2238 MCX-TTL

Features

- 16 channels.
- Input and output capable.
- No galvanic isolation.
- High speed and low jitter.
- MCX connectors.

Applications

- Photon counting.
- External equipment trigger.
- Optical shutter control.

General Description

The 2238 MCX-TTL card is a 4hp EEM module. It adds general-purpose digital I/O capabilities to carrier cards such as 1124 Kasli and 1125 Kasli-SoC.

Each card provides four banks of four digital channels each, with MCX connectors, controlled through 2 EEM connectors. Each EEM connector controls two banks independently. Single EEM operation is possible. The direction (input or output) of each bank can be selected using DIP switches. Each channel supports 50Ω terminations individually controllable using DIP switches. This card can achieve higher speed and lower jitter than the isolated 2118/2128 BNC/SMA-TTL cards.

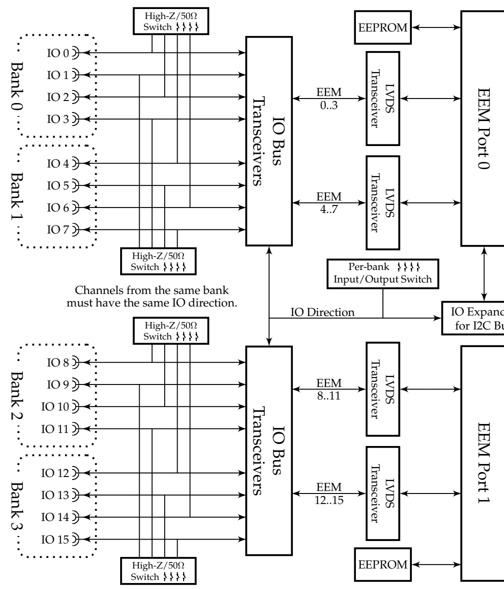

Figure 1: Simplified Block Diagram

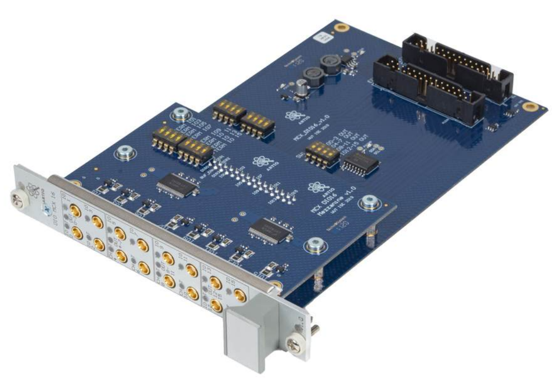

Figure 2: MCX-TTL Card photo

Electrical Specifications

Both recommended operating conditions and electrical characteristics are based on the datasheet of the bus transceivers IC (74LVT162245MTD[1]).

Table 1: Recommended Operating Conditions

Parameter	Symbol	Min.	Typ.	Max.	Unit	Conditions
Input voltage	V_I	0		5.5	V	
High-level output current	I_{OH}			-24	mA	
Low-level output current	I_{OL}			24	mA	
Input edge rate	$\frac{\Delta t}{\Delta V}$	0		10	ns/V	$0.8V \leq V_I \leq 2.0V$

The recommended operating temperature is $-40°C \leq T_A \leq 85°C$. All specifications are in the recommended operating temperature range unless otherwise noted.

Table 2: Electrical Characteristics

Parameter	Symbol	Min.	Typ.	Max.	Unit	Conditions
Input clamp diode voltage	V_{IK}			-1.2	V	$I_I = -36mA$
Input high voltage	V_{IH}	2.0			V	
Input low voltage	V_{IL}			0.8	V	
Output high voltage	V_{OH}	2.0			V	$I_{OH} = -24mA$
		3.1			V	$I_{OH} = -200\mu A$
Output low voltage	V_{OL}			0.8	V	$I_{OL} = -24mA$
				0.2	V	$I_{OL} = -200\mu A$
Input current	I_I			20	µA	$V_I = 5.5V$
				2	µA	$V_I = 3.3V$
				-10	µA	$V_I = 0V$

[1] https://www.onsemi.com/pdf/datasheet/74lvt162245-d.pdf

Configuring IO Direction & Termination

The termination and IO direction can be configured by switches. The per-channel termination and per-bank IO direction switches are found at the top and middle of the card respectively.

