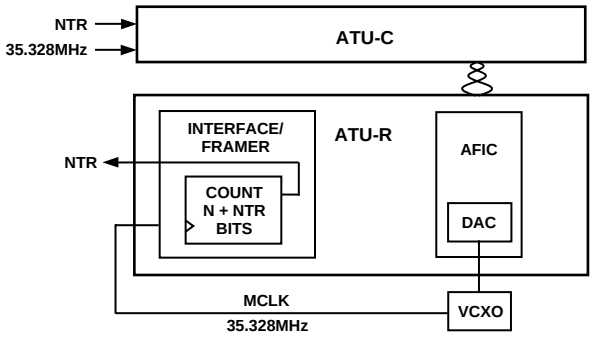


Figure 2. NTR

Higher layers in systems such as Circuit Emulation (CES), Voice and Telephony Over ATM (VTOA), and Desktop Video Conferencing (DVC) require a reference clock to guarantee end-to-end synchronization of transmit and receive sides.

Table I. Allocation of Overhead Bytes

Frame Number	Overhead Bytes Fast Byte[1]	Overhead Bytes Sync Byte[1]	Fast or Sync Byte[2]
0	CRC	CRC	CRC
1	IB_0	AOC	IB_0
34	IB_1	AOC	IB_1
35	IB_2	AOC	IB_2
4n and 4n+1, n Not Equal to 0	EOC_1 and _2	AOC	AOC
4n+2 and 4n+3, n Not Equal to 8	EOC_1 and _2	AOC	EOC_1 and _2

NOTES
[1]If no EOC message is sent in these frames, then the stuffing pattern XX0011X0 is inserted in both frames. The MSB is left and X is undefined.
[2]EOC_0 byte in even frame number, EOC_1 byte in odd frame number.

Table II highlights the indicator bits that have been added for ATM. In the AD6438, the indicator bits are written into the FEC by the ADTSP-2183.

Table II. Definition of Indicator Bits

Indicator Bit	Definition	Indicator Bit	Definition	Indictor Bit	Definition
ib23	ntr3	ib15	ncd-ni	ib7	Reserved*
ib22	ntr2*	ib14	ncd-i	ib6	Reserved
ib21	ntr1*	ib13	rdi	ib5	Reserved
ib20	ntr0*	ib12	los	ib4	Reserved
ib19	reserved	ib11	fecc-ni	ib3	Reserved
ib18	reserved	ib10	febe-ni	ib2	Reserved
ib17	hec-ni	ib9	fecc-i	ib1	Reserved
ib16	hec-i	ib8	febe-i	ib0	Reserved

*All reserved bits are forced to zero or the inactive state.

Refer to ANSI T1.413 or Chipset Users Guide for further details.

The ATU-C generates a modem-synchronous 8 kHz timing reference (MTR) and continuously measures frequency offset between the NTR and MTR. The offset is inserted in the overhead data once per 17 ms superframe. The ATU-R restores the NTR from the modem's downstream clock and the received NTR frequency offset.

Information Format

The NTR to MTR frequency offset is measured in cycles of the 2.208 MHz modem clock. A nine bit phase offset results from the phase offset measurement. The phase offset is measured on a superframe basis. A 100 ppm (10^{-4}) frequency offset between the NTR and the modem clock is considered worst case value and results in a phase offset average of 3.75 over one superframe (with possible fold-over of Δ phase to 0 or 275). The phase offset variation (thus frequency offset) over one superframe is therefore coded into a 4-bit signed integer representation of Δf.

The NTR frequency offset is mapped into four overhead bits. The MSB of the 4-bit phase offset value is ntr0 with ntr3-ntr0 representing a signed integer in the –8 to +7 range. The bits ntr3-ntr0 shall be carried in the indicator bits as defined in the overhead bit mapping table.

If the ATU-C chooses to lock the modem's downstream clock to the incoming 8 kHz timing marker, the frequency offset is set to zero.

The NTR signal on Pin 140 is an input signal when the AD6438 is configured for the CO end and an output signal when the AD6438 is configured for the RT end.

At the CO end, the 8 kHz network timing reference clock is input to the NTR pin. At the RT end the NTR output is an 8 kHz clock locked to the clock at the CO end. When not used, this pin should be tied to a pull-up resistor at the CO end; it can be left unconnected at the RT end.

INTERFACE BLOCK

The AD20msp918 chipset supports ATM (Asynchronous Transfer Mode) data transfer over a UTOPIA interface compliant with the ATM Forum Committee UTOPIA Level 1 and Level 2 specifications. Available configuration options are set in the CMV (Configuration Management Variable) OPTN.utopia (see Table IV).

ATM Block

The physical layer of the ATM interface for ADSL is separated into two sublayers, the Transmission Convergence (TC) sublayer and the Physical Medium Dependent (PMD) sublayer (see Figure 3, note that information on the higher level ATM layer that interfaces to the AD6438 is included for reference purposes).

Characteristics of the PMD sublayer:

- Medium, line code, connectors.
- Uses existing standards and technology (ANSI T1.413 ADSL standard).

Characteristics of the TC sublayer:

- Cell delineation in accordance with I.432 (see Figure 6).
- Cell rate decoupling (insert idle cells when there is no data available).

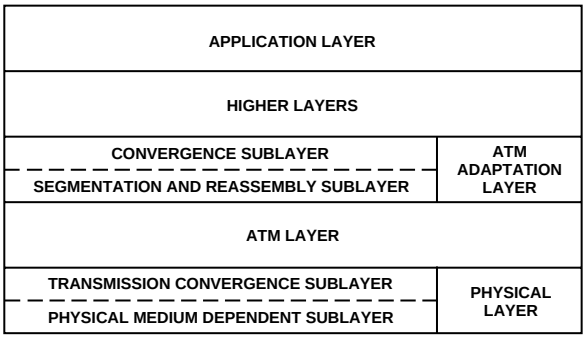


Figure 3. Protocol Stack

Figure 4 shows how the AD6438 and the ADSL chipset fit into the ATM Protocol Stack.

Transmission Convergence Sublayer

The transmission convergence (TC) sublayer is shown in Figure 4 and the associated data paths in Figure 5. The TC sublayer interface structure is a continuous stream of cells, each containing 53 octets.

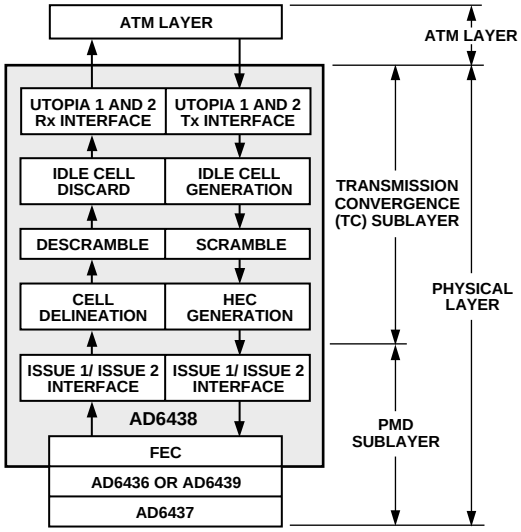


Figure 4. Protocol Stack—Single Latency

Transmit Path

This section and the block diagram in Figure 5 describe the AD6438 transmit paths.

Signals for UTOPIA Level 1 and Level 2 Tx interfaces are supported as described in the UTOPIA section. This block contains rate matching buffers or FIFOs. The selection of UTOPIA Level 1 and Level 2, the MPHY address, and octet/cell mode are done through the mode programmable registers.

The buffer is a 512 × 8 DPSRAM (dual port SRAM) that can be dynamically allocated between the two latency paths.

Idle Cell Generation

Idle cells are inserted and discarded to adapt the cell rate to the ADSL line data rate. Idle cells are identified by the standard pattern shown in Table III.

Scrambler
The 48 byte payload may optionally be scrambled. The self-synchronizing scrambler polynomial $X^{43} + 1$ (designed for the SDH-based physical layer) improves the security and robustness of the HEC cell delineation mechanism and randomizes the data in the information field for possible improvement of transmission performance. During the five octet header, the scrambler operation is suspended and the scrambler state retained.

HEC Generation
HEC (Header Error Control) is an error correction code covering the entire cell header that is calculated across the four bytes of the header and can detect multiple header errors. (Note: Error correction is not needed in ADSL; it primarily provides protection against the incorrect delivery of cells that have a corrupted address. Due to the processing already going on, single bit errors are not common).

The HEC field employs an 8-bit sequence, the remainder of the division (modulo 2) of the product [x^8 multiplied by the content of the header excluding the HEC field] by the generator polynomial [$x^8 + x^2 + x + 1$]. At the transmitter, the content of the register of the device computing this sequence is preset to all 0s and is then modified by division of the header excluding the HEC field generator polynomial. The resulting remainder is transmitted as the 8-bit HEC.

