1.0 Introduction

The **Ndigo5G** is a digitizer and transient recorder designed to sample relatively shorts pulses in rapid repetition. It produces a stream of output packets, each containing data from a single trigger event together with a timestamp.

By default the **Ndigo5G** is equipped with a passive cooling system. If necessary, the unit might be ordered with an active cooling system on demand (Fig 1.1).

1.1 Features

- **10 bit** dynamic range
- up to **5 Gsps** sample rate (in 1 channel mode)
- up to **4 channels**
- digital input with TDC that can also be used for gating and triggering
- 2nd digital input for gating or triggering.
- PCIe 4x 1.1 with **800 MB/s throughput** for simple and fast data transfer to most PCs.
- multiple boards can be synchronized via reference clock if more channels are required.
- extension board available with 4 additional digital inputs.

Figure 1.1: Ndigo5G equipped with active cooling system

2.0 Hardware

2.1 Installing the Board

The **Ndigo5G board** can be installed in any x4 (or higher amount of lanes) PCIe slot. If the slot electrically supports less than 4 lanes, the board will operate at lowerdata throughput rates.

Please ensure proper cooling of the device. The **Ndigo5G** has an onboard temperature detection. If the ADC chip temperature exceeds 90C a warning is issued to the device driver. In case the temperature is higher than 95C the ADC is disabled to avoid damage. Using a PCI-slot cooler is in many cases an appropriate solution to circumvent problems caused by overheating if the board is used inside a PC. The Ndigo-Crate will provide sufficient cooling under normal operating conditions.

Using a single **Ndigo5G**, no further connections need to be made.

For applications that require more than 4 ADC channels, several Ndigo boards can be operated in sync. Any board of the **Ndigo product line** can be synced to other Ndigo boards, allowing, for instance, for a combination of high speed ADCs (**Ndigo5G**) and slower high resolution ADCs (**Ndigo250M-14**).

The signals used for board synchronization and inter-board triggering are transferred on a bus be-tween the boards. Join all C2 connectors (see Figure Fig 2.3) on the boards using a ribbon cable. Both ends of the bus need to be terminated properly. If using a **Ndigo Crate**, connectors providing the termination are located on the crate mainboard next to the PCIe slots to the extreme left and right. <more details, peas.e refer to the Ndigo Crate user guide. In applications that use only a few Ndigo boards installed directly inside a PC, termination PCBs available from cronologic can be used.

Ndigo5G's standard device driver can be used to read out all boards and acquire data. For more complex scenarios, using the cronoSync-library, which is part of cronoTools, is recommended. The cronoSync library is provided with the Ndigo device driver. Please refer to the cronoTools user guide for more information.

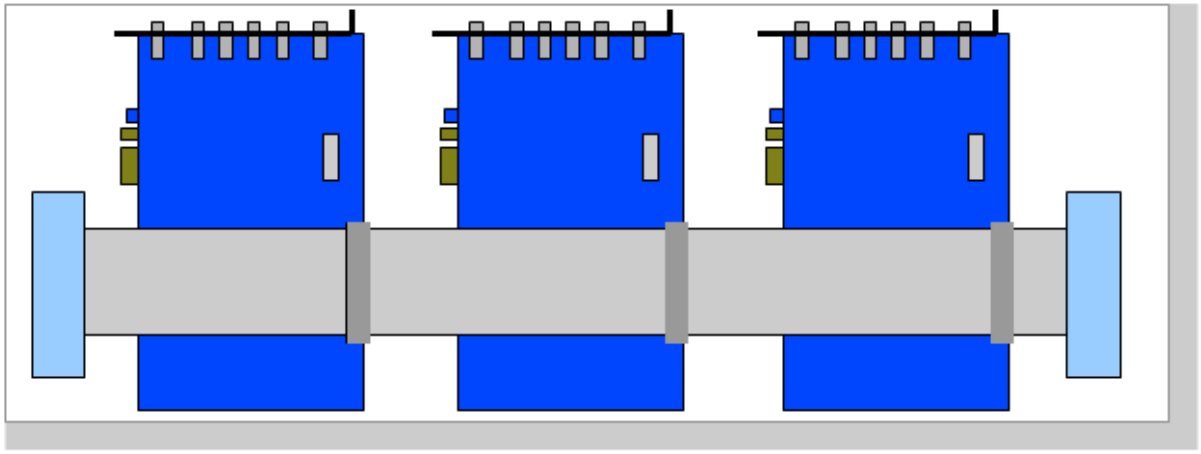

Figure 2.1: If several Ndigo boards are connected to work in sync, the boards must be connected using a ribbon cable as bus for synchronization and trigger signals. Proper termination is required at both ends of the cable.

2.2 Ndigo5G External Inputs and Connectors

2.2.1 Connectors

The inputs of the **Ndigo5G** are located on the PCI bracket. Figure Fig 2.3 on page shows the location of the 4 analog inputs A to D and the two digital inputs G (GATE) and T (Trigger). Furthermore, two board interconnection connectors can be found at the top edge of the **Ndigo5G**, as displayed in Figure Fig 2.3 on page . Connector C1 is used for a board-to-board connection (e. g. to link a HPTDC8-PCI and a **Ndigo5G** via a **Ndigo Extension board**, see chapter 2.3). Connector C2 is used as a bus interface between multiple Ndigo boards distributing clock, trigger and sync signals. Proper termination must be placed at both ends of the bus interconnection ribbon cable.

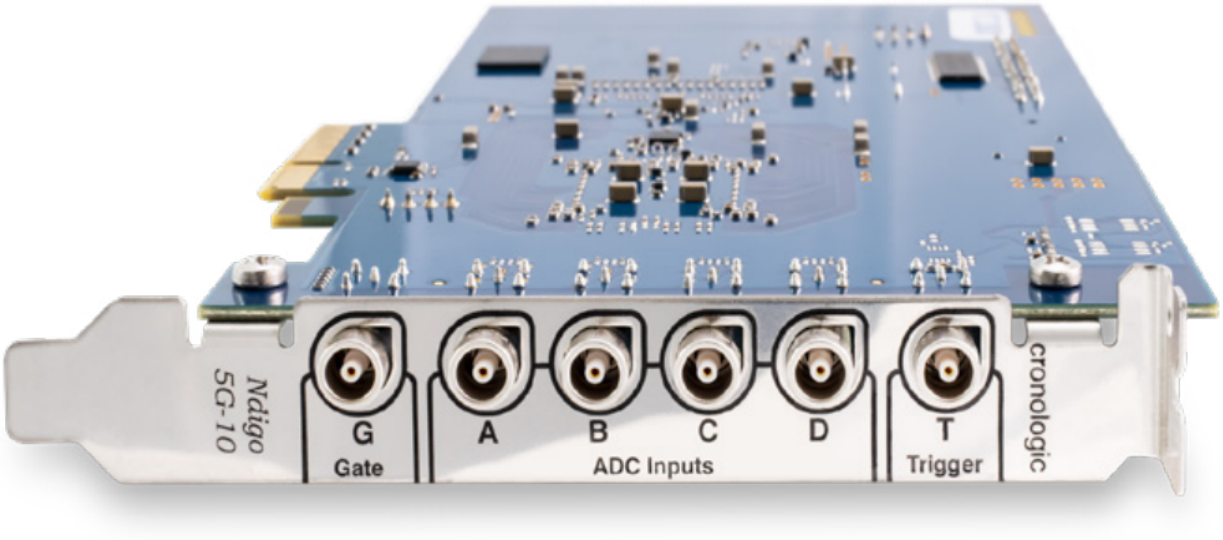

Figure 2.2: Input connectors of an Ndigo5G located on the PCI bracket.

Figure 2.3: Ndigo5G board showing inter-board connectors C1 and C2.