Termination switches selects the termination of each channel, between high impedence (OFF) and 50Ω (ON).

IO direction switches partly decides the IO direction of each bank.

- Closed switch (ON)
  Fix the corresponding bank to output. The direction cannot be changed by I^2C.
- Opened switch (OFF)
  Leave the direction configurable by I^2C.

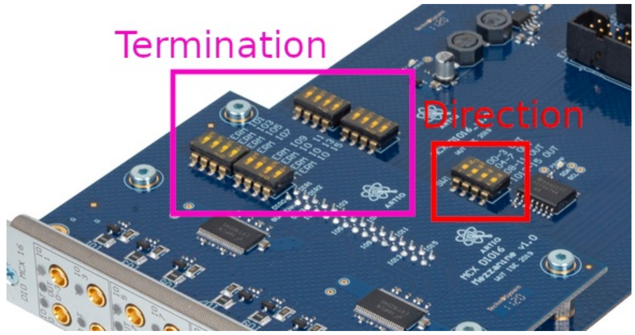

Figure 3: Position of switches

Example ARTIQ code

The sections below demonstrate simple usage scenarios of the 2245 LVDS-TTL card with the ARTIQ control system. They do not exhaustively demonstrate all the features of the ARTIQ system. The full documentation for the ARTIQ software and gateware is available at https://m-labs.hk.

Timing accuracy in the examples below is well under 1 nanosecond thanks to the ARTIQ RTIO system.

One pulse per second

The channel should be configured as output in both the gateware and hardware.

```
@kernel
def run(self):
    self.core.reset()
    while True:
        self.ttl0.pulse(500*ms)
        delay(500*ms)
```

Morse code

This example demonstrates some basic algorithmic features of the ARTIQ-Python language.

```
def prepare(self):
    # As of ARTIQ-6, the ARTIQ compiler has limited string handling
    # capabilities, so we pass a list of integers instead.
    message = ".- .-. - .. --.-"
    self.commands = [{".": 1, "-": 2, " ": 3}[c] for c in message]

@kernel
def run(self):
    self.core.reset()
    for cmd in self.commands:
        if cmd == 1:
            self.led.pulse(100*ms)
            delay(100*ms)
        if cmd == 2:
            self.led.pulse(300*ms)
            delay(100*ms)
        if cmd == 3:
            delay(700*ms)
```

Counting rising edges in a 1ms window

The channel should be configured as input in both the gateware and hardware.

```
@kernel
def run(self):
    self.core.reset()
    gate_end_mu = self.ttl0.gate_rising(1*ms)
    counts = self.ttl0.count()
    print(counts)
```

This example code uses the software counter, which has a maximum count rate of approximately 1 million events per second. If the gateware counter is enabled on the TTL channel, it can typically count up to 125 million events per second:

```
@kernel
def run(self):
    self.core.reset()
    self.edgecounter0.gate_rising(1*ms)
    counts = self.edgecounter0.fetch_count()
    print(counts)
```

Responding to an external trigger

One channel needs to be configured as input, and the other as output.

```
@kernel
def run(self):
    self.core.reset()
    self.ttlin.gate_rising(5*ms)
    timestamp_mu = self.ttlin.timestamp_mu()
    at_mu(timestamp_mu + self.core.seconds_to_mu(10*ms))
    self.ttlout.pulse(1*us)
```

Ordering Information

To order, please visit https://m-labs.hk and select the 2238 MCX-TTL in the ARTIQ Sinara crate configuration tool. The card may also be ordered separately by writing to mailto:sales@m-labs.hk.