To significantly improve cell delineation performance in the case of bit-slips, the following recommendation by ITU-T I.432 is implemented for ADSL ATM:

1. The check bits calculated by the use of the check polynomial are added (modulo 2) to the 8-bit pattern "01010101" before being inserted in the last octet of the header (the left bit is the most significant bit).
2. The receiver must subtract the same pattern from the eight HEC bits (equal to add modulo 2) before calculating the syndrome of the header.

Receive Path
The receive interface delivers data in UTOPIA Level 1 or Level 2 format (see Figure 5).

Two programmable addresses specify the address locations of the two latency paths.

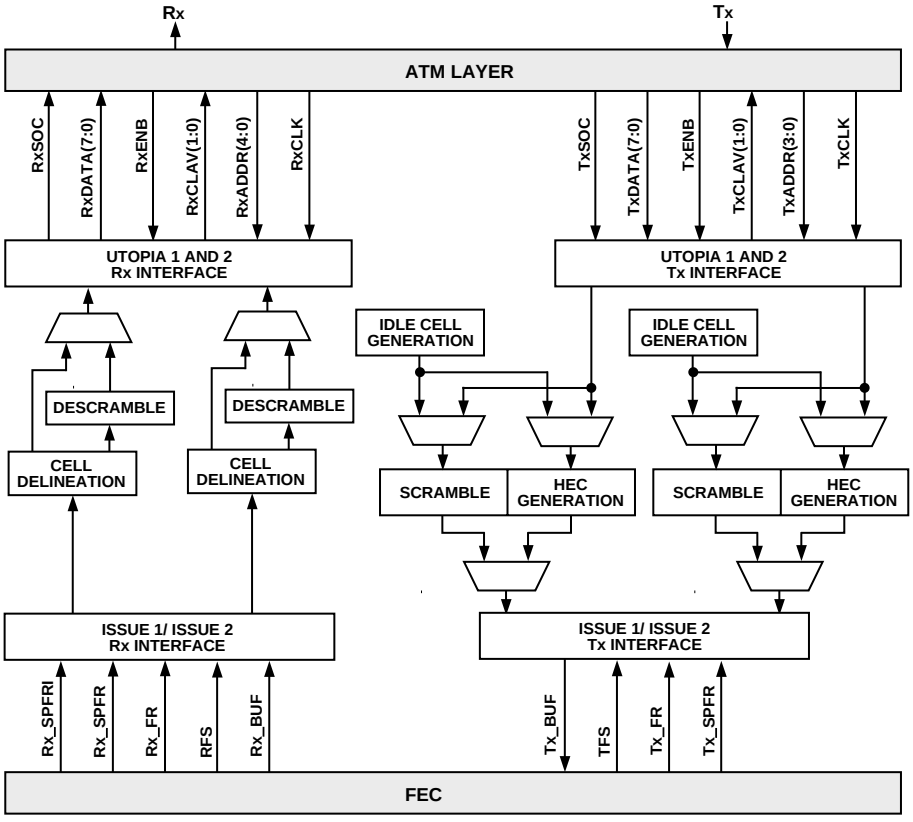


Figure 5. Data Path for Tx and Rx, TC Sublayer

Table III. Idle Cell Format

Header 1	Header 2	Header 3	Header 4	Header 5	48 Data Bytes (All Have Same Value)
00000000 0x00	00000000 0x00	00000000 0x00	00000001 0x01	01010010 0x52	01101010 0x6A

Cell Delineation

Figure 6 shows how the process of finding cell boundaries is achieved by finding the HEC field in the data stream. The receiver is in one of three states:

1. In the HUNT state, the delineation process is performed by checking bit by bit for correct HEC. Correct HEC occurs when the syndrome equals zero for the assumed header field. Once an agreement is found, it is assumed that a header has been found and the PRESYNC state is entered.
2. In the PRESYNC state, cell delineation is performed on every consecutive 53 byte cells. The process repeats until the correct HEC has been confirmed DELTA times consecutively. If an incorrect HEC is found, the process returns to the HUNT state.
3. In the SYNC state, cell delineation is assumed to be lost if an incorrect HEC is obtained ALPHA times consecutively.

NOTES
1. Cells received with bad HEC are discarded.
2. F3 cells are passed through the AD6438.

Idle cells cause no action at the receiving node except for cell delineation, including HEC verification.

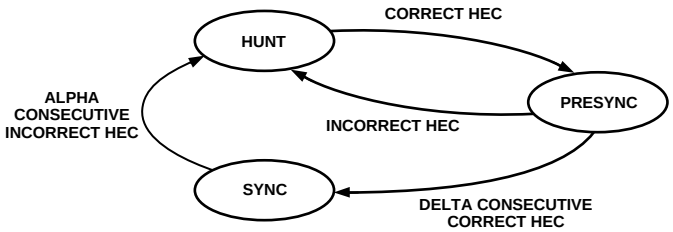


Figure 6. Cell Delineation State Diagram

Descrambler

While in the HUNT state of cell delineation, the descrambler is disabled. In PRESYNC and SYNC states, the descrambler is enabled for a number of bits equal to the length of the information field, and again disabled for the following assumed header. The self-synchronizing scrambler polynomial X^{43} +1 is implemented.

UTOPIA Rx Interface

Utopia defines the interface between the ATM layer and the TC sublayer. Signals for the UTOPIA Level 1 and Level 2 Rx Interfaces are supported as described in the UTOPIA Signals section.

UTOPIA Interface

The AD20msp918 chipset supports interfaces to a single physical layer (single PHY or SPHY) for Utopia Level 1 and extended Level 1 operation or multiple physical layers (multiple PHY or MPHY) for Level 2 operation. The ME can select SPHY or MPHY operation using Bit 22 of the OPTN.utopia CMV (see Table IV). In MPHY mode, port addressing is enabled and the required ATM port must be addressed in the standard way (a combination of CLAV, enable and address line activity) for it to be selected to transmit and/or receive ATM traffic.

To support UTOPIA Level 2 operation, the ME can configure the ATM interface with addressing information. The transmit and receive ports on both the fast and interleave paths are independently configurable with a 5-bit address. Normally the Tx and Rx portions of an ATM port have the same address; this configuration is handled using Bits 0 to 19 of the OPTN.utopia CMV. These bit fields are only effective if Bit 22 is set (MPHY mode). When operating in MPHY mode using port addressing, address polling is supported according to the UTOPIA Level 2 specification.

The ME can use Bits 20 and 21 of the OPTN.utopia CMV to configure the CLAV (cell available) lines. Bit 20 maps the fast and interleave paths to the two CLAV lines available on the chipset. Bit 21 is used to select single or multiCLAV mode. In single CLAV (SCLAV) mode, the CLAV0 line is three-stated if the ATM interface is not addressed (MPHY Level 2 mode is assumed). In multiple CLAV (MCLAV) mode, the CLAV0 line drives all the time (like CLAV1 and CLAV2), regardless of whether the ATM interface is addressed (for MPHY Level 2 mode) or enabled (for SPHY Level 1/extended Level 1 mode). Note that MCLAV mode is applicable to both Level 1 and Level 2 UTOPIA operation. In Level 2 operation, MCLAV operation is often referred to as “direct status polling.”

Table IV. OPTN.utopia

Bit	Name	Function
4:0	Tx Fast Path Address	Configurable Address Setting for Tx Fast Path
9:5	Rx Fast Path Address	Configurable Address Setting for Rx Fast Path
14:10	Tx Intl Path Address	Configurable Address Setting for Tx Interleave Path
19:15	Rx Intl Path Address	Configurable Address Setting for Rx Interleave Path
20	Cell available (CLAV) Config.	0 – CLAV[0] Maps to Fast Path, CLAV[1] Maps to Interleave Path 1 – CLAV[1] Maps to Fast Path, CLAV[0] Maps to Interleave Path
21	Single/Multiple CLAV	0 – Multiple CLAV Mode 1 – Single CLAV Mode
22	Single/Multiple PHY	0 – Multiple PHY Mode (Multiple Physical Connections)
23	ATM/Serial	0 – ATM 1 – Serial
25:24	Framing Mode	00 – Framing Mode 0 (Serial Mode) 01 – Framing Mode 1 (Full Overhead) 10 – Framing Mode 2, Reduced Overhead, Separate Fast and Sync Bytes 11 – Framing Mode 3, Reduced Overhead, Merged Fast and Sync Bytes
26	Cell/Octet Mode	0 – Cell Mode 1 – Octet Mode
31:27		Reserved

Table V. UTOPIA Modes vs. Bits 22, 21

PHY (Bit 22)	CLAV (Bit 21)	UTOPIA Mode
0	0	Level 2 – Direct Status Indication
0	1	Level 2 – Address Polling
1	0	Level 1 – Default
1	1	Not Used

UTOPIA Signals

Table VI describes the UTOPIA Tx signals. Figure 5 shows a diagram of the Tx path.

