Project Specification

Project Name: DTMROC99

Version: V2.1.4

Author	**Date**	**Changes**
Ph. Farthouat	17 Jul. 97	First draft
Ph. Farthouat	21 Jul. 97	Electrical characteristics
Ph. Farthouat	23 Jul. 97	Paul Keener’s corrections
Ph. Farthouat	29 Jul. 97	Peter Lichard’s corrections
Ph. Farthouat	13 Aug. 97	Rick Vanberg’s corrections
Ph. Farthouat	19 Sept. 97	16-channel version
Ph. Farthouat	22 Oct. 97	Indiana workshop modifications
Ph. Farthouat	03 Nov. 97	Indiana workshop modifications
Ph. Farthouat	13 Nov. 97	Command decoder modifications, test pulse & list of blocks
F. Anghinolfi	27 Mar. 98	Second draft for review on 6/7 April 98
M Söderberg	16 Jun. 98	Updates from the design review in Penn
Ph. Farthouat	24 Jul. 98	Dead-time requirements updated
P.T. Keener	04 Aug. 98	Command decoder modifications
M Söderberg	16 Aug. 98	By-pass and test points documentation
Ph. Farthouat	26 Aug. 98	A few mistakes correction
N. Dressnandt	4 May 2000	Update Spec to reflect real DTMROC

1 Scope

The aim of the project is to design the digital read-out chip of the ATLAS TRT. The main constraints are:

- The inputs are connected to the outputs of the ASDBLR99.
- The timing and control interface is done according to the SCT protocol.
- This version is designed in DMILL.

This document covers the DTMROC99 used with the ASDBLR99. There are a few relevant documents which should be read:

1. ABC specifications. ftp://ftp.te.rl.ac.uk/atlas/specification.pdf

2. ASDBLR specifications.

3. Technical specification of the ATLAS TRT DTMROC.
http://www.quark.lu.se/activities/atlas/atlas.info/documentation/dtmroc4.5.ps

1.1 ATLAS TRT Readout Architecture

The ATLAS TRT front-end readout architecture is based on a bipolar amplifier/shaper/base line restorer (ASDBLR99) and a CMOS digital chip (DTMROC99).

To achieve the tracking and TR performances desired, the readout electronics has to perform the following functions:

1. Amplify the incoming straw signal.
2. Shape the amplified signal and remove the tail arising from the ion drift.
3. Apply two thresholds, one for tracking functionality (low threshold to detect minimum ionising particles) and one for transition-radiation functionality (high threshold to detect transition-radiation X rays).
4. Obtain timing information in 3ns bins for the low-threshold signal in order to yield the proper position resolution (and hence momentum resolution DeltaPt=??).
5. Store information about every bunch crossing in a pipeline as to whether or not the input signal exceeded the high-threshold values during the full level-1 trigger latency. The pipeline should contain at least 128 positions (3200 ns latency). Note that data is stored in the PIPELINE for 132 clock cycles (132 positions) however because the Command Decoder requires 5 clock cycles to decode a L1A trigger command, the effective delay is only 127 cycles (3175 ns latency).
6. For each level-1 trigger signal Accept (L1A), extract from the pipeline the information corresponding to the bunch crossing, which gave rise to the L1A and to the two following bunch crossings.

7. Gather in a Readout Driver (ROD) the data from many channels, compress them, format them and send them to the Readout Buffer.

Tasks 1 to 6 are performed by the *front-end electronics*, which is located on the detector, while task 7 is performed by the *back-end electronics*, which is located off the detector in USA15. An overview of the system is presented in Figure 1-1.

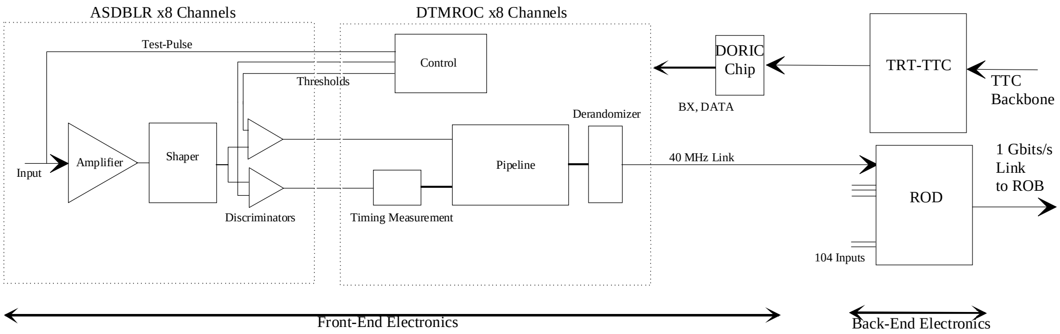


Figure 1-1 Schematic diagram of the TRT electronics

2 Technical aspects

2.1 Requirements

1. The chip will be designed to accept the 16 signals from the two ASDBLR99 chips and decode the ternary current levels to produce the high threshold and low threshold signals. The high threshold signal has to be glitch sensitive: as soon as a signal longer than 5ns is detected it should be extended until the next clock (see paragraph 2.2.3 of this document). Measurements show that the chip exceeds the required performance detecting pulses as short as 4ns.
2. The low threshold signal must be sampled for its presence or absence each 3.125 ns during the 25 ns clock cycle. An 8-bit word should be formed with these 8 samples.
3. At the start of each clock cycle (rising edge of the clock) the chip samples the high threshold signal and the 8 bit word described above and stores these values into a pipeline of 132 positions until a decision can be made whether to keep the data.
4. Upon receipt of a Level1 Accept signal (L1A) the data corresponding to that bunch crossing which gave rise to the L1A as well as the data of the two following bunch crossings are copied into another buffer called the derandomiser. There are always at least 4 empty bunch crossings between two consecutive L1A and thus there is no need for a pipeline position to be readout twice.
5. The data written into the derandomiser is transmitted to the ROD through a 40 Mbits/s copper link. The LVDS/Penn standard is used on this link and the chip follows the defined protocol to transmit the data (see paragraphs 2.2.8 - 9 of this document).
6. The chip provides two test-pulses to stimulate all odd or all even ASDBLR channels. The amplitude and delay of the test-pulses are programmable. See paragraphs 2.2.13 – 2.2.15 .
7. The chip provides two 8-bit threshold voltages to the tracking and transition-radiation comparators of each ASDBLR99 chip. See paragraph 2.2.12 .
8. The chip should provide, along with the data, the following status information in a header:
    a. A single mode bit indicating whether the chip is in test-mode or data-taking mode.
    b. A 3-bit L1A trigger-counter (L1ID) indicating the number of triggers since the last reset.
    c. A 4-bit bunch-crossing counter (BCID) indicating the number of bunch-crossings since the last reset.
    d. A 1-bit status indicating whether or not the DLL is locked (DLL-Lock).
9. The chip will incorporate features that will enable it to be tested at the wafer level and in situ. Tests include but are not restricted to:
    a. Transmission of programmable pattern through the pipeline and readout circuitry.
    b. Transmission of the chip ID.

c. The functionality of the LVDS/Penn receiver and driver blocks can be tested using a spy-point mode. These components may also be bypassed entirely to test the digital core independently. Special non-production bonding is required.

d. The Functionality of channel zero ternary receiver. Special non-production bonding is required.