2.2.2 Analog Inputs

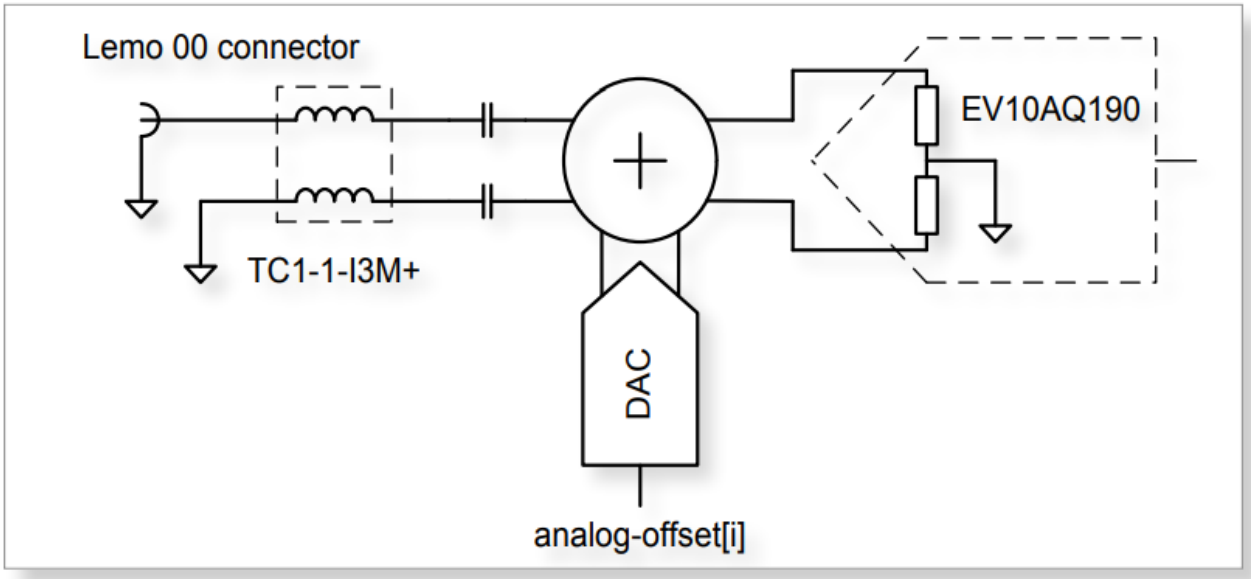

Figure 2.3: Input circuit for each of the four analog channels.

The analog inputs of the ADC are single ended LEMO00 coax connectors. The inputs have a 50Ω impedance and are AC coupled. The inputs are converted to a differential signal using a balun.

Analog Offsets

AC coupling removes the common mode voltage from the input signal. Users can move the common mode voltage to a value of their choice using the analogoffset parameter of each channel before sampling.

This feature is useful for highly asymmetric signals, such as pulses from TOF spectrometers or LIDAR systems. Without analog offset compensation, the pulses would begin in the middle of the ADC range, effectively cutting the dynamic range in half (see Figure Fig 2.6). By shifting the DC baseline to one end of the ADC range, the input range can be used fully, providing the maximum dynamic range. The analog offset can be set between $\pm 0,25V$.

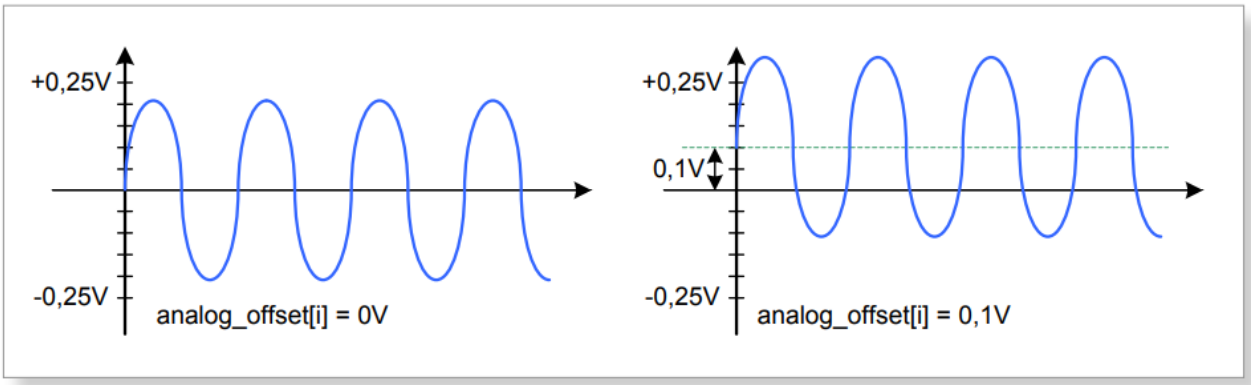

Figure 2.5: Users can add analog offset to the input before sampling

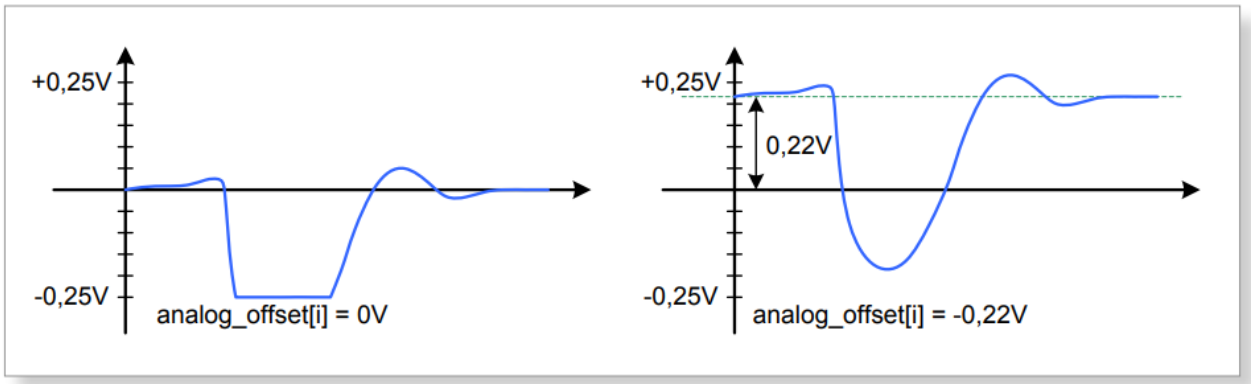

Figure 2.6: Asymmetric signal shifted to increase dynamic range

2.2.3 Digital Inputs

There are two digital inputs on the front slot cover called Trigger and GATE.

Both inputs provide a digital input signal routed to the trigger matrix. These signals can be used to trigger any of the trigger state machines and gating blocks. The inputs are AC coupled. DC offset is configurable via **dc_offset_parameter** in the **configurations structure** to support positive and negative input pulses.

The configuration is set via the structures **trigger[8] and trigger[9]** in the **configuration structure**.The input circuit is shown in Fig 2.17 on page 15.

TDC on Trigger Input

There is a TDC connected to the Trigger input. When used with the TDC, the Trigger input supports negative pulses only . The TDC creates packets of **type 8**. These packets first contain a coarse timestamp and a payload that can be used to calculate the trigger position with higher precision. The function **ndigo_process_tdc_packet()** can be used to replace the coarse timestamp with the precise timestamp. This function is described in section 3.5 on page 40 . TDC pulses must have a minimum duration of 3.3ns. The dead-time of the TDC is 32ns.

2.3 Extension Card

The Ndigo Extension card provides additional inputs or outputs to the FPGA. It is connected to the C1(Samtec QSS-025) connector on an **Ndigo5G** by an Samtec SQCD cable assembly.

The **Ndigo Extension Card** provides up to ten single ended LEMO00 connectors. The circuit connecting to each of these circuits can be chosen to provide inputs or outputs. These can be AC or DC coupled. AC coupled inputs support NIM signaling.

The signals connect to 2.5V IO Pins of the Xilinx Virtex-5 FPGA. The current firmware revision provides the following signal connections.

Table

The 4 digital inputs are routed to the bus inputs of the trigger matrix to be used for triggering. The routing can be configured to either ORing the sync bus and extension channels or use the extension channels exclusively.

Table

2.4 Ndigo5G Functionality

2.4.1 ADC Modes

Depending on board configuration, the analog input signal is quantized to 8 or 10 bits. However, the board always scales and offsets the data to 16 bit signed data centered around 0.

Data processing such as trigger detection or packet building are always performed on **3.2ns** intervals. Depending on the ADC mode, this interval may contain 4, 8 or 16 samples.