Table VI. UTOPIA Tx Signals

Signal	Description
TxSOC	Start of Cell. Active high signal asserted by the ATM layer when TxDATA contains first valid cell byte.
TxDATA[7:0]	Data. Byte-wide true data driven from ATM to PHY layer. TxDATA[7] is the MSB.
$\overline{\text{TxENB}}$	Enable. Active low signal asserted by ATM layer during cycles when the TxDATA contains valid cell data.
TxCLAV[0]/$\overline{\text{TxFULL}}$	Level 1—Full/Cell Available. For octet-level flow control, $\overline{\text{TxFULL}}$ is an active low signal from the PHY to the ATM layer, asserted by the PHY layer to indicate a maximum of four more transmit data writes are accepted. For cell-level flow control, TxCLAV is an active high signal from the PHY to the ATM layer, asserted by the PHY layer to indicate it can accept the transfer of a complete cell. Level 2—Cell Available. For cell-level control in an Multi-PHY (MPHY) environment, TxCLAV is an active high three-state signal from the MPHY device to the ATM layer. A polled MPHY device (port) drives TxCLAV only during each cycle following one with its address on the TxADDR lines. The polled MPHY device (port) asserts TxCLAV high to indicate it can accept the transfer of a complete cell, otherwise it deasserts the signal. Note the change in the label for the signal to TxCLAV[0].
TxCLAV[1] (Level 2 Only)	For Multi-PHY Transmit Interface. • Level 1—Not used. • Level 2—Additional Cell Available Signal identical in functionality to the TxCLAV[0] signal. A PHY device (as opposed to a PHY port) may include a total of up to four TxCLAV signals corresponding four PHY ports, which may be used either for direct status indication or for multiplexing status polling. For the dual latency we have two ports so we need two CLAV signals.
TxADDR[4:0] (Level 2 Only)	Required additional signals for the Multi-PHY Transmit Interface • Level 1—Pull pins low. • Level 2—Address. Five-bit wide data driven from the ATM to MPHY layer to poll and select the appropriate MPHY device (port in presence of multiple TxCLAV signals). TxADDR[4] is the MSB. Each MPHY device needs to maintain its address. This value should be programmed through the management interface. The value for the Tx and Rx portions of an MPHY device (port in presence of multiple TxCLAV signals) should be identical. Address 31 indicates a null PHY port.
TxCLK	Data transfer/synchronization clock provided by the ATM layer to the PHY layer for synchronizing transfers on TxDATA.

Table VII describes the UTOPIA Rx signals that are shown in Figure 5.

Table VII. UTOPIA Rx Signals

Signal	Description
RxSOC	Start of Cell. Active high signal asserted by the PHY layer when RxDATA contains the first valid byte of the cell. RxSOC is three-state, enabled only in cycles following those with the $\overline{\text{RxENB}}$ asserted.
RxDATA[7:0]	Data. Byte-wide true data driven from ATM to PHY layer. RxDATA[7] is the MSB. RxDATA is three-state, enabled only in cycles following those with $\overline{\text{RxENB}}$ asserted.
$\overline{\text{RxENB}}$	Enable. Active low signal asserted by ATM layer to indicate that RxDATA and RxSOC are sampled at the end of the next cycle. In multiple PHY configurations, $\overline{\text{RxENB}}$ is used to three-state RxDATA and RxSOC PHY layer outputs. RxDATA and RxSOC should be enabled only in cycles following those with $\overline{\text{RxENB}}$ asserted.
RxCLAV[0]/ RxEMPTY	• Level 1—Full/Cell Available. For octet-level flow control, RxEMPTY is an active low signal from the PHY to the ATM layer, asserted by the PHY layer to indicate that in the current cycle there is no valid data for delivery to the ATM layer. For cell-level flow control, RxCLAV is an active high signal from the MPHY to the ATM layer, asserted by the PHY layer to indicate it has a complete cell available for transfer to the ATM layer. In both cases this signal indicates cycles when there is valid information on RxDATA/RxSOC. • Level 2—Cell Available. For cell-level control in an Multi-PHY (MPHY) environment, RxCLAV is an active high three-stateable signal from the MPHY device to the ATM layer. A polled MPHY device (port) drives RxCLAV only during each cycle following one with its address on the RxADDR lines. The polled MPHY device (port) asserts RxCLAV high to indicate it has a complete cell available for transfer to the ATM layer, otherwise it deasserts the signal.
RxCLAV[1] (Level 2 Only)	For Multi-PHY Transmit Interface. • Level 1—Not used. • Level 2—Additional Cell Available Signal, identical in functionality to the RxCLAV[0] signal. A PHY device (as opposed to a PHY port) may include a total of up to four RxCLAV signals corresponding to four PHY ports, which may be used either for direct status indication or for multiplexing status polling. For the dual latency there are two ports so two CLAV signals are needed.
RxADDR[4:0] (Level 2 Only)	Required additional signals for the Multi-PHY Transmit Interface. • Level 1—Pull pins low. • Level 2—Address. Five-bit wide true data driven from the ATM to MPHY layer to poll and select the appropriate MPHY device (port in presence of multiple RxCLAV signals). RxADDR[4] is the MSB. Each MPHY device needs to maintain its address. This value should be programmed through the management interface. The value for the Tx and Rx portions of an MPHY device (port in presence of multiple TxCLAV signals) should be identical. Address 31 indicates a null PHY port.
RxCLK	Data transfer/synchronization clock provided by the ATM layer to the PHY layer for synchronizing transfers on RxDATA.

UTOPIA LEVEL 1 INTERFACE

Within Utopia, the direction from the ATM layer to the physical layer is defined as the Transmit direction and the direction from the physical layer to the ATM layer as the Receive direction. Signals that pass between layers are shown in Figure 7.

When operating in this mode, the AD6438 supports only one data flow, interleaved or fast, and transfers one byte per clock cycle, transforming cells in 53 clock cycles. Independent clocks generated by the ATM layer chip synchronize transmit and receive (see timing diagrams Figure 8, Figure 9, and Figure 10).

If $\overline{\text{RxENB}}$ is asserted, the AD6438 delivers data from its internal FIFO to RxDATA and RxSOC at each positive transition of RxCLK. If $\overline{\text{TxENB}}$ is asserted, the AD6438 samples TxDATA and TxSOC during the positive transition of TxCLK.

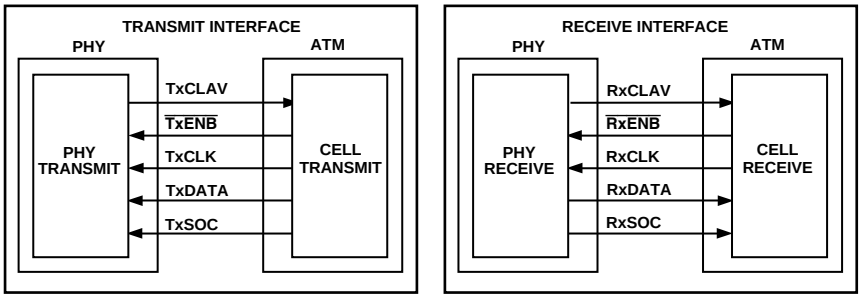


Figure 7. ATM to PHY Interconnection (Level 1)

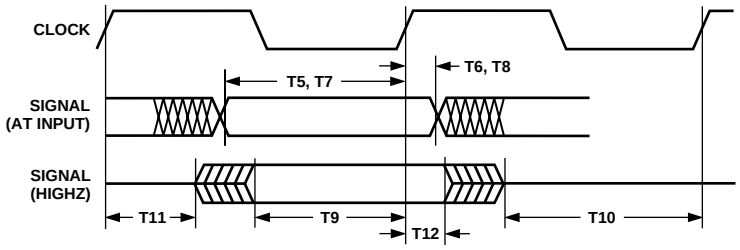


Figure 8. Utopia 1 Timing Specification

Table VIII. TxCLK, RxCLK Characteristics

Symbol	Parameters	Min	Max	Unit
F	Clock Frequency	1.5	25	MHz
Tc	Clock Duty Cycle	40	60	%
Tj	Clock Peak-to-Peak Jitter		5	%
Trf	Clock Rise Fall Time		4	ns
L	Load		100	pF

Table IX. TxDATA, TxSOC Characteristics

Symbol	Parameters	Min	Max	Unit
T5	Input Set-Up Time to TxCLK	10		ns
T6	Hold Time to TxCLK	1		ns
L	Load		100	pF