10. It is a system requirement that the fraction of data that is lost due to the finite derandomiser storage on the chip is less than 1% with an average L1A rate of 75 kHz. The fraction of data lost when the L1A rate is 100 kHz must be less than 3%. This requirement is met with 12-event storage. The chip provides storage for 13 events. See paragraph 2.2.8

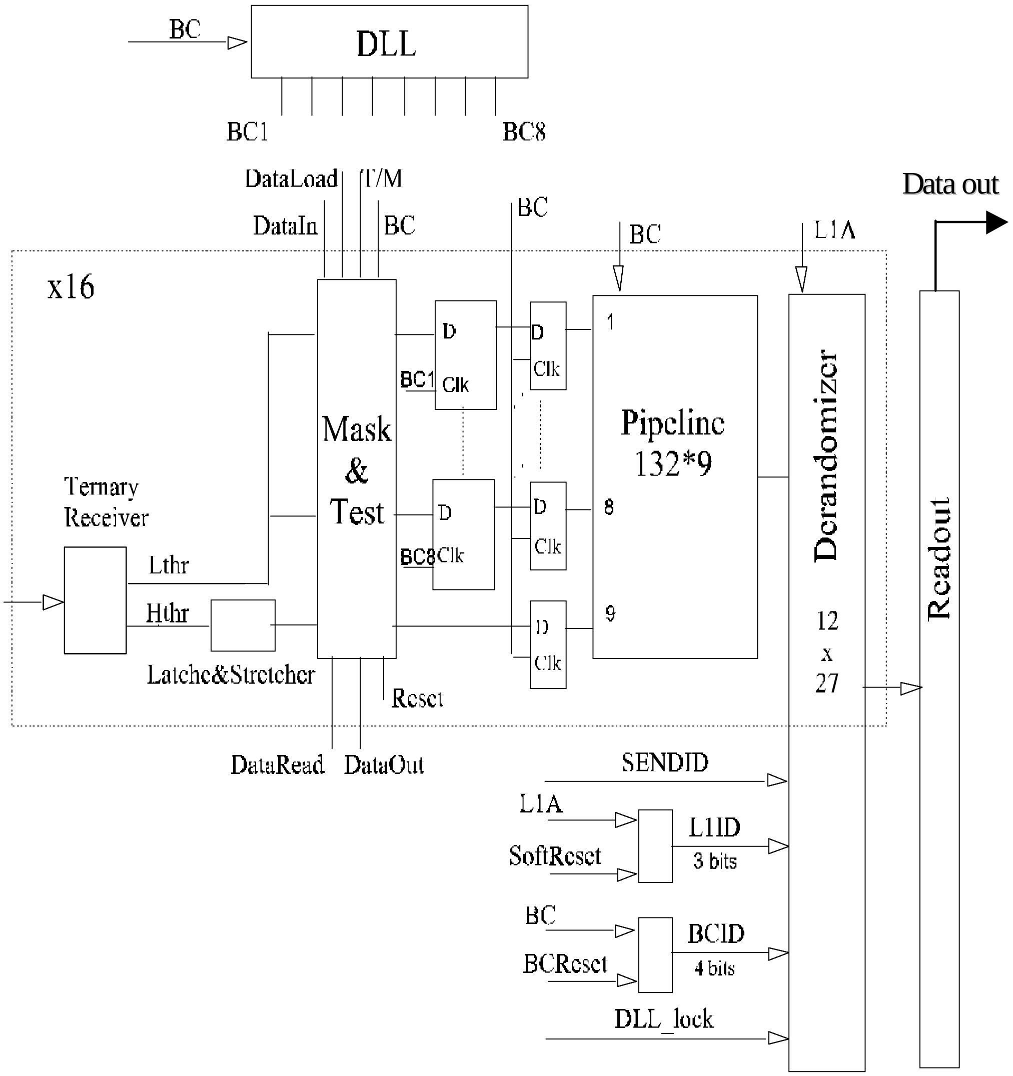


Figure 2-1 Block diagram of the main data flow.

2.2 Specification of deliverables

2.2.1 Input Level Translators: Ternary Receivers

The communication between the ASDBLR99 and the DTMROC99 is implemented with a ternary encoded differential current. The Table 2-1 gives the encoding function of the ternary receiver.

Table 2-1 Ternary current input

Amplified Signal in the ASDBLR	True Signal	Complementary Signal
No signal above thresholds	- 400 µA	0 µA
Signal between low and high thresholds	- 200 µA	- 200 µA
Signal above both thresholds	0 µA	- 400 µA

The DTMROC99 receives ternary encoded data form the ASDBLR 99 consisting of 0uA , 200uA, or 400uA of current leaving the TRUE or COMP inputs. The ternary receivers convert these currents back into two separate digital waveforms. The ASDBLR99 “A” ternary output should be connected to the DTMRO99 TRUE input. Under quiescent conditions (no pulse) there is 400uA leaving the TRUE port of the DTMROC99. A pulse will be viewed as a rising voltage on the TRUE input. One DTMROC99 services two ASDBLR99 chips.

The Ternary Receiver circuit is capable of receiving 4ns wide (at base) , tri-level differential current pulses. The circuit permits high-density communication between high gain front-end amplifiers (ASDBLR99 chips) and digital BICMOS DTMROC99 chips without driving up pin-counts and without causing self-oscillations via stray capacitive couplings back to amplifier inputs.

Using NPN common base transistors, an effective input impedance of about 100 Ohms can be achieved. A typical current mode logic level of 200uA will cause voltage swings of 20mV. Furthermore, the time constant due to this input impedance and stray capacitance is quite short (~0.5ns for 5 pF of stray capacitance). Quiescent power dissipation per channel depends upon the supply voltage. For typical VDD of 5V the power dissipated is 5.5mW per channel.

2.2.1.1 Testing and Bypassing Ternary Receiver Channel Zero.

Setting BYPASS_TERNREC = 5V will cause ternary receiver channel 0 to be bypassed. Input data is then taken from TRUE_HIGH_TERNREC and COMP_HIGH_TERNREC, which are 5v signals corresponding to a decoded ternary signal. Setting SPY_TERNREC = 5V will cause ternary receiver channel 0 outputs to be routed to SPY_HIGH_TERNREC and SPY_LOW_TERNREC. BYPASS_TERNREC and SPY_TERNREC have built-in pull-down resistors.