The board supports using one, two or four channels:

1 Channel Modes A, B, C and D

In these modes, only a single channel is used. The analog signal on that channel is digitized at 5Gsps. Packet size is always a multiple of 16 samples per **3.2ns** . See Figure 2.9 on page and ` Fig2.15 <#target-link6>`_ on page.

2 Channel Modes AC, BC, AD and BD

In these modes, two channels are used simultaneously. The analog signals on these channels are digitized at **2.5Gsps** each. Packet size is always a multiple of 8 samples per **3.2ns**. See Fig2.8 on page 11 and see ` Fig2.14 <#target-link8>`_ on page 14.

4 Channel Mode ABCD

In this mode, all four channels are digitized independently at **1.25Gsps** each. The packet size is always a multiple of 4 samples per **3.2ns**. See Fig2.7 on page 11 and see ` Fig2.13 <#target-link10>`_ on page 14.

Multiple Sampling Modes AAAA, BBBB, CCCC and DDDD

In these modes, only one analog input channel is used, but the channel is sampled independently and simultaneously by four ADCs at **1.25Gsps**. The board creates four independent streams with 4 samples each per **3.2ns**.

Using the same trigger setting on all ADCs, can be used to reduce noise by averaging the four channels. To deal with complex triggering conditions, different trigger settings on each of the ADCs can be used.

The **Ndigo5G** provides 4 ADCs sampling at **1.25Gsps** each. Higher speed modes are implemented by interleaving two or four of these ADCs.

During interleaving, the **Ndigo5G** firmware reorders and groups the data into a linear sample stream. The process is fully transparent. For users, the only difference is that a **3.2ns** cycle can contain 4, 8 or 16 samples, depending on mode.

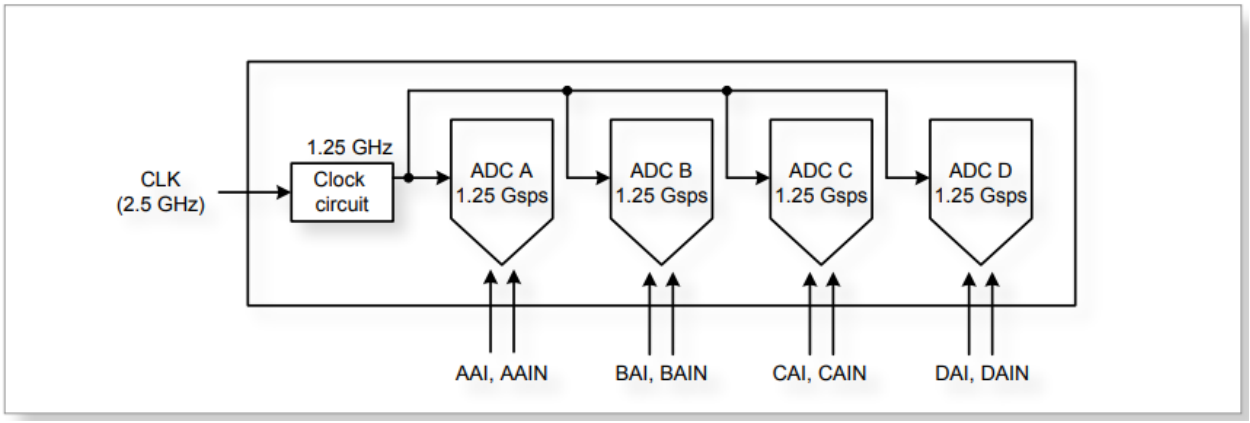

Figure 2.7: ADCs in 4 channel mode ABCD at 1.25Gsps.

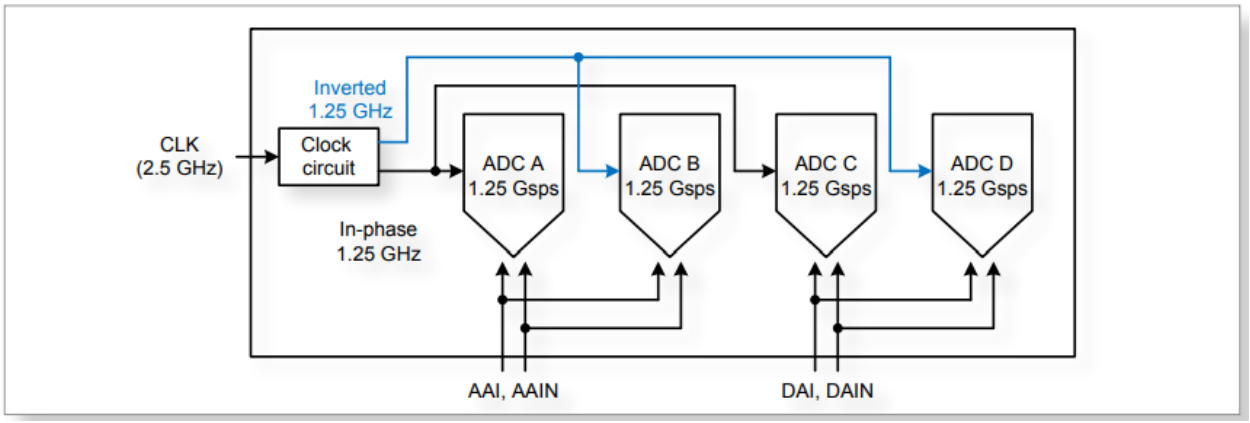

Figure 2.8: ADCs in 2 channel mode AD, interleaved for 2.5Gsps.

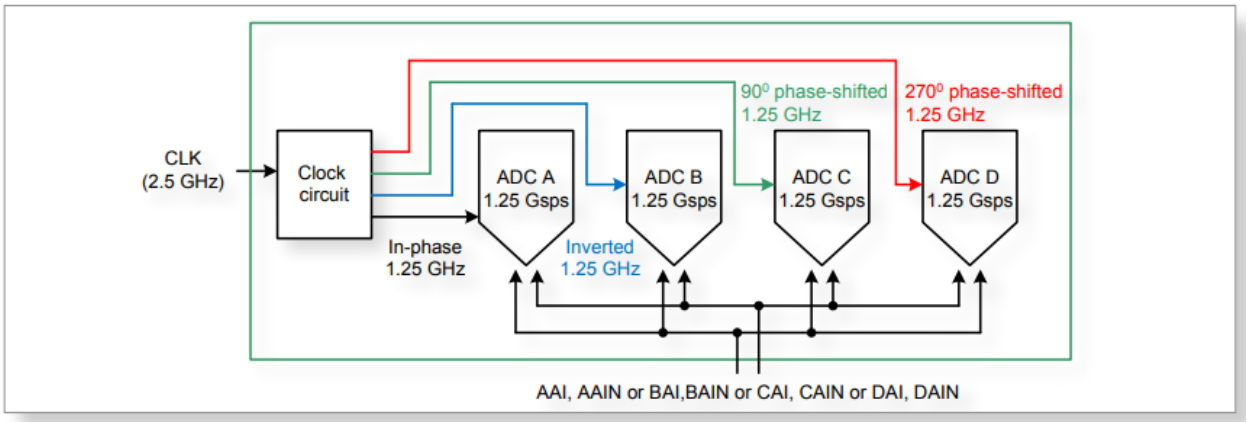

Figure 2.9: ADCs in 1 channel mode A, B, C or D interleaved for 5Gsps.

2.4.2 Zero Suppression

One of **Ndigo5G's** key features is on-board zero suppression to reduce PCIe bus load. Only data that passes specifications predefined by the user is transmitted. This guide refers to transmitted waveform data as "packets". A packet contains the waveform data and a timestamp giving the absolute time (i.e. the time since start of data acquisition) of the packet's last sample.

` Fig2.10 <#target-link11>`_ shows a simple example: Data is written to the PC only if values exceed a specified threshold. Expanding on that, **Ndigo5G's** zero suppression can be used to realize much more complex scenarios.

Figure 2.10: Simple zero suppression: Only data with values above a threshold are written to the PC.