Table X. RxDATA, RxSOC, RxCLAV Characteristics

Symbol	Parameters	Min	Max	Unit
T7	Input Set-Up Time to TxCLK	10		ns
T8	Hold Time to TxCLK	1		ns
T9	Signal Going Low Impedance to RxCLK	10		ns
T10	Signal Going High Impedance to RxCLK	0		ns
T11	Signal Going Low Impedance to RxCLK	1		ns
T12	Signal Going High Impedance to RxCLK	1		ns
L	Load		100	pF

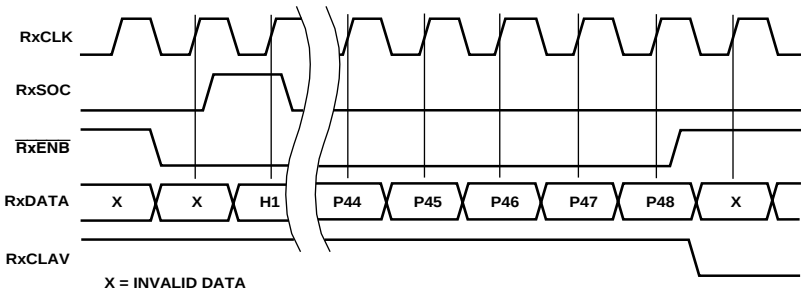


Figure 9. Timing—UTOPIA 1 Receive Interface

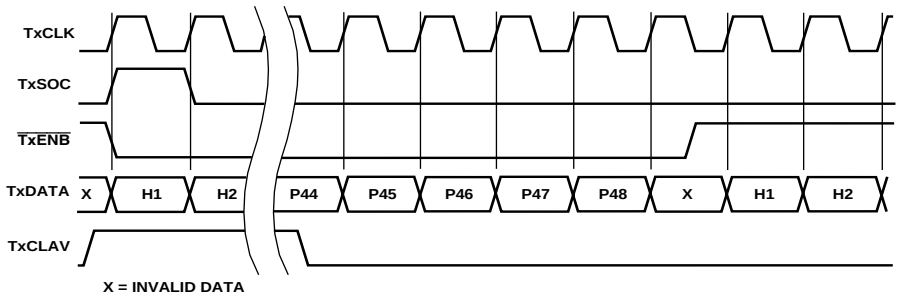


Figure 10. Timing—UTOPIA 1 Transmit Interface

TRANSMIT OPERATION

Octet-Level Handshake

When $\overline{\text{TxENB}}$ is asserted, the PHY layer stores data from TxDATA on the positive transition of TxCLK for a time period called the "transmit window." It starts when the PHY layer deasserts $\overline{\text{TxFULL}}$ to indicate it can accept data and lasts until four valid write cycles after the PHY layer asserts $\overline{\text{TxFULL}}$ (which can happen at any time). Once asserted, the ATM layer can transfer a maximum of four data words on TxDATA until $\overline{\text{TxFULL}}$ is deasserted again.

$\overline{\text{TxENB}}$ must be deasserted within four data writes of $\overline{\text{TxFULL}}$ assertion and not reasserted until deassertion of $\overline{\text{TxFULL}}$ is detected (this allows the ATM layer to track the cycles during which data was transferred on TxDATA). The PHY layer considers any assertion of $\overline{\text{TxENB}}$ outside the transmit window to be an error, and ignores it.

Octet-level handshaking between the PHY and ATM layer are shown in Figure 11 and Figure 12.

In Figure 11, the ATM layer recognizes that $\overline{\text{TxFULL}}$ was asserted by the PHY on clock edge 2 and therefore deasserts $\overline{\text{TxFULL}}$ after four additional write cycles, at clock edge 6. The ATM layer detects $\overline{\text{TxFULL}}$ deasserted on clock edge 9 and continues transmitting valid data by asserting $\overline{\text{TxENB}}$ and driving P47 on TxDATA. The ATM layer deasserts $\overline{\text{TxENB}}$ at clock edge 12 to stop data transmission since it has no valid data to transmit, then resumes transmission by driving $\overline{\text{TxENB}}$ low again on clock edge 13. Note: This example assumes a fast ATM layer that minimizes the time for $\overline{\text{TxENB}}$ to react to $\overline{\text{TxFULL}}$; a slower ATM layer might require more clock cycles to assert $\overline{\text{TxENB}}$.

For the example in Figure 12, the ATM layer starts data transmission after detecting $\overline{\text{TxFULL}}$ deasserted at clock edge 3. It detects $\overline{\text{TxFULL}}$ asserted on clock edge 6 and prepares to deassert $\overline{\text{TxENB}}$ on clock edge 10, but before transmission interruption can take place detects $\overline{\text{TxFULL}}$ deasserted again on clock edge 9 and therefore keeps $\overline{\text{TxENB}}$ asserted. Note: This situation occurs in theory; actual implementations usually deassert $\overline{\text{TxENB}}$ for a few clock cycles.

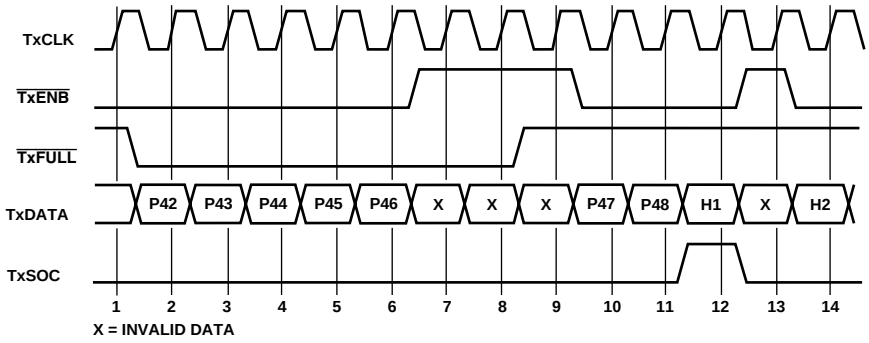


Figure 11. Octet-Level Handshaking (1)

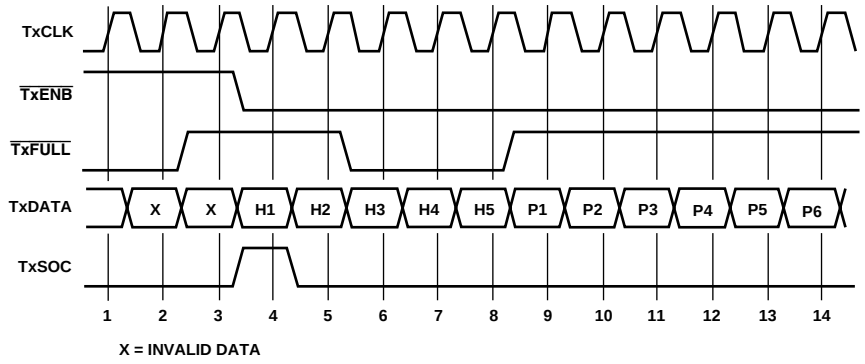


Figure 12. Octet-Level Handshaking (2)

Cell-Level Handshake

The only difference between cell-level and octet-level handshake is that in cell-level handshake the PHY layer accepts the transfer of a whole cell when TxCLAV is asserted. As for octet-level handshake, the ATM layer can use $\overline{\text{TxENB}}$ to control the flow of data at an octet level. Four cycles before the end of a cell, the PHY layer deasserts TxCLAV if it cannot accept the immediate transfer of the subsequent cell. This prevents the ATM layer from causing transmit overrun. In practice, If a PHY device using cell level handshake is connected to a byte-based ATM device, it is suggested that the TxCLAV signal should remain asserted until four cycles before the end of a cell.

Examples of cell-level handshaking between the PHY and ATM layers are shown in Figure 13 and Figure 14.

In Figure 13, the ATM layer recognizes on clock edge 2 that the PHY has asserted TxCLAV and starts transmitting a complete cell.

Four cycles before the end of the cell (clock edge 51), the PHY indicates it cannot accept another cell and the ATM layer deasserts $\overline{\text{TxENB}}$ on clock edge 55. Once it indicates that the PHY can accept a cell, TxCLAV must stay asserted until the PHY confirms that the ATM layer has transmitted payload byte 43 of the cell.

In Figure 14, the PHY indicates on clock edge 3 that it can accept another cell from the ATM layer, so the ATM layer begins transmitting the next cell after P48. Like with octet-level-handshake, the ATM layer can interrupt data transmission at any time by deasserting $\overline{\text{TxENB}}$ (shown here between clock edges 11–12).