2.2.2 Input Latch

The asynchronous data input from the ASDBLR99 has a reliable duration, typically from 5 ns to a few clock cycles. These signals are not synchronous with respect to the clock and in order to detect short transitions on the signal a latching mechanism has to be introduced on the Hthr output signal from the ternary receiver according to the following specification:

- Any logical high level (active) on the Hthr signal longer than 5 ns shall be registered as a high level in the pipeline in the following time slice. (Next leading edge of the BC clock.)

- ▪ Measurements indicate latching on signals longer than 4ns. Still need measurement of set-up and hold times for latch function with respect to BC clock. What happens for hold times less than 1ns?

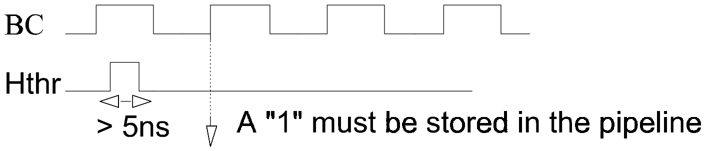


Figure 2-2 Input Latch

2.2.3 Time digitiser

The Lthr signal is time digitised: the level of the signal is stored each 3.125ns (25/8) leading to an 8 bit word showing the history of the signal during a clock cycle. This 8-bit word is loaded in the pipeline.

The following specifications are required:

1. The time between 2 consecutive sampling should be 25/8 ns +- 0.5 ns. Actual performance is addressed below. See 2.2.3.1 DLL
2. Special care should be taken to avoid any ambiguity during the loading in the pipeline (see Figure 2-3) as one could easily introduce a 25 ns error if the phase between the BC8 signal and the pipeline load is not properly adjusted. We should not have any events for which a 25 ns error is introduced.

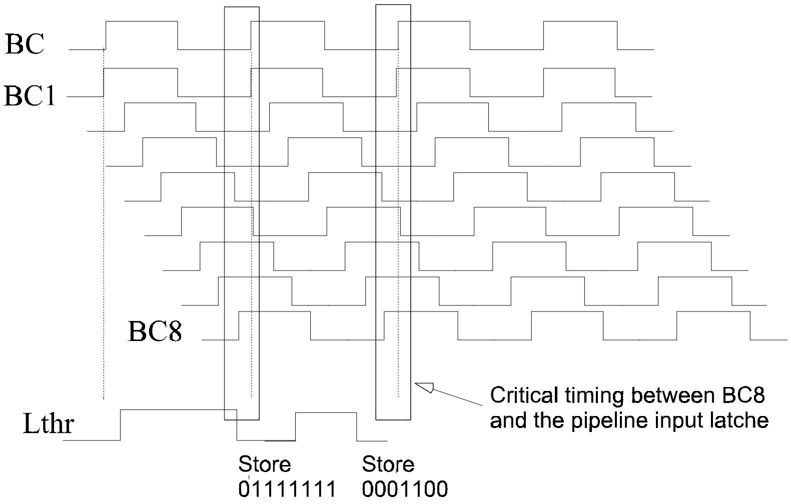


Figure 2-3 Time digitiser

3. A status bit indicating that the DLL is in lock must be provided. Lock is defined as ???
4. It is possible to reset (to put it in proper working conditions) the DLL with a software reset (DLL_reset) from the command decoder. An external hardware reset (hard_reset) signal will also reset the DLL. Need to check this.

2.2.3.1 Time digitiser performance

The Delay Lock Loop (DLL) effectively splits the system clock period into 8 finer intervals by generating 8 separate 40MHz clocks (BC1 – 8) which are each delayed by different multiples of 3.125ns. The FRONTEND block uses these to record the arrival of the tracking pulses.

Some DLL signals may be monitored when SPY_DIGITAL_ENABLE = 5V. (The pad has a pull-down) PD_DN is a copy of the DLL’s (discharge_b) correction signal. DT8 is a copy of DLL’s BC8 clock.. The DLL is known to have one odd-sized time bin. The bin has been mapped to the BC4-5 position at the FRONTEND block to avoid synchronization (latching) problems. The DLL will only operate correctly at 40MHz for this implementation. See performance measurements in appendix I (Vladimir/Penn).

2.2.4 Test and mask register

A 144-bit register is used either to mask the channels or to insert test data in the pipeline. This register is serially loaded and read-out via the command decoder. The T/M bit (see 2.2.11, Configuration Register) is used to select the mode of operation: if equal to 1 the register content is used as test input of the pipeline; if equal to 0 it is used as a mask. (Figure 2-4).
After reset (soft or hard?) T/M is set to 0 (Mask mode) and no locations are masked.

If in test-mode the ?? bit from the register is transferred into the pipeline. If in mask-mode a 1-bit will cause a zero to be put in that position in the pipeline. ???

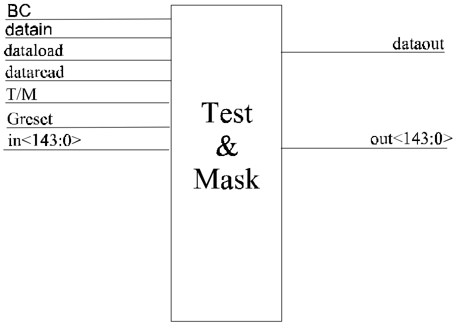


Figure 2-4 Test and mask register

2.2.5 Synchronisation and drive-strength issues

In order to avoid timing problem, an additional flip-flop clocked with BC at a well controlled phase with respect to BC1 and BC8 is introduced (see Fig 2-1). The architecture of the DTMROC99 might not guarantee proper latching into the PIPELINE over the full process/temperature/voltage/radiation variations. In particular, a race condition is observed in ELDO simulations for the FAST-FAST (FF) conditions (Fast process: 5.5V, 25C). Margin or hold time after latching is less then 2ns. It was decided to design to meet irradiated-slow-slow (ISS: 4.5V, 85C, irradiated) conditions, and to decrease the power supply voltage if necessary to meet FF conditions.

This problem can be ameliorated to some extent by increasing the drive strength out of the FRONTEND block in future designs. Currently for the ISS case the rise time is 4.4ns. This slow rise-time seriously degraded our ability to sacrifice ISS set-up time for improved FF margin.

2.2.5.1 Other Drive Strength Problems.

1. The drive strength out of the pipeline for D<1> and D<2> is marginal for ISS case. (But still latches) There should be buffers on these outputs since they drive test-pads across the chip.

2. Channel zero Ternary receiver has additional delay as compared to other channels because of spy-enable and bypass functions.