2.4.3 Trigger Blocks

Ndigo5G-10 and **Ndigo5G-8** record analog waveforms using zero suppression. Whenever a relevant waveform is detected, data is written to an internal FIFO memory. Each ADC channel has one trigger block determining whether data is written to the FIFO. The parameters are set in Structure **ndigo_trigger_block** (See chapter 3.4.3 on page 36).

Each trigger block consists of two independent units that check the incoming raw data stream for trigger conditions (` Fig2.10 <#target-link11>`_ on page 12). Users can specify a threshold and can choose whether triggering is used whenever incoming data is below or above the threshold (level triggering) or only if data exceeds the threshold (edge triggering).

A gate length can be set to extend the trigger window by multiples of **3.2ns**. Furthermore, if users choose precursor values > 0, the trigger unit will start writing data to the FIFO precursor 3:2ns before the trigger event.

When using edge triggering, all packets have the same length (` Fig2.11 <#target-link12>`_ on page 13): precursor + length + 1 cycles of **3.2ns**. For level triggering, packet length is data dependent (` Fig 2.12 <#target-link13>`_ on page 14).

Please note that triggering is not accurate to sample. For each **3.2ns** clock cycle, it is determined whether on any sample during that clock cycle a trigger condition is met. The clock cycle is then selected as the trigger point. As a result, the trigger sample can be anywhere within a range of up to 16 samples in single channel mode (` Fig2.15 <#target-link6>`_ on page 15) at 16 samples per **3.2ns** .

If retriggering is active, the current trigger window is extended if a trigger event is detected inside the window.

A trigger block can use several input sources:

- the 8 trigger decision units of all four ADC channels (` Fig2.16 <#target-link14>`_ on page 15)
- the GATE input (` Fig2.17 <#target-link4>`_ on page 15)
- the Trigger or TDC input, (` Fig2.17 <#target-link4>`_ on page 15)
- a function trigger providing random or periodic triggering (Section 2.4.5 on page 20)
- triggers originating from other cards connected with the sync cable or from the Ndigo Extension card (BUS0, BUS1, BUS2, BUS3)
- A second set of trigger units with names ending in pe for the digital inputs Trigger, GATE, BUS0, BUS1, BUS2, and BUS3 configured for positive edge triggering. Together with the regular trigger units on this inputs, both edges of a pulse can be used in the trigger logic. This set of triggers is not available as inputs for the gate blocks.

Trigger inputs from the above sources can be concatenated using logical OR (` Fig 2.19 <#target-link15>`_ on page 16) by setting the appropriate bits in the trigger blocks source mask.

Triggers can be fed into the gate blocks described on page (` Fig 2.20 <#target-link16>`_ Gate blocks can be used to block writing data to the FIFO. That way, only zero suppressed data occurring when the selected gate is active is transmitted. This procedure reduces PCIe bus load even further (` Fig 2.20 <#target-link16>`_).

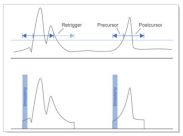

Figure 2.10: Simple zero suppression: Only data with values above a threshold are written to the PC.

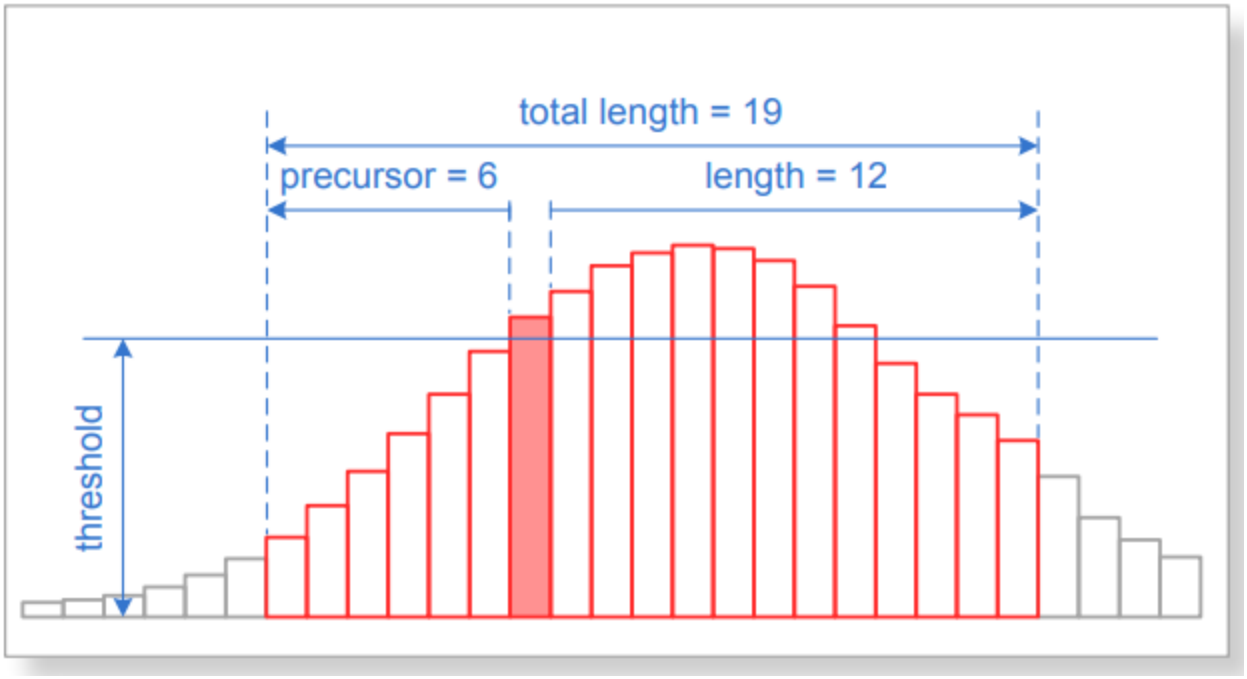

Figure 2.11: Parameters for edge triggering

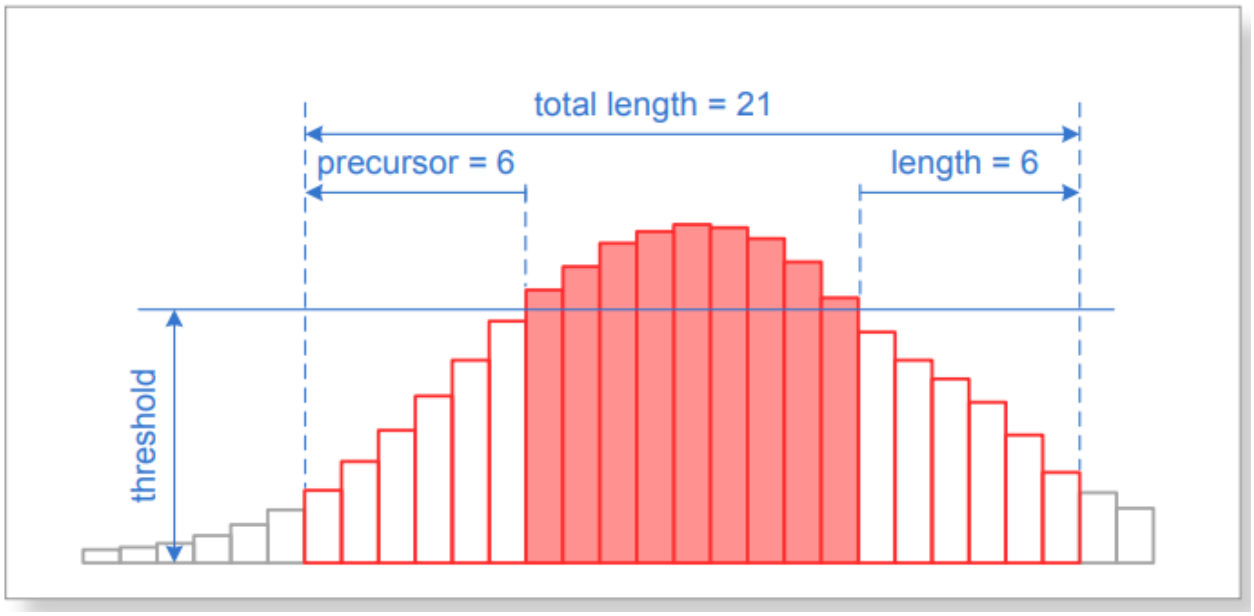

Figure 2.12: Parameters for level triggering

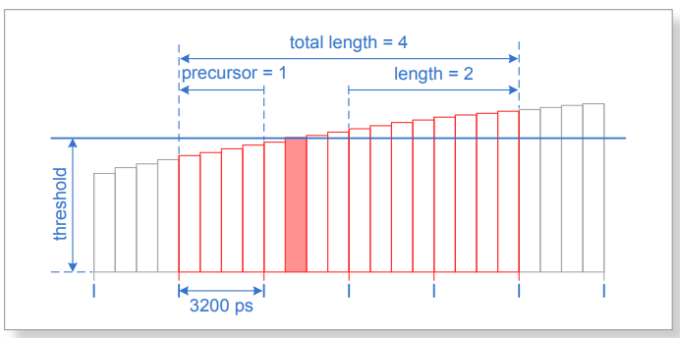

Figure 2.13: Triggering in 4 channel mode at 4 samples per clock cycle.