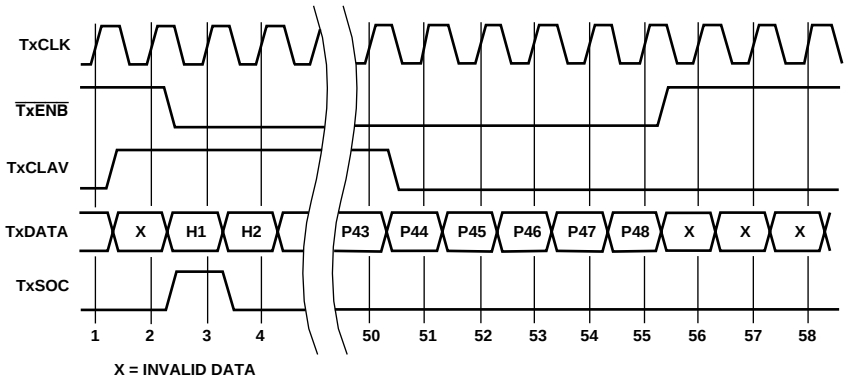


Figure 13. Cell-Level Handshaking (1)

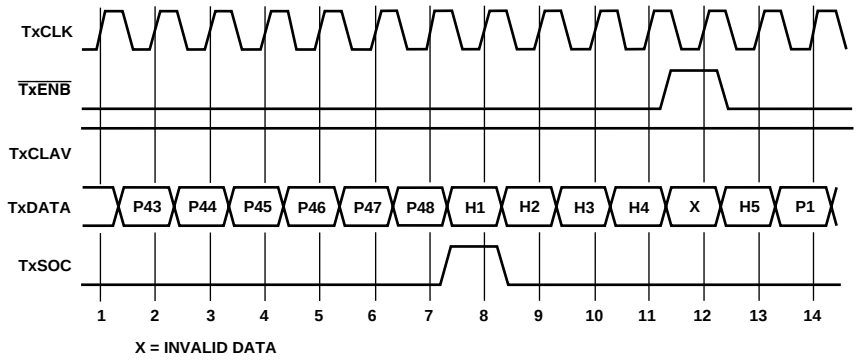


Figure 14. Cell-Level Handshaking (2)

RECEIVE OPERATION

Octet-Level Handshake

The PHY Layer deasserts RxEMPTY to indicate it has valid data. To show that it wants to read PHY data, the ATM layer asserts $\overline{\text{RxENB}}$ for a period of time known as a "read window." During this time, the PHY layer presents data read from its internal FIFO at RxDATA/RxSOC on positive transitions of RxCLK. RxEMPTY and $\overline{\text{RxENB}}$ can be asserted and deasserted at any time ($\overline{\text{RxENB}}$ can even be permanently asserted). Asserting $\overline{\text{RxENB}}$ while RxEMPTY is asserted produces undefined values of RxDATA.

In Figure 15, the ATM layer recognizes at clock edge 2 that RxEMPTY was deasserted by the PHY layer so it asserts $\overline{\text{RxENB}}$.

On clock edge 3, the PHY layer detects $\overline{\text{RxENB}}$ asserted and drives the first cell octet on RxDATA. The PHY layer runs out of data between clock edges 5–6 and asserts RxEMPTY to indicate invalid data on RxDATA. $\overline{\text{RxENB}}$ is deasserted and the ATM stops accepting data at clock edge 10. By asserting $\overline{\text{RxENB}}$ on clock edge 11, the ATM layer continues the transfer on clock edge 12. Deasserting $\overline{\text{RxENB}}$ causes RxDATA and RxSOC to become three-stated in the following cycle.

Between clock edges 1–3 and 10–12, RxEMPTY indicates the availability of an octet to transfer. Between clock edges 3–10 and after edge 12, RxEMPTY indicates whether data on RxDATA is valid.

Cell-Level Handshake

The only difference between Cell-level handshake and octet-level handshake is that when RxCLAV is asserted, the PHY layer transfers a whole cell (see Figure 16). RxCLAV and RxEMPTY have identical timing, i.e., RxCLAV asserted in a cycle following one with the final octet of a cell indicates the presence of a new cell to transfer.

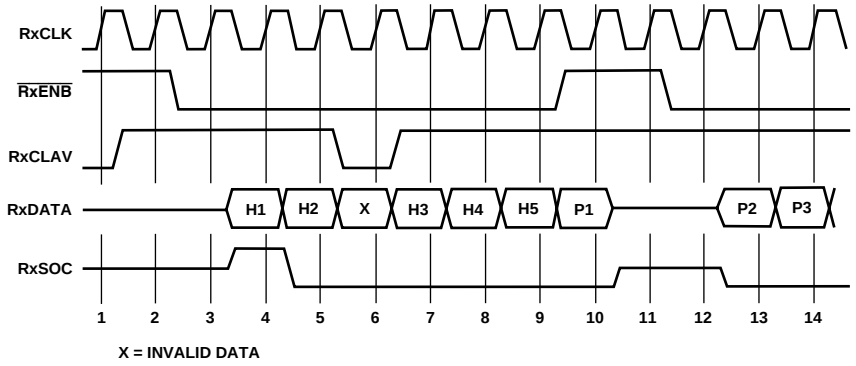


Figure 15. Receive Timing for Octet-Level Handshake

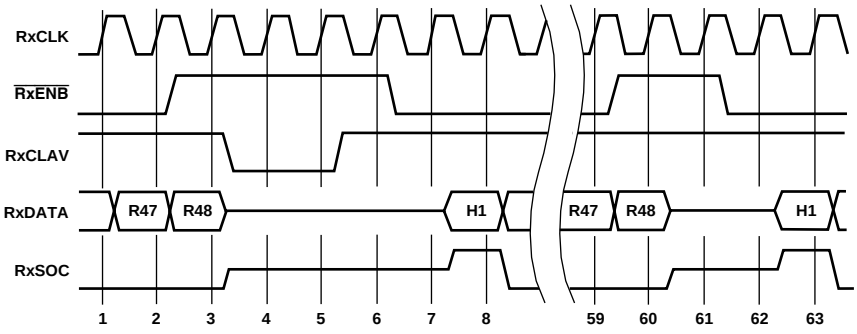


Figure 16. Receive Timing for Cell-Level Handshake

UTOPIA LEVEL 2 INTERFACE

UTOPIA Level 2 adds support for point-to-multipoint configurations to the capabilities of the Level 1 interface. Figure 18 details the standard interconnections between ATM and PHY layers.

Device Identification

PHY addresses for the AD6438's two PHY layer parts (one for fast data channel, one for interleaved data channel) are specified by CMV fields in the PHY address register. Note that an incorrect address can cause bus conflicts. Enabling the TRI_STATE_EN bit in the Rx_interface control register disables all outputs of the UTOPIA interface. For multi-PHY, each AD6438's RxCLAV signal is connected to a dedicated ATM layer chip input, so a three-state property is not needed.

Operation with One TxCLAV and One RxCLAV

The AD6438 supports the required Utopia Level 2 multi-PHY (MPHY) mode, known as operation with one TxCLAV and one RxCLAV.

Transmit Interface: MPHY Layer Cell-Level Handshake with One TxCLAV

UTOPIA Level 1 uses TxCLAV to convey transfer status from its single PHY layer device to the ATM layer. In UTOPIA Level 2, MPHY ports are individually selected for cell transfer, however TxCLAV status of another port can be polled while the selected device is transferring data.

The ATM layer *polls* the TxCLAV status of an MPHY port by placing its address on TxADDR when the ENB line is asserted. On the next cycle, the addressed MPHY device responds with its status on TxCLAV.

The ATM layer *selects* a specific MPHY port (target or source of the next cell transfer) by putting its address on TxADDR when the ENB line is deasserted and asserting the ENB line on the next cycle. An MPHY port is selected the cycle after its address appears on the TxADDR lines and $\overline{\text{TxENB}}$ is deasserted. Once a port is selected, the cell transfer process proceeds as per UTOPIA Level 1 cell-level handshake.

Note: MPHY devices can be operated in a single PHY environment by setting the address pins to the value required by the management interface.

The example in Figure 18 shows the case where PHYs are polled to the end of a cell transmission cycle.

As indicated by the TxCLAV signal, PHYs N–3 and N+3 can accept cells and PHY N+3 is selected (Note: PHY values are for example only).

The PHY is selected at rising clock edge #16. After cell transmission to PHY N+3 is started, the ATM layer resumes polling. Using two-clock polling cycles, up to 26 PHYs can be polled if all responses occur with minimum delay. In this case, this occurs when the response of the last PHY is obtained at clock edge #15 and the $\overline{\text{TxENB}}$ pulse to the PHYs follows immediately. If additional clock cycles are required to select the PHY, fewer PHYs can be polled during one cell cycle.

Back-to-back cell transmission can occur if the ATM selects PHY N again for the next cell transmission, leaves the $\overline{\text{TxENB}}$ line asserted, and starts transmitting the next cell with clock edge #15.