3. Must double tpanalog drivers: tptr_o tptr_e. The rise time is too slow.

2.2.6 Pipeline

The pipeline consists of a 132 x 16 x 9 bits memory. On each clock rising edge 9 data bits per channel are stored (High threshold value + 8 bit time sample) [144 bits in total] at address 0 and the 144 bits of the address 132 are put on the output bus for possible readout.

The PIPELINE stores data for 132 clock cycles (25ns per cycle). The COMMAND DECODER requires 5 clock cycles to decode a L1A trigger. The effective storage time is therefore 3175ns. ((132-5)*25ns) Each data set contains 144 bits. (16 channels x 9 bits per channel)

If the chip is in test mode (T/M =1) the input of the pipeline is provided by the Test and Mask register (see 2.2.4). There is also a spy-point mode in which the first two bits of the 144 inputs into the PIPELINE may be read in from external pads and readout on external pads. This mode is activated when SPY_DIGITAL_ENABLE = 5V. The following then become active: SPY_LATCH1_OUT, SPY_LATCH2_OUT, and SPY_PIPELINE1OUT, SPY_PIPELINE2OUT.

2.2.7 L1A and BC identifiers: L1ID and BCID

Two counters keep track of the number of L1A triggers and the number of Bunch Crossing clock cycles since the last reset: L1ID and BCID.

L1ID is implemented using a 3 bit counter incremented by the L1A signal decoded by the timing and control interface and reset by the Soft Reset signal or the external Hard_reset signal. It is initialised to “111” so that the first BC is numbered zero.

BCID is implemented using a 4-bit counter incremented by the BC clock rising edge and reset by the BCR signal or the Soft Reset signal from the Command Decoder or the Hard_reset signal. It is initialised to zero so the first BC is numbered one.

The outputs of these registers are stored into the derandomiser (not the pipeline in order to save space and to accommodate the fact that the SCT TTC protocol does not allow simultaneous command and trigger transmission) to provide the header of the data readout stream (see 2.2.9)

2.2.8 Derandomiser

The derandomiser is an additional buffer acting as a FIFO. Upon receipt of a L1A (as decoded by the control interface) the current pipeline output and the following two consecutive ones are stored in the derandomiser for readout. In addition to these data, the SENDID status bit, the L1ID, the BCID (at the time the L1A occurs), and the DLL status bit are stored. This gives 444 bits[1] to be stored per event. The derandomiser can store 13 events[2].

2.2.9 Readout

As soon as the derandomiser is not empty the readout is placed serially on the DATA_OUT line at a 40MHz rate. The LVDS/Penn standard is used. As the event size is constant a simple protocol is used: a 3 bit preamble “101” is sent at the beginning of an event. When idle, the data output signal is “0”.

Data format
The data format should be:

PREAMBLE [3 bits] “101”
SENDID [1-bit], L1ID [3-bits], BCID [4-bits], PLL-lock [1-bit]
STRAW1 BC1 [9-bits]
STRAW1 BC2 [9-bits]
STRAW1 BC3 [9-bits]
...
...
STRAW16 BC1 [9-bits]
STRAW16 BC2 [9-bits]
STRAW16 BC3 [9-bits]

If the chip is placed in the Send_ID mode (see 2.2.11, Configuration Register), all the STRAW BC [9 bits] words are replaced by a 9 bits word made of the 6 bit chip address (CHIP_ID) and three zeroes. This word is repeated 48 times to comply with the normal data stream length.

The Send_ID mode format should be:

PREAMBLE [3 bits] “101”
SENDID, L1ID, BCID, PLL-lock [9 bits]
CHIP_ID 000 (1)
CHIP_ID 000
...
...
CHIP_ID 000
CHIP_ID 000
CHIP_ID 000 (48)

[1]. 16 channels time 3 time slices time 9 bits, plus 9 bits for status bits

[2]. The derandomizer shall be able to contain a minimum of 9 events to cope with the 1% data lost at L1A rate of 75 kHz and 12 events to cope with the 3% data lost at L1A rate of 100 kHz. The derandomizer itself contains 12 events; an additional register in front of it contains the 13rd one. See LUNFD6 / NFFL-7118 1995 , Technical Specification of the ATLAS TRT Drift Time Measurement Read-OUT chip.

Readout modes

Depending on the F/R bit (see 2.2.11, Configuration Register) the full data set is transmitted (F/R=1) or the last 4 bits of the third time slice time digitiser is skipped (F/R=0). The same function applies if the chip is in Send_ID mode : the last 4 bits of every third CHIP_ID word is skipped if F/R = 0.

If placed in Time-adjustment mode (see 2.2.11, Configuration Register) a succession of 101010... is sent. It does not depend on F/R or SENDID bits.

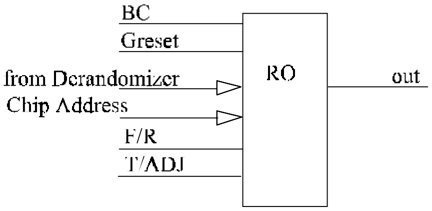


Figure 2-5 Read-out block

2.2.10 Command Decoder

The command decoder block receives the BC and a DATA signal from the ROD. Both signals are LVDS/Penn standard. It decodes the DATA stream and issues all the necessary timing signals (L1A,...), internal registers read/write strobes and data.

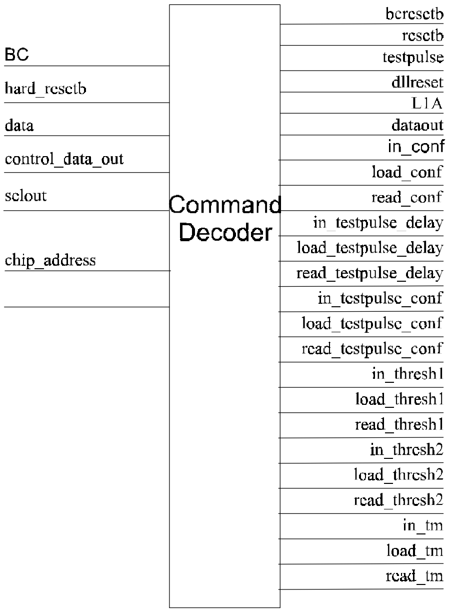


Figure 2-6 Command Decoder

Table 2-2 Command Decoder Input Signals

Name	Function
Bc	Clock
hard_reset_b	External reset
Data	Incoming data stream
chip_address<5..0>	6 bit chip address
In_conf	Contents of conf register on read
In_testpulse_delay	Contents of testpulse_delay
in_thresh1	Contents of threshold1 register
in_thresh2	Contents of threshold2 register
in_testpulse_conf	Contents of testpulse _conf register on read
in_tm	Contents of tm register on read
spy_enable	Enable test outputs
test_load_conf	External control of load_conf signal, xor'ed with internal control
test_load_testpulse_conf	External control of load_testpulse_conf signal, xor'ed with internal control
test_load_thresh1	External control of load_thresh1 signal, xor'ed with internal control
test_load_thresh2	External control of load_thresh2 signal, xor'ed with internal control
test_load_testpulse_delay	External control of load_testpulse_delay signal, xor'ed with internal control
test_load_tm	External control of load_tm signal, xor'ed with internal control
test_l1a	External control of l1a signal, xor'ed with internal control