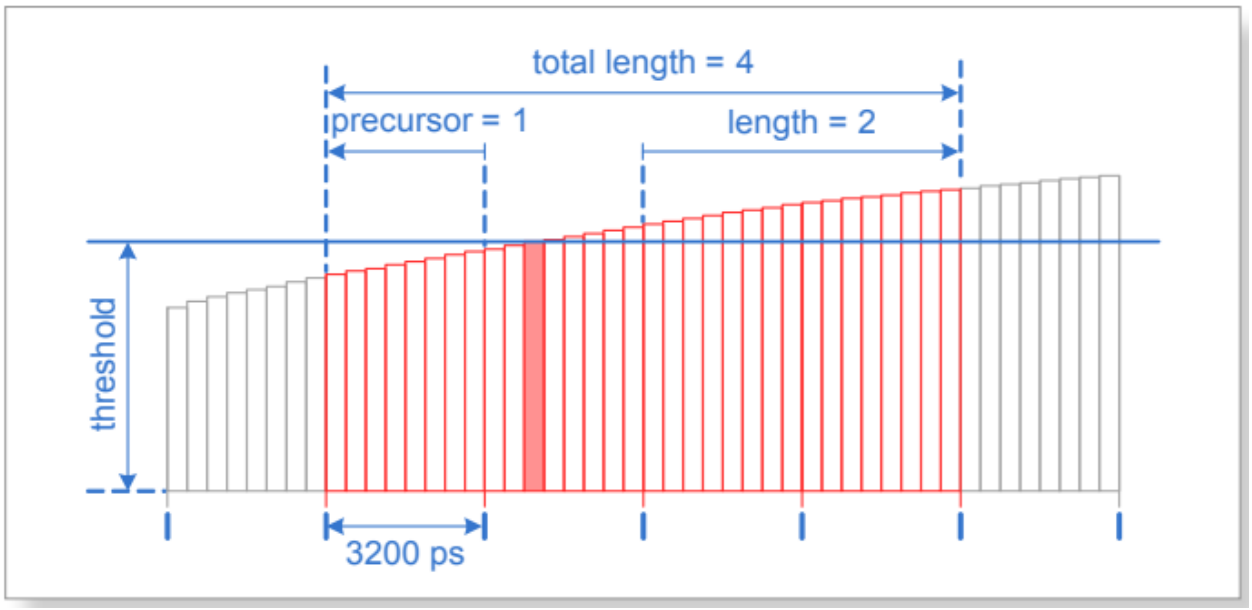

Figure 2.14: Triggering in 2 channel mode at 8 samples per clock cycle.

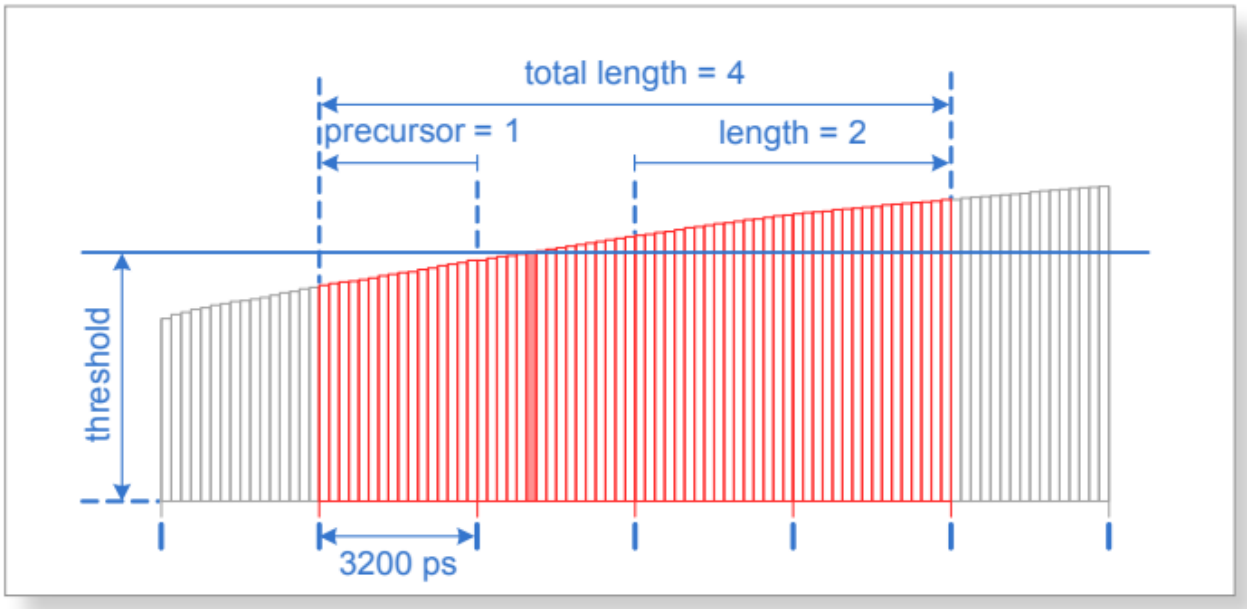

Figure 2.15: Triggering in 1 channel mode at 16 samples per clock cycle.

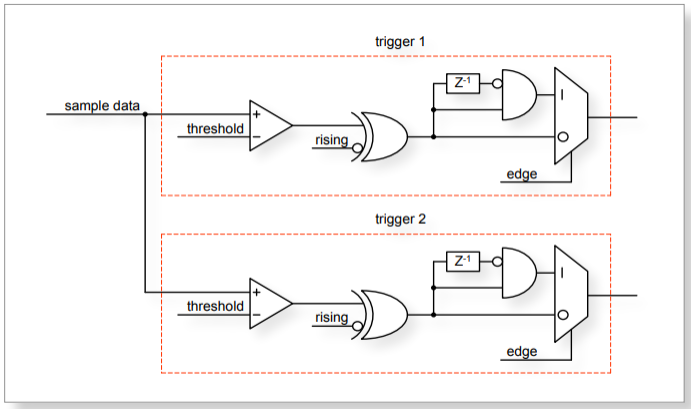

Figure 2.16: From the ADC inputs, a trigger unit creates an input flag for the trigger matrix. Each digitizer channel (A, B, C, D) has two trigger units.

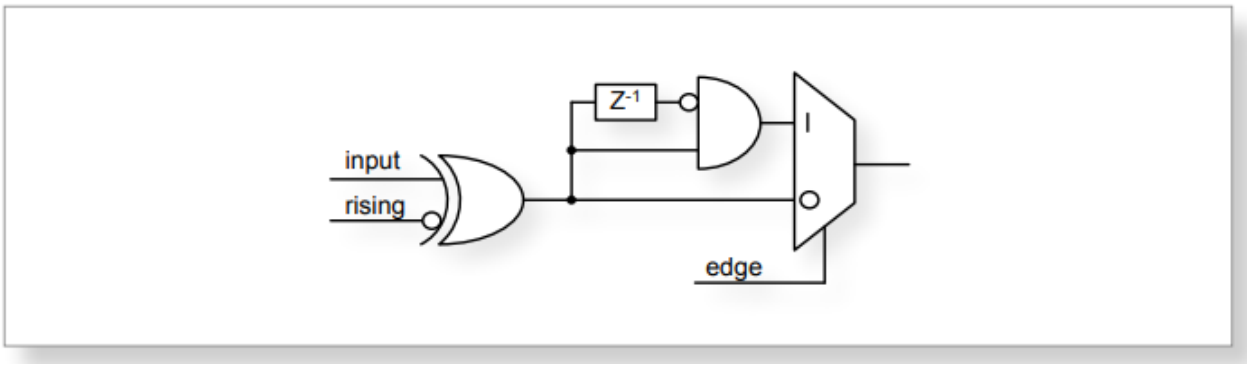

Figure 2.17: The digital inputs Trigger, GATE, BUS0, BUS1, BUS2 and BUS3 have simpler trigger units.