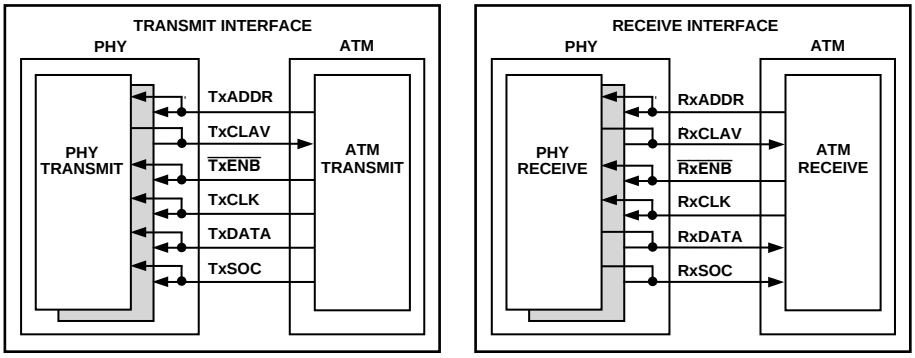


Figure 17. ATM to PHY Interconnection (UTOPIA Level 2)

Receive Interface: MPHY Cell-Level Handshake with One RxCLAV

UTOPIA Level 1 uses RxCLAV to convey transfer status from its single PHY layer device to the ATM layer. In UTOPIA Level 2, MPHY ports are individually selected for cell transfer, however RxCLAV status of another port can be polled while the selected port is transferring data.

The ATM layer *polls* the RxCLAV status of an MPHY port by placing its address on RxADDR when the ENB line is asserted. On the next cycle, the addressed MPHY device responds with its status on TxCLAV.

The ATM layer *selects* a specific MPHY port by putting its address on RxADDR when the ENB line is deasserted and asserting the Enb line on the next clock cycle. All MPHY devices only examine the value on RxADDR for selection purposes when $\overline{\text{RxENB}}$ is deasserted. An MPHY port is selected the cycle after its address appears on the RxADDR lines and $\overline{\text{RxENB}}$ is deasserted. Once a port is selected, the cell transfer process proceeds as per UTOPIA Level 1 cell-level handshake.

Note: MPHY devices can be operated in a single PHY environment by setting the address pins to the value required by the management interface.

The example in Figure 19 shows the corresponding case to Figure 18 for the receive interface. While receiving a cell from PHY N, the ATM layer polls other PHYs. PHY N–3 and PHY N+3 have cells available and PHY N+3 is selected. (Note: PHY values are for example only). As for the transmit interface, two-clock polling cycles allow polling of up to 26 PHYs in 8-bit mode and up to 13 PHYs in 16-bit mode.

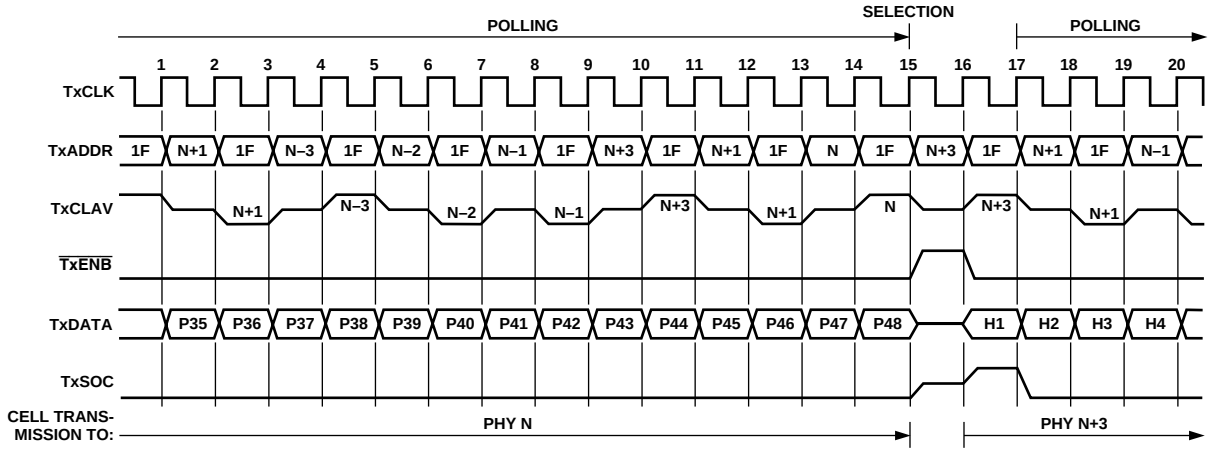


Figure 18. Polling and Selection Phases at Transmit Interface

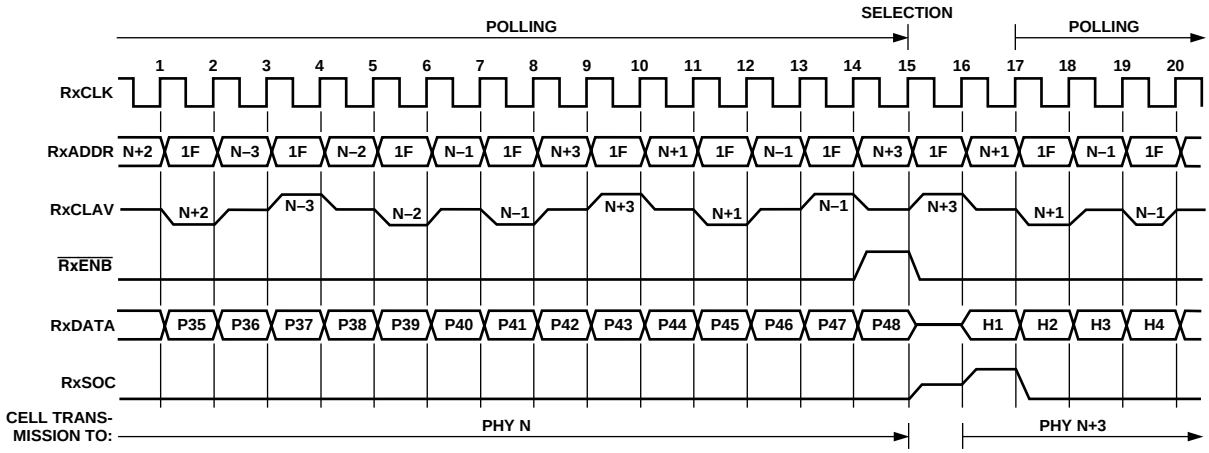


Figure 19. Polling and Selection Phases at Receive Interface

Timing Details at 33 MHz

Timing specification for the receiver side of a signal are the measurement base for the AD6438's ac characteristics. Setup and hold times are measured as shown in Figure 20. Three-state timing for the multi-PHY application (multiple PHY devices, multiplexed multiple output signals) is defined as shown in Figure 21. The timing references (tT5 to tT12) are covered in Table XI and Table XII (Note: A = P defines a signal from the ATM layer to the PHY layer, A = P defines a signal from the PHY layer to the ATM layer).

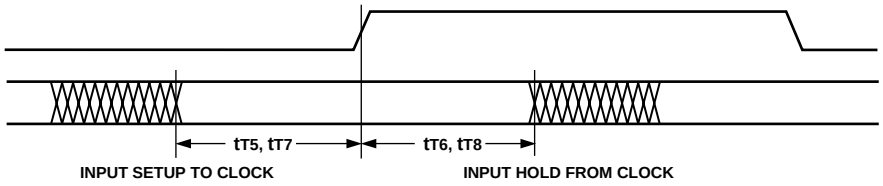


Figure 20. Setup and Hold Time Definition (Single and Multi-PHY)

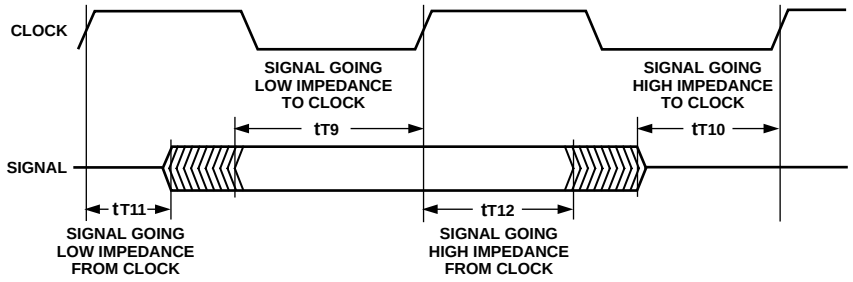


Figure 21. Three-State Timing (Multi-PHY, Multiple Devices Only)

Table XI. Transmit/Receive Timing on 8-Bit Data Bus (33 MHz at Cell Interface, Single-PHY)