Table 2-3 Command Decoder Output Signals

Name	Function
Bcresetb	Bunch counter reset, active low,
Resetb	General reset, active low, 4 clocks
Testpulse	Testpulse signal, 1 clock
Dllreset	DLL reset, 1clock
l1a	Level 1 trigger accept, 1 clock
data_out	Data stream for writing registers
load_conf	Load conf register control
read_conf	Read conf register control
load_testpulse_delay	Load testpulse_delay register control
read_testpulse_delay	Read testpulse_delay register control
load_testpulse_conf	Load testpulse_conf register control
read_testpulse_conf	Read testpulse_conf register control
load_thresh1	Load thresh1 register control
read_thresh1	Read thresh1 register control
load_thresh2	Load thresh2 register control
read_thresh2	Read thresh2 register control
load_tm	Load tm register control
read_tm	Read tm register control
control_data_out	Output of register contents on read
Selout	Select signal to tri-state drive when control_data_out is active
Tcunusedcode	Field 1 unknown test output
Fcunusedcode	Field 2 unknown test output
Flunusedcode	Field 3 unknown test output
Scunusedcode	Field 4 unknown test output
Endcommand	Internal test signal output, marks reset of decoding

Info on write?

Upon a read request (what is delay?) the command decoder serially transmits the contents of the selected register on the differential cmd_out line. This line uses the LVDS/Penn standard and is common to several chips. Therefore it has a "tri-state" capability. A 3-bit preamble is introduced ["101"] and the idle state of this line is "HiZ".

The Table 2-4 gives the commands that must be decoded. "aaaaaa" is the chip address. If "aaaaaa" equals "111111" then all the chips are addressed (broadcast). The chip address is defined by setting external pins.

All the bits are received on the DATA input starting with the most significant one except for the testpulse delay.

Error handling

The protocol is based on the ABCD chip specification, which lacks any advanced protection against transmission errors. The bit patterns are however chosen such that a single bit error should not cause an acceptance of a wrong command.
The chip must take the following actions if it receives commands that it does not recognise:

> **Unrecognised field 1**
> The command decoder must flush the unrecognised field 1 and start looking at the following bit for a new command.

Unrecognised field 2
If field1 = 101 but field2 does not match a valid pattern, the command decoder must flush all 7 bits from the unrecognised field 1 and 2 and start looking at the next bit for a new command.

Field 3 < 12
If field1 = 101, field2 = 0111 but field3 is some number less than 12, the command decoder must flush 15 bits from the unrecognised fields 1,2 and 3 plus however many bits field3 designates and starts looking at the next bit for a new command.

Mismatched chip address or unrecognised field 5
If the chip address (field4) does not match the chip address established on external pins or the field5 does not match a valid pattern, the remaining bits in the input stream indicated by field3 are flushed and the command decoder starts looking at the next bit for a new command.

Table 2-4 Control commands and access. Description of fields.

Field 1	**Field 2**	**Field 3**	**Field 4**	**Field 5**	**Field 6**	
110						L1A (on the BC following the last bit)
101	0100					Soft Reset (on the BC following the last bit)
101	0010					BC Reset (on the BC following the last bit)
101	0111	00010010	Aaaaaa	000000	6 data bits	Write Configuration Register
101	0111	00011100	Aaaaaa	001010	16 data bits	Thresholds Register ASDBLR1
101	0111	00011100	Aaaaaa	001100	16 data bits	Thresholds Register ASDBLR2
101	0111	00011100	Aaaaaa	010010	16 data bits	Test Pulse Configuration
101	0111	00001100	Aaaaaa	010100		Dll Reset
101	0111	00010010	Aaaaaa	000110	6 data bits	Test pulse delay
101	0111	00001100	Aaaaaa	011000		Test Pulse trigger (on the BC following the last bit)
101	0111	10011100	Aaaaaa	011110	144 data bits	Test/Mask register
101	0111	00001100	Aaaaaa	1rrrrr		Read Registers. “rrrrr” being the register address as defined above

2.2.11 Configuration Register

The Configuration register contains the 6 following control bits Bit<5> gets sent first. Its initial value at start-up is “xxxxx”:

Table 2-5 Configuration register.

Bit<5>					Bit<0>
SPARE2	SPARE1	T/M	T/ADJ	SENDID	F/R

- The two first bits are spare bits (not used in DTMROC) accessed through internal test pads.
- T/M : Test mode bit. If set, the chip is in test mode and the pipeline is filled with the test pattern register content. If clear, the chip is in data taking mode.
- T/ADJ : Time adjustment mode. If it is set, the chip transmits “010101...” on the readout data line to the ROD.
- SENDID : Send-ID bit. If it is set, the chip ID (address) is sent upon a L1A reception (see 2.2.9).

- F/R : Full/Reduced read-out bit. If it is set, the 3 BC data sets are transmitted. If not, only the first half of the last BC data is sent.

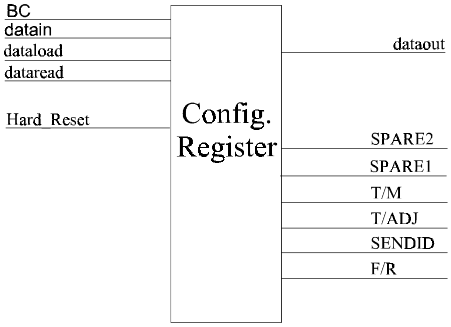


Figure 2-7 Configuration Register

2.2.12 Thresholds Registers 1 and 2

There are two 16 bits threshold registers. One controls the threshold setting for ASDBLR1, and the other one controls the threshold setting for ASDBLR2 (see 2.2.10). Each register sets two 8-bit voltage DAC values: One for the low and one for the high thresholds for one ASDBLR.

The data format for each of the two registers is (Bit <15> is first in serial order):

Table 2-6 Thresholds register.