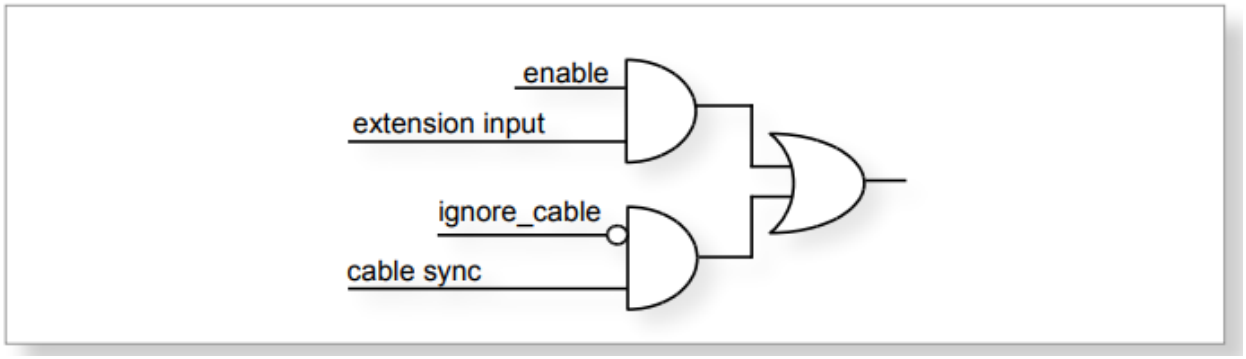

Figure 2.18: The extension block combines signals from the optional extension board and the sync cable.

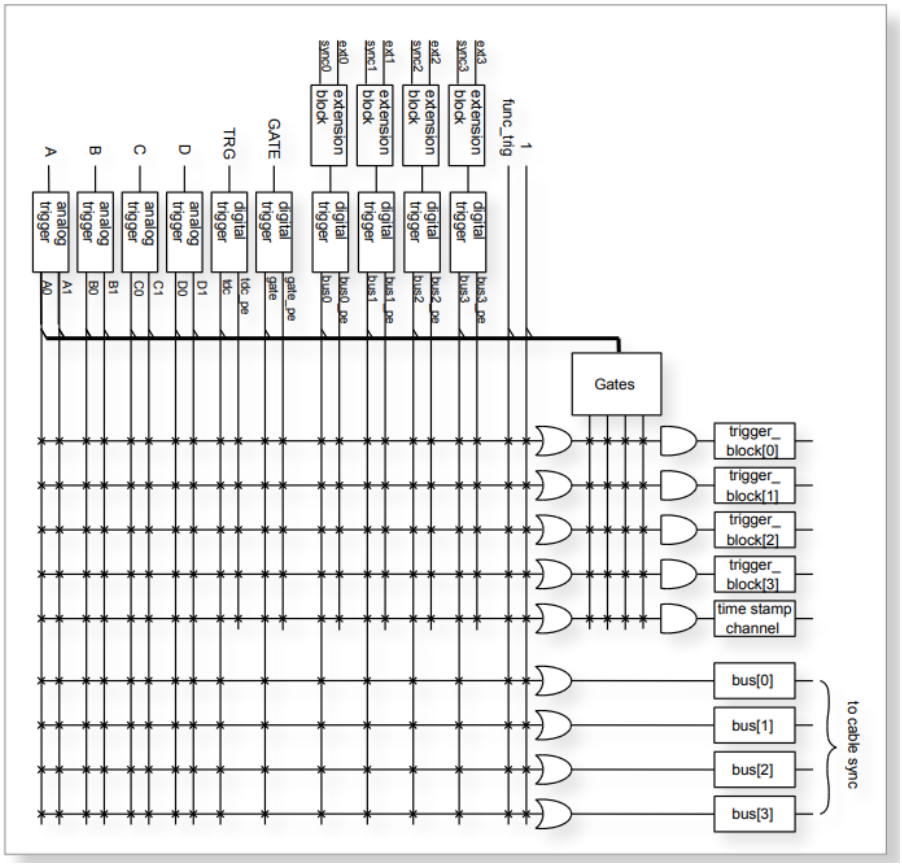

Figure 2.19: Trigger Matrix: The trigger signals of each ADC channel, the trigger input, the GATE input or the sync cable can be combined to create a trigger input for each trigger block. The four gate signals can be used to suppress triggers during certain time frames.

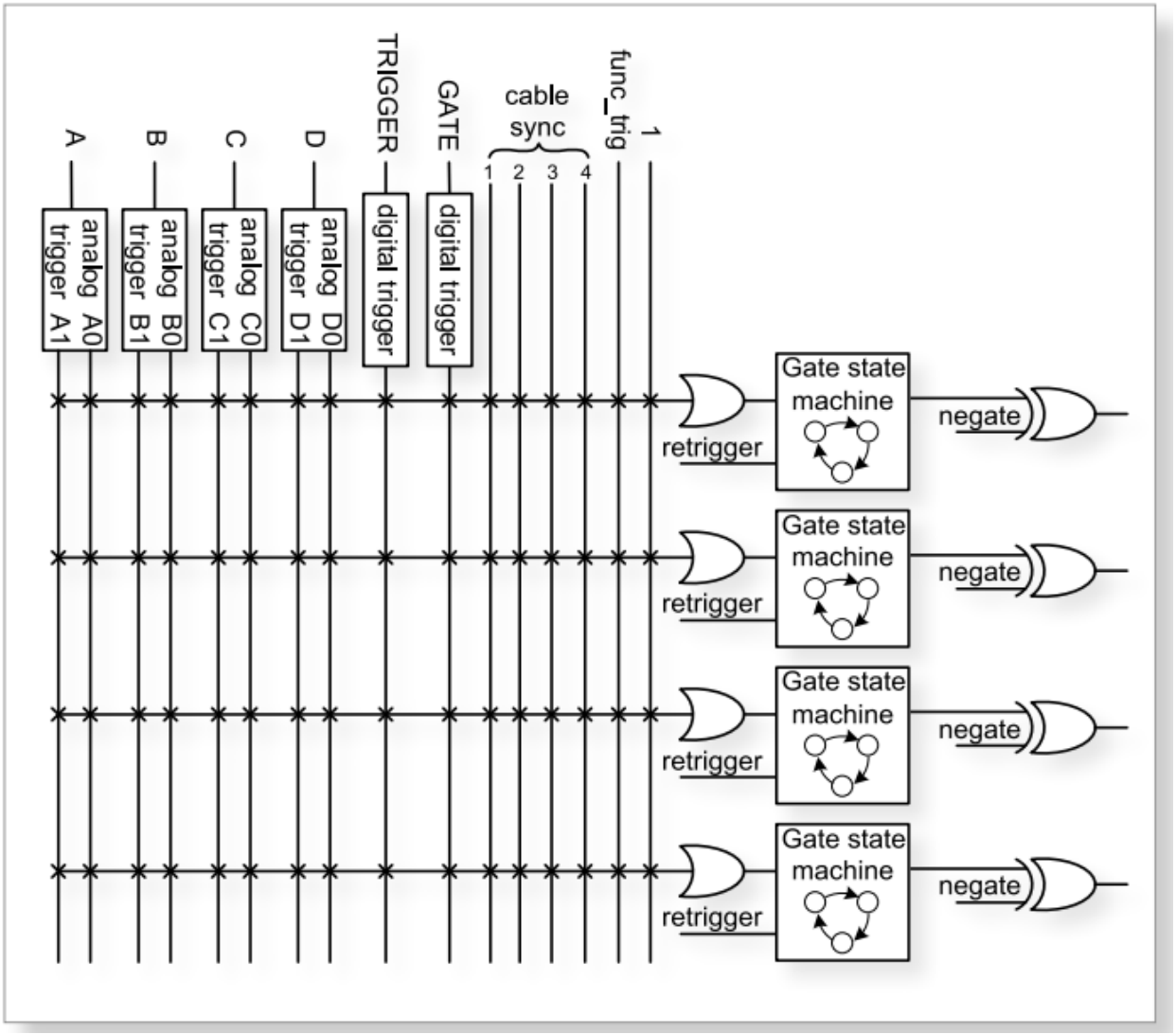

Figure 2.20: Gating Blocks: Each gating block can use an arbitrary combination of inputs to trigger its state machine. The outputs can be individually inverted and routed to the AND-gate feeding the trigger blocks.