Signal Name	DIR	Item	Description	Min	Max
TxCLK/RxCLK	A→P	f1	Frequency (Nominal)	0	33 MHz
		tT2	Duty Cycle	40%	60%
		tT3	Peak-to-Peak Jitter	—	5%
		tT4	Rise/Fall Time		3 ns
TxDATA [7:0], TxPRTY, TxSOC, $\overline{\text{TxENB}}$, $\overline{\text{RxENB}}$	A→P	tT5	Input Setup to TxCLK/RxCLK	8 ns	—
		tT6	Input Hold from TxCLK/RxCLK	1 ns	—
$\overline{\text{TxFULL}}$/TxCLAV, RxDATA [7:0], RxPRTY, RxSOC, RxEMPT/RxCLAV	A←P	tT7	Input Setup to TxCLK/RxCLK	8 ns	—
		tT8	Input Hold from TxCLK/RxCLK	1 ns	—

Table XII. Transmit/Receive Timing on 8-Bit Data Bus (33 MHz at Cell Interface, Multi-PHY)

Signal Name	DIR	Item	Description	Min	Max
TxCLK/RxCLK	A→P	f1	Frequency (Nominal)	0	33 MHz
		tT2	Duty Cycle	40%	60%
		tT3	Peak-to-Peak Jitter	—	5%
		tT4	Rise/Fall Time		3 ns
TxDATA [7:0], TxPRTY, TxSOC, $\overline{\text{TxENB}}$, TxADDR [4:0]	A→P	tT5	Input Setup to TxCLK/RxCLK	8 ns	—
$\overline{\text{RxENB}}$, RxADDR [4:0]		tT6	Input Hold from TxCLK/RxCLK	1 ns	—
$\overline{\text{TxFULL}}$/TxCLAV [3:0], RxDATA [7:0], RxPRTY, RxSOC, RxEMPTY/RxCLAV [3:0]	A←P	tT7	Input Setup to TxCLK/RxCLK	8 ns	—
		tT8	Input Hold from TxCLK/RxCLK	1 ns	—
		tT9	Signal Going Low Impedance to TxCLK/RxCLK	8 ns	—
		tT10	Signal Going High Impedance to TxCLK/RxCLK	0 ns	—
		tT11	Signal Going Low Impedance from TxCLK/RxCLK	1 ns	—
		tT12	Signal Going High Impedance from TxCLK/RxCLK	1 ns	—

SERIAL MODE

Serial Mode Description

Serial mode provides a synchronous clock and serial data that provides access to the ADSL PMD layer of the AD6438-1 IC. The Tx and Rx clocks are locked to the ADSL payload data rate and are output from the AD6438-1 IC. The phase locking of the DPLL (Digital Phase Lock Loop) provides the user access to framing modes 1, 2 and 3.

The serial mode bypasses all of the ATM TC sublayer incorporated into the AD6438-1. The serial interface on the AD6438-1 provides a bytes strobe that allows the user to align the ATM bytes with the ADSL PMD sublayer bytes per the ITU G.992.1 specification.

Advantage

The serial mode allows for the use of external asynchronous devices to connect to the ADSL PMD sublayer such as an external ATM TC sublayer. The serial mode also allows for the use of bit error rate testers.

AD6435/AD6438-1 Differences

The AD6435 provides an STM interface that is compliant with the original T1E1.4 standard. This part accepts a clock on the transmit side and performs byte stuffing and robbing to allow the modem to adapt to the incoming clock frequency. This is now known as framing mode 0 in the more recent ANSI T1E1.4 Issue 2 specifications and ITU 992.1 specifications. This part does not have the capability to do framing modes 1, 2 and 3.

The AD6438-1 provides a serial mode interface that differs from the STM mode on the AD6435 in that the AD6438-1 has a DPLL on the transmit side that provides a clock that is locked to the ADSL line rate. The AD6438-1 can receive all framing modes 0, 1, 2 and 3. The AD6438-1 transmits in framing modes 1, 2 and 3. Framing modes 0 and 1 have the same frame structure. The only difference between the two modes is that framing mode 1 never performs the synchronization action using byte stuffing and robbing.

The AD6438-1 serial mode was designed to provide a user access to the PMD sublayer and enable a modem to provide its own external ATM TC layer. It was not intended to be used in a modem compliant to the STM specification.

To be a compliant STM modem, it would be necessary to provide a dual latency downstream path as well as framing mode 0 on the transmit side.

BERT Testing

The serial mode is designed to be used for BERT testing. In addition to the typical setting up of a TTC Fireberd 6000 tester, it is sometimes necessary to invert the polarity on the transmit clock. This is due to the fact that the Fireberd outputs data and clock on the transmit side and the IC has its own output clock that is used to sample the data.

If it is desired to automate the testing process with no manual intervention, external hardware can be added to perform this clock polarity adjustment function or the automated software can be written to include the ability to invert the transmit clock polarity via the control interface.

Enabling Serial Mode

To enable serial mode it is necessary to set Bit 23 of OPTN 2 CMV to a 1.

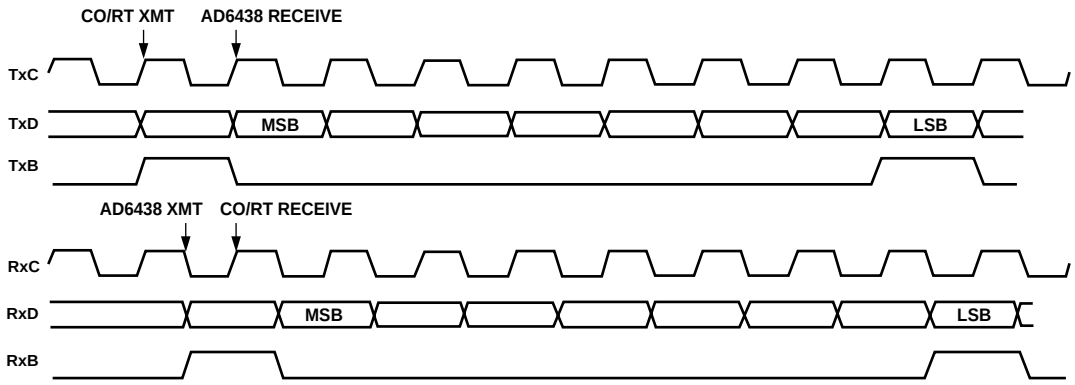


Figure 22. Serial Mode Timing

Interleave RAM Interface

The AD6438 interfaces an external 32k × 8 interleave RAM. The RAM interface consists of M_A (14:0), M_D (7:0), NM_WE, and NM_OE.

Since this RAM is used by both the Tx and Rx paths, there exists the possibility of memory contention if both want access to RAM at the same time. The interface logic is designed to access the RAM in two cycles, and as long as the RAM can support access at this speed, there will be no contention.

The external interleaver RAM is operated in WE controlled mode (chip select held low by hardware) and needs access time of <20 ns to operate without additional wait states.

Interface Timing

All signals transmitted by the AD6439 to the AD6438 are transmitted on the rising edge and sampled on the falling edge, except for the Tx_DREQ signal, which is transmitted by the AD6439 on the falling edge and sampled by the AD6438 on the rising edge. All output signals from the AD6438 to the AD6439 are transmitted by the AD6438 in the rising edge and received by the AD6439 on the rising edge (see Table XIV, Figure 23, Table XV, Figure 24).

Table XIV. Tx Serial I/F Timing

Parameter	Description	Typ	Units
t_{TFRM-S}	Setup Time of Tx_FRM from Falling Edge of Tx_Rx_SCLK	5	ns
t_{TFRM-H}	Hold Time of Tx_FRM from Falling Edge of Tx_Rx_SCLK	15	ns
$t_{TDREQ-S}$	Setup Time of Tx_DREQ from Rising Edge of Tx_Rx_SCLK	5	ns
$t_{TDREQ-H}$	Hold Time of Tx_DREQ from Rising Edge of Tx_Rx_SCLK	15	ns
t_{TBS-DV}	Data Valid of Tx_BS from Rising Edge of Tx_Rx_SCLK	10	ns
t_{TD-DV}	Data Valid of Tx_SDATA from Rising Edge of Tx_Rx_SCLK	5	ns