Bit <15:8>	**Bit <7:0>**
8 bits High Threshold; Bit<15> MSB ; Bit<8> LSB	8 bits Low Threshold; Bit<7> MSB ; Bit<0> LSB

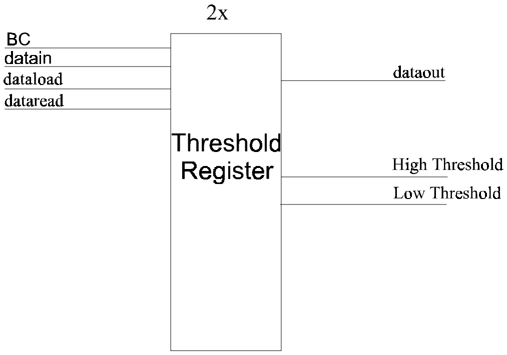


Figure 2-8 Threshold Register

2.2.13 Test pulse Configuration

One 16 bits register controls (see 2.2.10) the Testpulse amplitude using two 6-bit internal DAC's: one for odd and one for even channels. The two remaining bits are used for enabling/disabling the odd/even test pulse generators.

The data format is (Bit <15> is first in serial order):

Table 2-7 Test pulse configuration.

Bit <15>	Bit <14>	Bit <13:8>	Bit <7>	Bit <6>	Bit <5:0>
Not used	Odd Enable Bit 1 = Enable 0= Disable	6 bits Odd channels test pulse amplitude; Bit<13> MSB ; Bit<8> LSB	Not used	Even Enable Bit 1 = Enable 0= Disable	6 bits Even channels test pulse amplitude; Bit<5> MSB ; Bit<0> LSB

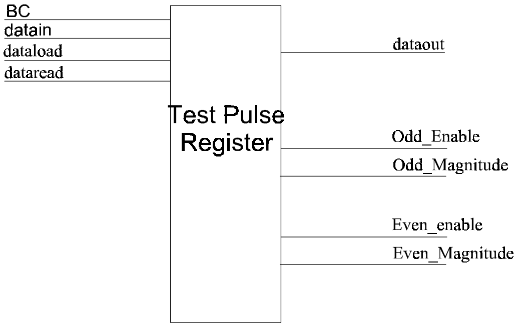


Figure 2-9 Test pulse register

2.2.14 Test pulse Delay

The test pulse signal generated by the command decoder (see 2.2.10) can be delayed with a minimum delay spread of 25ns in 32 steps. The delay is controlled by 5 bits out of the 6-bit Testpulse delay register. The delay set by this register is not absolutely calibrated, but it guarantees that the test pulse delay spreads over more than one clock cycle for all conditions of operations and process. The typical time step for delay adjustment is 1.56ns.

The data format is (Bit<5> is last? in serial order):

Table 2-8 Test pulse delay

Bit <5 >	Bit <4 :0>
Unused	Bit<4> = MSB Bit<0> = LSB

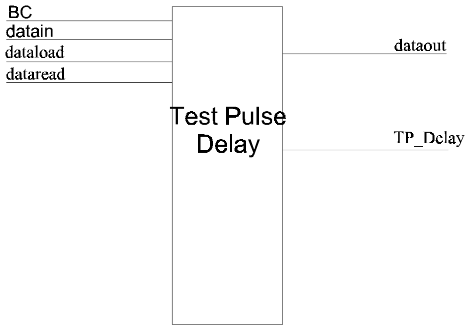


Figure 2-10 Test pulse delay Register

2.2.15 Test pulse Shape

The two test-pulses (odd and even) are generated from a voltage step whose amplitude is fixed by the Test pulse Configuration register (see 2.2 13) and timing is fixed by the Test pulse delay register (see 2.2.14).

The test-pulses' DAC's provide the voltage step amplitude for odd and/or even channels and the Test pulse delay circuit provides the time with a typical 1.56ns step resolution at which the pulse(s) is (are) applied.

When connected to the ASDBLR99 inputs, TST_E and TST_O, the DTMROC99 outputs, tp_odd and tp_even will produce currents at the odd and even channel inputs to the ASDBLR99 preamp that closely mimics the current pulse produced by the TRT straw filled with an Xe/Ar/CO_2 gas. The anticipated stray capacitance of the interconnect between the DTMROC99 and ASDBLR99 is 7pF.

2.2.16 Miscellaneous

Power-up condition
After power-up the chip should be in a known state and work without external intervention. The chip should be in the same state as after a Hard_reset.

Hard reset
A Hard_reset input is provided. It must be LVDS/Penn compliant. An active signal on the hard reset lasting for 4 rising edges of the clock, is accepted as a valid hard reset, otherwise it is rejected. After a Hard_reset the DLL should be locked, the masks should be off, the chip should be in normal data taking mode and the pipeline and derandomiser pointers are reset. The DAC threshold values will be reset.

Soft reset
A Soft reset command is provided through the serial DATA input of the chip and the Command Decoder (see 2.2.10). The "resetb" signal provided by the command decoder is the logical sum of Hard_reset and Soft reset. Soft reset does not change DAC register values, or does it reset the readout controllers state machine or the configuration register.

DLL reset

A DLL reset command is provided through the serial DATA input of the chip and the Command Decoder to reset only the DLL.

Bunch Crossing Reset (BCR)

A Bunch Crossing Reset command is provided through the serial DATA input of the chip and the Command Decoder (see 2.2.10). The “bcresetb” signal provided by the command decoder is the logical sum of Hard_reset, Soft reset, and BCR (see 2.2.7).

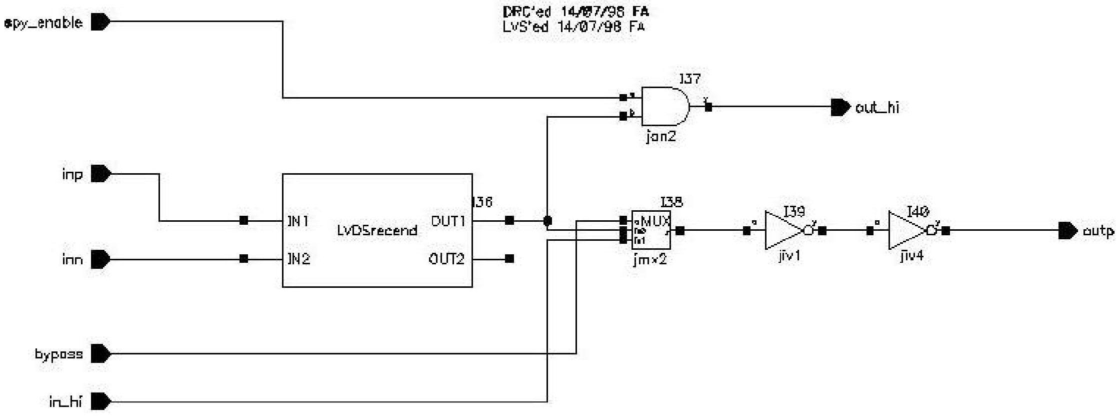
Test points

Critical I/O or nets should have test pads available to allow sufficient testing and characterising capability in the event of questions on cell functionality. In some cases these pads will only be passive spy points available for test probe or test bonding, but in other cases it may be necessary to plan for an external overdrive signal.