4.0 Packet Format

4.0.1 Output Structure ndigopacket

0 to 3 for the ADC input channels, 4 for the TDC, 5 for the timestamp channel.

Identifies the source card in case there are multiple boards present. Defaults to 0 if no value is assigned to the parameter boardid in Structure ndigoinitparameters or set via
int ndigosetboardid(ndigodevice *device, int boardid).

For the ADC channels this is set to 1 to signify 16 bit signed data.

For the TDC channel it is set to 8 to signify 64 bit unsigned data.

If the type field is 128 or greater then there is no data present, even if length is not 0. In this cases the length field may contain other data.

Type	Length Field	Description
1	Number of payload words	16 bit signed samples from one of the ADCs
8	Number of payload words	64 Bit unsigned TDC Data, only for internal processing
128	Bit pattern of trigger sources	Whenever at least one of the sources that is enabled for the timestamp channel triggers, one of these packets is generated. The length field contains the triggers active when this packet was created.

1
If the bit with weight 1 is set, the packet was truncated because the internal FIFO was full. Less than the requested number of samples have been written due to the full FIFO.

2
If the bit with weight 2 is set, there are lost triggers immediately preceding this packet due to insufficient DMA buffers. The DMA controller has discarded packets due to full host buffer.

4
If the bit with weight 4 is set, the packet contains ADC sample overflows.

8
If the bit with weight 8 is set, there are lost triggers immediately preceding this packet due to insufficient buffers. The trigger unit has discarded packets due to full FIFO.

16
If the bit with weight 16 is set, the internal DMA FIFO was full. Triggers only got lost if a subsequent package has the bit with weight 8 set.

32
If the bit with weight 32 is set, the host buffer was full. Triggers only got lost if a subsequent package has the bit with weight 8 set.

64
If the bit with weight 64 is set, the packet from the TDC does not contain valid data and the timestamp is not corrected. No valid edge was found in TDC packet.

Number of 64-bit elements (each containing 4 samples) in the data array if type < 128.

If type = 128 this is the pattern of trigger sources that where active in the clock cycle given by the timestamp. Bits are set according to the trigger sources, i.e. bit 0 is set if trigger A0 was active, bit 29 is set if trigger BUS3PE was active. Use the NDIGOTRIGGERSOURCEdefines to check for the bits set.

ADC channels A to D: Timestamp of the last word in the packet in ps.

TDC: Timestamp of the trigger event (falling edge) on the TDC channel in ps. When ndigoprocesstdcpacket() is called once on the packet the timestamp is replaced with the precise timestamp for the edge.

Timestamp channel: Timestamp of the trigger event in ps.

Sample data. For the Ndigo5G each 64 bit word contains four 16 bit signed words from the ADC. The user can cast the array to short* to directly operate on the sample data.

5.0 C-Example

```
#include "Ndigo_interface.h"
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char* argv[])
{
    ndigo_init_parameters params;
    ndigo_get_default_init_parameters(&params);

    params.card_index = 0;
    params.buffer_size[0] = 1<<23;
    params.drive_external_clock = true;
    params.is_slave = false;
    params.use_external_clock = false;

    int error_code;
    const char*error_message;
    ndigo_device* ndgo = ndigo_init(&params, &error_code, &error_message);
    if( error_code != NDIGO_OK ) {
        printf("\nError %d: %s\n", error_code, error_message);
        exit(-1);
    }

    ndigo_configuration config;
    ndigo_get_default_configuration(ndgo, &config);
    config.adc_mode = NDIGO_ADC_MODE_ABCD;

    // disable unused trigger blocks
    config.trigger_block[1].enabled = false;
    config.trigger_block[2].enabled = false;
    config.trigger_block[3].enabled = false;
    config.trigger_block[4].enabled = false;

    // configure trigger block 0
    config.trigger_block[0].enabled = true;
    config.trigger_block[0].minimum_free_packets = 1.0;
    config.trigger_block[0].precursor = 0;
    config.trigger_block[0].retrigger = 0;

    config.trigger_block[0].sources = NDIGO_TRIGGER_SOURCE_A0;
    config.trigger_block[0].length = 16;
    config.trigger_block[0].gates = NDIGO_TRIGGER_GATE_NONE;

    config.analog_offset[0] = 0.1;

    config.trigger[NDIGO_TRIGGER_A0].edge = true;
    config.trigger[NDIGO_TRIGGER_A0].rising = false;
    config.trigger[NDIGO_TRIGGER_A0].threshold = 0;

    if( ndigo_configure(ndgo, &config) != NDIGO_OK ) {
        printf("\nFatal configuration error. Aborting...\n");
        exit(-1);
    }

    ndigo_start_capture(ndgo);

    // counts the number of packets received
    int count = 0;

    while( count < 10 ) {
        ndigo_read_in in;
        // Do not wait for data
        // (if set to 1 the ndigo_acknowledge function has to be removed)
        in.acknowledge_last_read = 0;
        ndigo_read_out out;
        int result = ndigo_read(ndgo, &in, &out);
        if( !result ) {
            // buffer received with one or more packets
            ndigo_packet *packet = out.first_packet;
            while( packet <= out.last_packet ) {
                int length = 0;
                if( !(packet->type & NDIGO_PACKET_TYPE_TIMESTAMP_ONLY) )
                    length = packet->length;

                printf("Card %02x, Channel %02x, Flags %02x, Length %6d, Timestamp %llu \n", packet->card, packet->channel, packet->flags, length, packet->timestamp);
                if( !(packet->type & NDIGO_PACKET_TYPE_TIMESTAMP_ONLY) ){
                    short* data = (short*) packet->data;
                    for( inti = 0; i < packet->length * 4; i++ )
                        printf("%6d, ", *(data++));
                    printf("\n\n");
                }
                // current packet pointer is invalid after call to ndigo_acknowledge
                ndigo_packet *next_packet = ndigo_next_packet(packet);
                ndigo_acknowledge(ndgo, packet);
                packet = next_packet;
                count++;
            }
        }
    }
    ndigo_close(ndgo);
    return0;
}
```

6.0 Technical Data

Input Passband: 4.5MHz to 950MHz.

Power Requirements: 25W

Mechanical Dimensions: 170mm × 106mm

Throughput: 800MByte/s on PCIe x4

6.1 Digitizer Characteristics

Each board is tested against the values listed in the ‘Min’ column. ‘Typical’ is the mean value of the first 10 boards produced.