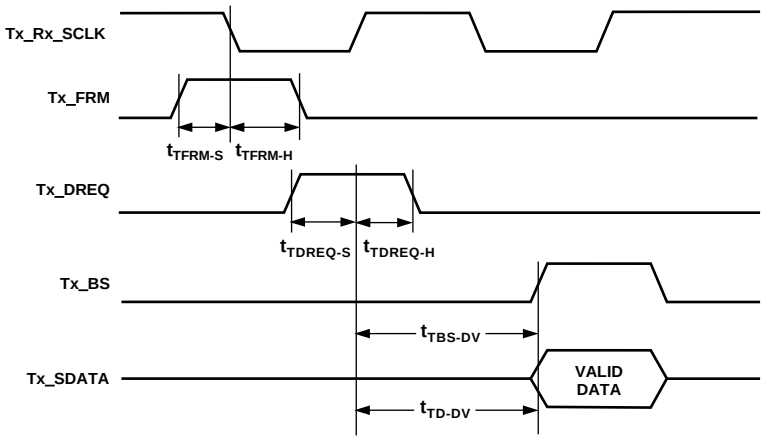


Figure 23. Tx Serial I/F Timing

Table XV. Rx Serial I/F Timing

Parameter	Description	Typ	Units
t_{RFRM-S}	Setup Time of Rx_FRM from Falling Edge of Tx_Rx_SCLK	5	ns
t_{RFRM-H}	Hold Time of Rx_FRM from Falling Edge of Tx_Rx_SCLK	15	ns
$t_{RDREQ-DV}$	Data Valid of Rx_DREQ from Rising Edge of Tx_Rx_SCLK	8	ns
t_{RBS-S}	Setup Time of Rx_BS from Rising Edge of Tx_Rx_SCLK	5	ns
t_{RBS-H}	Hold Time of Rx_BS from Rising Edge of Tx_Rx_SCLK	10	ns
t_{RD-S}	Setup Time of Rx_SDATA from Rising Edge of Tx_Rx_SCLK	5	ns
t_{RD-H}	Hold Time of Rx_SDATA from Rising Edge of Tx_Rx_SCLK	10	ns

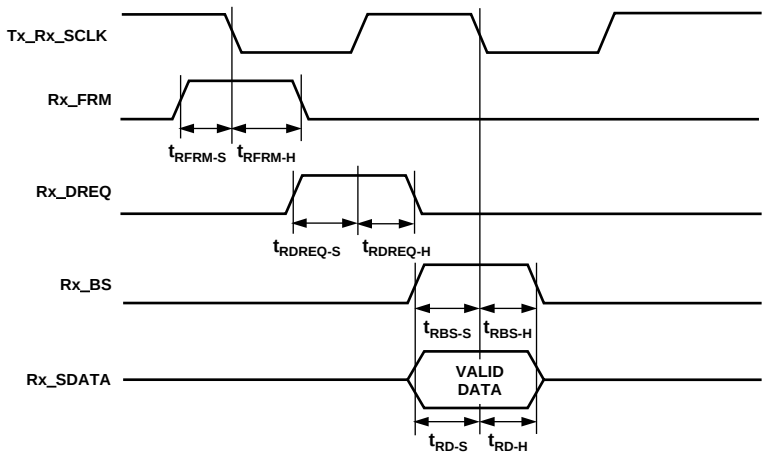


Figure 24. Rx Serial I/F Timing

Tx Serial Interface

The Tx serial interface between the AD6439 and the AD6438 uses five signals:

Tx_Rx_SCLK	Serial Clock Provided by AD6439
Tx_DREQ	Data Request Provided by AD6439
Tx_FRM	Frame Strobe Provided by AD6439
Tx_BS	Byte Strobe Provided by AD6438
Tx_SDATA	Serial Data Provided by AD6438

Rx Serial Interface

The Rx serial interface between the AD6439 and the AD6438 uses five signals:

Tx_Rx_SCLK	Serial Clock Provided by AD6439
Rx_FRMRx_FRM	Frame Strobe Provided by AD6439
Rx_BS	Byte Strobe Provided by AD6439
Rx_SDATA	Serial Data Provided by AD6439
Rx_DREQ	Data Request Provided by AD6438

DSP Port

The DSP port consists of a 14-bit address bus A[13:0], a 16-bit data bus D[15:0], DSP_CLK and three bus control pins, NRD, NWR, NCS.

Table XVI. Read Operation

Parameter		Min	Max	Unit
Timing Requirements:				
t_{RDD}	NRD Low to Data Valid		8	ns
t_{AA}	A0–A13, NCS to Data Valid		14	ns
t_{RDH}	Data Hold from NRD High	0		ns
Switching Characteristics:				
t_{RP}	NRD Pulsewidth	12		ns
t_{CRD}	DSP_CLK High to NRD Low	3	16	ns
t_{ASR}	A0–A13, NCS Setup Before NRD Low	2		ns
t_{RDA}	A0–A13, NCS Hold After NRD Deasserted	5		ns
t_{RWR}	NRD High to NRD or NWR Low	12		ns

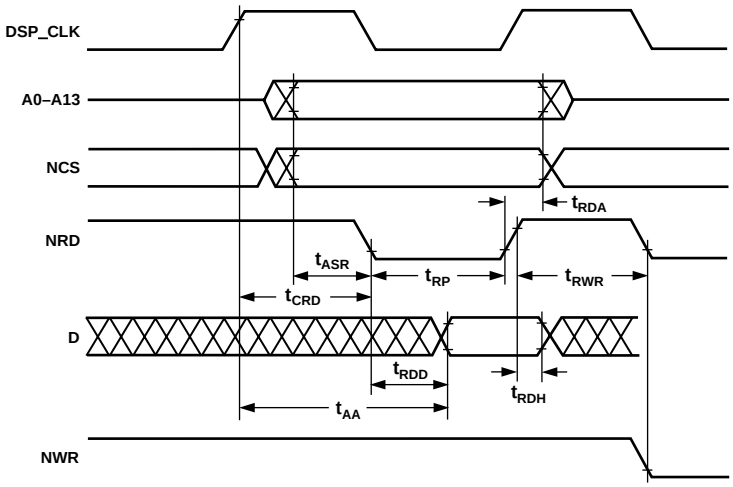


Figure 25. Read Operation

Table XVII. Write Operation

Parameter		Min	Max	Unit
Switching Characteristics:				
t_{DW}	Date Setup Before NWR High	10		ns
t_{DH}	Data Hold After NWR High	6		ns
t_{WP}	NWR Pulsewidth	12		ns
t_{WDE}	NWR Low to Data Enabled	0		
t_{ASW}	A0–A13, NCS Setup Before NWR Low	2		ns
t_{DDR}	Data Disable Before NWR or NRD Low	1		ns
t_{CWR}	DSP_CLK High to NWR Low	3	16	ns
t_{AW}	A0–A13, NCS Setup Before NWR Deasserted	17		
t_{WRA}	A0–A13, NCS Hold After NWR Deasserted	5		
t_{WWR}	NWR High to NRD or NWR Low	12		

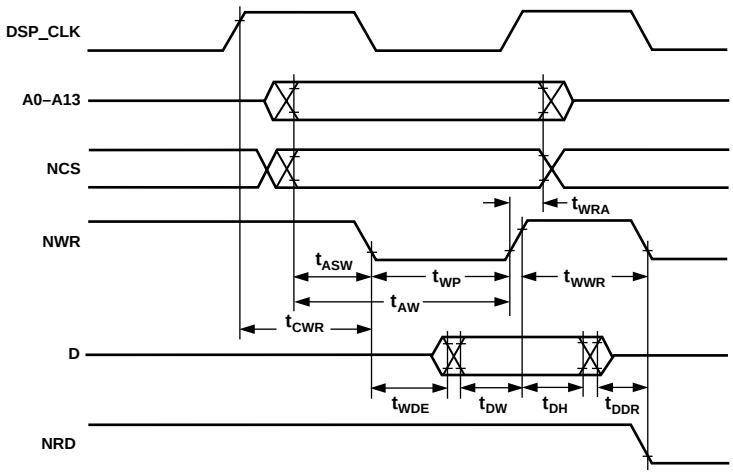
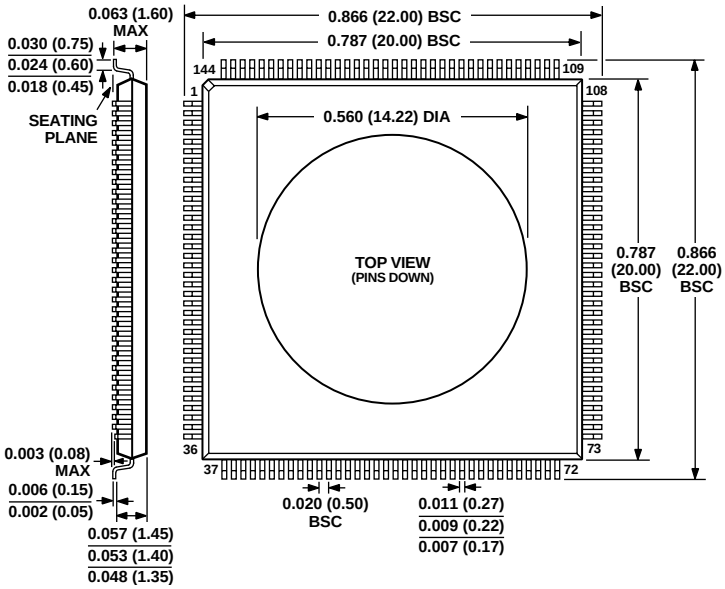


Figure 26. Write Operation

OUTLINE DIMENSIONS

Dimensions shown in inches and (mm).

144-Lead LQFP Package
(ST-144)



C3513–3–7/99

PRINTED IN U.S.A.