In order to facilitate testing there are several blocks that have been expanded with additional logic. These extra signals of course require pads to access them and these have, with few exceptions, been placed in the pad ring. All test inputs have pads with a pull-down[1] in order for them to be inoperative without necessitating any particular attention. The plan is to use a large package for prototype evaluation and bond out all signals, but later on in production only bond out the essential pads.
All test outputs are gated and controlled by appropriate enable signals. These enable signal are, as explained above, connected to pull-down pads and therefore normally off.

LVDS/Penn receiver.

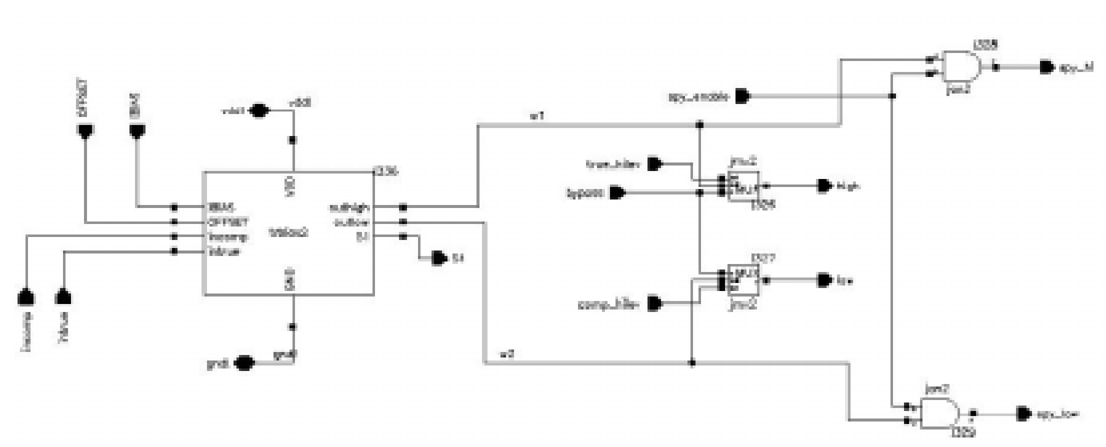
It is possible to completely bypass the driver and use a CMOS level input instead. It is also possible to monitor the CMOS output of the receiver to help in debugging the receiver itself. This modified receiver is used for all LVDS/Penn inputs, i.e. bc, hard_reset and cmd_in. Bypass and spy_enable on the receivers are controlled by the external signals bypass_lvds_rec and spy_lvds_rec.



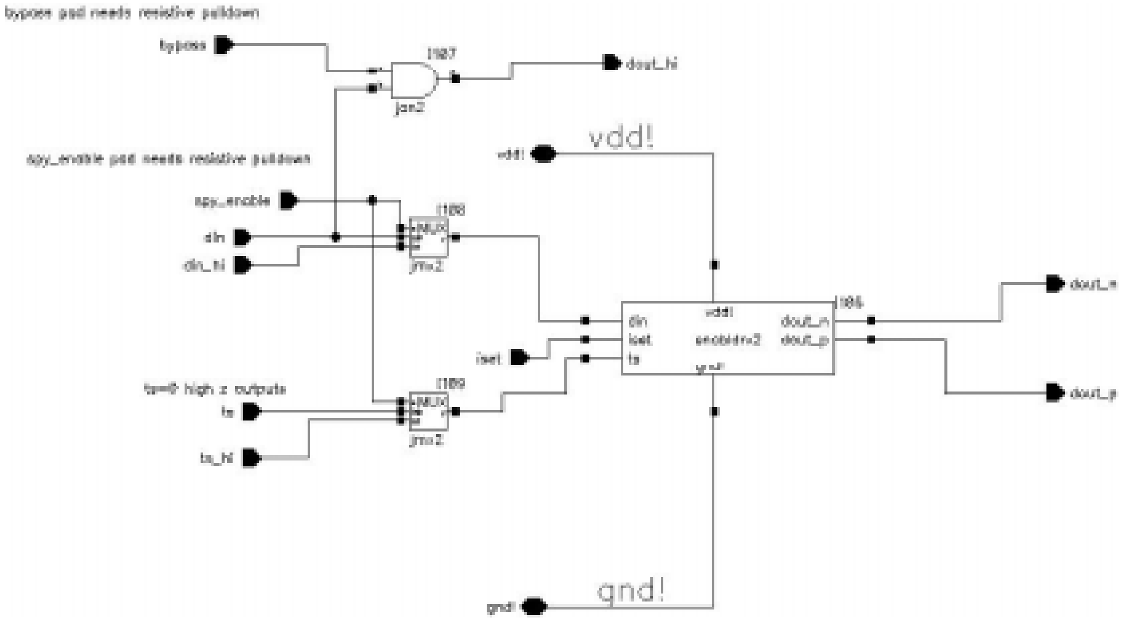
Ternary receiver.

[1]. Except for hard_reset_hi, which is connected to a pull-up pad since it is active low.

The ternary receiver may also be bypassed, but because of its particular interface there is no 1 to 1 correspondence between high-level inputs and internal signals as can be seen below. Instead there are CMOS inputs for low and high threshold. There are also spy outputs to monitor the function of the ternary receiver itself. Because of the large number of extra pads needed this modified receiver is only used on channel 0. Bypass and spy_enable is controlled by the external signals bypass_ternrec and spy_ternrec respectively.



LVDS/Penn driver.
The driver can be bypassed to send its input data to a CMOS output. As with the other drivers there are also extra pins added to be able to test the block itself in this case din_hi and ts_hi. This driver is used on the data_out signal but not on the cmd_out in order to save pins.



Digital spy points.
There are a number of nodes on the chip that it is possible to spy on. The monitoring of all these nodes is controlled by the spy_digital_enable signal.
The DLL has 2 signals connected to the output, dt8 and pd_dn. Dt8 is the output of the eighth DLL element, to give a rough estimate of the delay and pd_dn is the discharge signal that discharges the capacitor that holds the control voltage for the delay elements.
There are 2 signals to spy on after the latch: spy_latch1_out, spy_latch2_out. These are slice 1 and 2 of channel 0 respectively.
There are 2 more signals between the pipeline and the derandomiser, spy_pipeline1_out and spy_pipeline2_out, that represent the same bits as above, but after the pipeline.