6.1.1 1-Channel-Mode (5Gsps)

Symbol	Parameter	Min	Typical	Max	Units
THD1	Total Harmonic Distortion		-60	-56	dB
SNR1	Signal to Noise Ration	47	49		dB
SFDR1	Spurious Free Dynamic Range (including Harmonics)	55	59		dB
SFDR1	Spurious Free Dynamic Range (excluding Harmonics)	55	60		dB
SINAD1	Signal-to-Interference Ratio including Noise and Distortion	47	48		dB
ENOB1	Effective Number of Bits	7.5	7.7		

6.1.2 2-Channel-Mode (2.5 Gsps)

Symbol	Parameter	Min	Typical	Max	Units
THD2	Total Harmonic Distortion		-60	-56	dB
SNR2	Signal to Noise Ration	49	51		dB
SFDR2	Spurious Free Dynamic Range (including Harmonics)	58	60		dB
SFDR2	Spurious Free Dynamic Range (excluding Harmonics)	58	61		dB
SINAD2	Signal-to-Interference Ratio including Noise and Distortion	49	50		dB
ENOB2	Effective Number of Bits	7.8	8.1		

6.1.3 4-Channel-Mode (1.25 Gsps)

Symbol	Parameter	Min	Typical	Max	Units
THD4	Total Harmonic Distortion		-60	-56	dB
SNR4	Signal to Noise Ration	49	51		dB
SFDR4	Spurious Free Dynamic Range (including Harmonics)	58	60		dB
SFDR4	Spurious Free Dynamic Range (excluding Harmonics)	68	73		dB
SINAD4	Signal-to-Interference Ratio including Noise and Distortion	49	51		dB
ENOB4	Effective Number of Bits	7.9	8.1		

6.2 Electrical Characteristics

6.2.1 Oscillator

The Ndigo5G uses an OCXO oscillator with 25ppb stability. After power up the oscillator needs to run for 10 minutes to reach this stability.

6.2.2 Environmental Conditions for Operation

The board is designed to be operated under the following conditions:

Symbol	Parameter	Min	Typical	Max	Units
T	ambient temperature	5		40	:math :^{*circ*}C
RH	relative humidity at 31:math :^{*circ*}C	20		75	%

6.2.3 Environmental Conditions for Storage

The board shall be stored between operation under the following conditions:

Symbol	Parameter	Min	Typical	Max	Units
T	ambient temperature	-30		60	:math :^{*circ*}C
RH	relative humidity at 31:math :^{*circ*}C non condensing	10		70	%

6.2.4 Power Supply

Symbol	Parameter	Min	Typical	Max	Units
I	PCIe 3,3V rail power consumption			4	mA
VCC	PCIe 3,3V rail power supply	3,1	3,3	3,6	V
I	PCIe 12V rail power consumption			2,1	A
VCC	PCIe 12V rail power supply	11,1	12	12,9	V
I	PCIe 3,3VAux rail power consumption		0		A
VCC	PCIe 3,3VAux rail power supply		3,3		V

6.2.5 Analog Input

AC coupled single-ended analog inputs (standard version).

Symbol	Parameter	Min	Typical	Max	Units
V	Peak to peak input voltage			0.5	V
Z	input impedance		50		Ω
	Analog offset	-0,25		0.25	V

AC coupled differential analog inputs (S version).

Symbol	Parameter	Min	Typical	Max	Units
V	Input common mode	-4		6	V
V	Differential input Voltage	-125		125	mV
Z	Input impedance		100		Ω
	Analog offset	-0,25		0.25	V

Analog inputs

Single ended AC coupled inputs Trigger and GATE with configurable DC offset bias.

Symbol	Parameter	Min	Typical	Max	Units
V	Pulse height			5.0	V
V	DC offset	-1.25		1.25	V
V	DC offset for TDC	-1.25		-0.01	V
Z	input impedance		50		Ω
t	pulse width	7		100	ns

6.3 Information Required by DIN EN 61010-1

6.3.1 Manufacturer

The Ndigo5G is a product of:

cronologic GmbH & Co. KG
Jahnstraße 49
60318 Frankfurt

HRA 42869 beim Amtsgericht Frankfurt/M

VAT-ID: DE235184378

6.3.2 Intended Use and System Integration

The devices are not ready to use as delivered by cronologic. It requires the development of specialized software to fulfill the application of the end user. The device is provided to system integrators to be built into measurement systems that are distributed to end users. These systems usually consist of a the Ndigo5G, a main board, a case, application software and possible additional electronics to attach the system to some type of detector. They might also be integrated with the detector.

The Ndigo5G is designed to comply with DIN EN 61326-1 when operated on a PCIe compliant main board housed in a properly shielded enclosure. When operated in a closed standard compliant PC enclosure the device does not pose any hazards as defined by EN 61010-1.

Radiated emissions, noise immunity and safety highly depend on the quality of the enclosure. It is the responsibility of the system integrator to ensure that the assembled system is compliant to applicable standards of the country that the system is operated in, especially with regards to user safety and electromagnetic interference. Compliance was only tested for attached cables shorter than 3m.

When handling the board, adequate measures have to be taken to protect the circuits against electrostatic discharge (ESD). All power supplied to the system must be turned off before installing the board.

6.3.3 Cooling

The Ndigo5G in its base configuration has passive cooling that requires a certain amount of air flow. If the case design can't provide enough air flow to the board, a slot cooler like Zalman ZM-SC100 can be placed next to the board. Active cooling is also available as an option to the board.

6.3.4 Environmental Conditions

6.3.5 Inputs

All inputs are AC coupled. The inputs have very high input bandwidth requirements and therefore there are no circuits that provide over voltage protection for these signals. Any voltage on the inputs above 5V or below -5V relative to the voltage of the slot cover can result in permanent damage to the board.

6.3.6 Known Bugs

The Ndigo5G does not work in most Thunderbolt PCIe extension enclosures. The reason is unknown.

Workarounds

Use Ndigo6G

All other cronologic products work reliably in Thunderbolt enclosures. The Ndigo6G offers very similar functionality to the Ndigo5G at a higher performance. When using the Ndigo6G as a replacement, there are some software changes required in the device configuration. The readout data format and API is identical.
See www.cronologic.de/products/adcs/ndigo6g-12 for details.

Use Ndigo Crate

Up to eight Ndigo5G can be used in an Ndigo Crate connected to a PC. Electrically the setup is similar to an external Thunderbolt enclosure, but the PC must have a vacant PCIe slot.
See www.cronologic.de/products/pcie/pcie-crates for details.

All other cronogic products work reliably in Thundberbolt enclosure. Consider using an Ndigo6G as a replace

Recycling

cronologic is registered with the "Stiftung Elektro-Altgeräte Register" as a manufacturer of electronic systems with Registration ID DE 77895909.

The Ndigo5G belongs to category 9, "Überwachungs und Kontrollinstrumente für ausschließlich gewerbliche Nutzung". The last owner of an Ndigo5G must recycle it, treat the board in compliance with §11 and §12 of the German ElektroG, or return it to the manufacturer's address listed on page .

Export Control

The Ndigo5G product line is a dual use item under Council Regulation (EC) No 428/2009 of 5 May 2009 setting up a Community regime for the control of exports, transfer, brokering and transit of dual-use items in section 3A002h. Similar regulations exist in many countries outside Europe.

An export permit is required to export this product from the European Community (EC) which will cause additional lead time. When ordering from outside the EC, the seller will ask you for additional information needed to obtain this permit.

Before reexporting an Ndigo5G or any product containing an Ndigo5G as a component please check you local regulations whether an export permit is required.

7.0 Revision History

7.1 Firmware

Revision	Date	Comments
	2022-05-02	Support for Ndigo5G-S custom board variant
name: fwrev		
1.5412		
1.4937	2019-04-01	TiGer added
1.4865	2015-07-28	Internal optimizations
1.4824	2015-02-27	Fixed intel PCIe link training issues

7.2 Driver & Applications

Revision	Date	Comments
1.4.5	2022-09-12	Fixed bluescreen triggered by hot unplugging
1.4.4	2022-05-12	Fixed broken backup in firmware tool and display errors in NdigoScope
1.4.3	2019-10-21	Fixed a card initialization error in x64 32 mode
1.4.0	2019-06-04	Added Windows 10 support
1.3.0	2017-06-08	NdigoScope application now supports Ndigo250M-14

7.3 User Guide

Revision	Date	Comments
1.2.2	2022-11-04	Documented bug: Ndigo5G does not work in Thunderbolt enclosures Reduced temperature range Improved ADC characteristics due to tighter production tests Added dual use disclaimer Clarified pe positive edge triggers Corrected polarity for dcoffset Corrected extension card clock from
1.2.1	2020-09-20	Cosmetic changes
1.1.3	2020-11-27	Dual-Use information
1.1.2	2019-10-27	
1.1.0	2019-08-27	API clarifications