Command decoder.
It has 5 test inputs to be able to write to all registers + issue an L1A trigger, even if the command decoder would fail completely. Read-back of registers will not be possible though. The inputs are: test_load_conf (config_register), test_load_testpulse_conf (testpulse magnitude and enable), test_load_thresh1 (threshold1 magnitude), test_load_thresh2 (threshold2 magnitude), test_load_testpulse_delay (testpulse delay length), test_load_tm (mask register), test_l1a (assert L1A trigger). These inputs must be held high for the duration of the data stream on cmd_in_hi. Test_l1a should be high for one clock cycle to issue a trigger during test-mode. To facilitate examination of the command decoder itself there are also 5 test outputs available: Tcunusedcode, fcunusedcode, flunusedcode, scunusedcode and endcommand. These are also enabled by spy_digital_enable. The pad-frame is completely filled in the vicinity of the command decoder and rather than routing test outputs across the entire die, the test outputs were connected to probe pads inside the pad-frame.

Testpulse.
There are 2 digital test outputs on this block, tp_even_test and tp_odd_test. They are enabled by the tp_spy_enable. These signals represent the outputs from the delay chain that controls when the Testpulse is asserted, but before shaping. There are also 2 analog probe pads in the test pulse block where a mirrored version of the Testpulse DAC's outputs can be tested.

DACs.
There is no direct way of looking at the DACs, but there is a way of feeding them their reference current, if the internal references do not work. The backup reference input for the threshold DACs is named th_bias and is enabled by th_bias_control. For the Testpulse DACs it is vrp that is enabled by tp_bias_control.

2.2.17 List of Blocks.

Table 2-9 gives the list of blocks and the responsibility for them.

Table 2-9 List of blocks

Block name	**Responsibility**
Ternary Receiver	Penn
DLL	CERN/Michigan
Input Latch (Frontend)	Cracow
T/M Register	Cracow
Pipeline	Cracow
Derandomizer	Cracow
L1ID, BCID	Geneva
Command Decoder	Penn
Configuration Register	Geneva
Thresholds DACs	CERN
Test-pulse DACs	CERN
Read-out	Geneva
Test-pulse generation	Penn
Test-pulse delay	Penn
Test-pulse configuration	CERN
LVDS drivers	Penn
LVDS receivers	CERN
Hard-reset and power-up circuitry	Lund/CERN

2.2.18 Input/Output Connections

The Tables 2-10 and 2-11 describe the names and functions of the various input/output connections to the chip. The table is not exhaustive, but lists the primary signals. This is relevant for the DTMROC only not for ASTRAL.

Table 2-10 Input Signals

Name	Function	Type
tern<00-15>_<comp,true>	Signal Inputs from ASDBLRs	Ternary current
diff_bias	Bias of the differential drivers	Analog current
Vrp	Bias of the Test Pulse DACs	Analog current
bc_<pos,neg>	Input clock of the chip	LVDS
cmd_in_<pos,neg>	Input DATA of the chip	LVDS
hard_reset_<pos,neg>	Hard reset input	LVDS
address<5..0>	6 bit chip address	CMOS

Table 2-11 Output Signals

Name	Function	Type
data_out_<pos,neg>	Data out	LVDS
cmd_out_<pos,neg>	Out data for control	LVDS
tp_odd	test-pulse for ASDBLR (odd inputs)	Analog
tp_even	test-pulse for ASDBLR (even inputs)	Analog
Threshold1_high	High threshold for ASDBLR1	Analog
Threshold1_low	Low threshold for ASDBLR1	Analog
Threshold2_high	High threshold for ASDBLR2	Analog
Threshold2_low	Low threshold for ASDBLR2	Analog

There are 49 input pins and 10 output pins[1].

[1]. Power pads and test pads excluded

2.2.19 Electrical Specifications

2.2.19.1 Supply Voltage

The supply voltage should be 5 V +/- 5% for the DTMROC.

The minimum ramping up speed necessary for the chip to be initialised at power up must be specified.

2.2.19.2 Power Consumption

The power consumption shall not exceed 60 mW per channel or 960 mW for the total chip under ISS conditions. Nominal power consumption was measured at 37.5mW per channel. After irradiation to 10^{14} n/cm^2 the power consumption was 47mW per channel. (~125mA → ~150mA at 5v)

2.2.19.3 Input/Output levels

Table 2-12 Ternary Receivers.

Table 2-12 Input levels for hits from the ASDBLR (positive input)

Parameter	Minimum	Typical	Maximum
Low Level Input Current	15 µA	0 µA	-15 µA
Medium Level Input Current	185 µA	- 200 µA	-215 µA
High Level Input Current	385 µA	- 400 µA	415 µA

Table 2-13 Input Levels for LVDS inputs

Parameter	Conditions	Minimum	Maximum
Input Voltage Range	Vgpd < 950 mV	0 mV	2400 mV
Input Voltage Common Mode Vicm	Vgpd < 950 mV	50 mV	2350 mV
Differential High Input Threshold +Vidth	Vgpd < 950 mV		100 mV
Differential High Input Threshold -Vidth	Rload = 100 Ohms	- 100 mV	
Threshold hysteresis	(+Vid)-(-Vid)	25 mV	
Receiver Input Impedance		100 kOhms	

Table 2-14 CMOS Input Levels

Parameter	Minimum	Typical	Maximum
Low Level Input Voltage			800 mV
High Level Input Voltage	VDD/2 + 100 mV		
Input capacitance		5 pF	

Table 2-15 CMOS Output Level

Parameter	Minimum	Typical	Maximum
Low Level Output Voltage			500 mV
High Level Output Voltage	VDD/2 + 100 mV		

Table 2-16 LVDS/Penn Output Levels

Parameter	Conditions	Minimum	Typical	Maximum
Differential Output Voltage	$R_{L = 100 Ohms}$	250 mV	345 mV	450 mV
Offset Voltage	$R_{L = 100 Ohms}$	1.125 V	1.25 V	1.375 V
Output Voltage High	$R_{L = 100 Ohms}$		1.41 V	1.60 V
Output Voltage Low	$R_{L = 100 Ohms}$	0.90 V	1.07 V	

Table 2-17 Analog Inputs/Outputs

Signal		Current (typ)	Voltage
ext_diff_Bias	Reference for LVDS drivers	250uA (max) current through a xxx size resistor to gnd.	
vrp	Test Pulse Amplitude Reference	125uA current source to GND. Typ 10K res.	
tp_odd	Testpulse output odd channels	Mitch	
tp_even	Testpulse output even channels	Mitch	
threshold<0,1>_high	High Threshold DAC output	5K Output impedance	0 to 1.25V by step of 4.8mV
threshold<0,1>_low	Low Threshold DAC output	5K Output impedance	0 to 1.25V by step of 4.8mV

2.2.19.4 Pad Frame

2.2.19.5 Production Bonding Diagram

2.2.19.6 Test Bonding Diagram

Appendix I. Differential Non-linearity.