Przedsiębiorstwo Badawczo-Produkcyjne
OPTEL Sp. z o.o.
ul. Morelowskiego 30
PL-52-429 Wrocław
tel.: +48 (071) 329 68 54
fax.: +48 (071) 329 68 52
e-mail: optel@optel.pl
http://www.optel.pl

Wrocław

OPBOX ver 2.1

Miniature ultrasonic data acquisition system with integrated pulser&receiver

Description and manual

Valid for hardware with firmware: **Rev. 2.1.60**

Document version: 1v3
Date of creation: 2016-08-08

Content

1. Technical parameter

Parameter of analogue receiver:	
Adjustable amplifier:	-31dB to 65dB (step 0.5dB)
Switchable preamplifier:	+24dB
Switchable attenuator:	-20dB
Input voltage:	± 275mV; ± 2.0V with attenuator -20dB.
Bandwidth:	0.5 MHz - 25 MHz (-3dB)
Switchable hardware filters (-3dB):	0.5 - 6MHz, 0.5 - 10MHz, 0.5 - 15MHz, 0.5 - 25MHz, 1 – 6MHz, 1 - 10MHz, 1 - 15MHz, 1 - 25MHz, 2 - 6MHz, 2 – 10MHz, 2 – 15MHz, 2 - 25MHz, 4 - 6MHz, 4 - 10MHz, 4 - 15MHz, 4 – 25MHz.

Pulser (short circuit pulser):	
Voltage adjustment:	0V - 360V (positive pulse);
Loading time:	Adjustable from 0 do 3.1µs step 0.1µs ;
Short circuit time:	<= 20 ns;
Bandwidth:	Up to 50MHz.

A/D converter:	
Resolution:	10 bit (8-bits are stored);
Maximum input voltage:	±0.5V;
Sampling frequency - switchable (MHz):	100; 50; 33.3; 25; 20; 16.7; 14,3; 12,5;
	11,1; 10; 9,1; 8,3; 7,7; 7,14 i 6,67;
Data buffer:	1 - 262090 (256k) samples;
Measuring frame delay:	0 - 65535 sample periods.

Hardware data processing:	
Data representation:	RF, Absolute;
Measurements in defined frames:	3 peak detectors;
	3 level comparators – modes of transition detection: Level, Rising, Falling, Transition.

DAC (TGC) with arbitrary function generator:	
Sampling frequency	100MHz;
Resolution	8 bit;
Max. amplification step	48dB pro sample.

Trigger:	
Software	Software command;
Internal	Programmable timer;
External	2 incremental encoder modules; 2 TTL inputs.
Counter modules for incremental encoders	
	2 modules with CHA, CHB and IDX inputs;

Position counter	32-bit;
Special functions	Programmable input modes: 1X, 2X, 4X; Programmable triggering on position; Storing of position after triggering.

Transducer connectors:	
PE (sending&receiving)	BNC or Lemo;
TT (only receiving)	BNC or Lemo.

CONTROL connector:			
Type:	DB15 female		
DB15 pin	Description	Standard	Description
1	GPO 0	TTL 5V	Digital input GPO0 or synchronization output SYNC_OUT
2	GPO 2	TTL 5V	Digital output GPO2
3	GND		Ground
4	GPI 1	TTL/LVTTL	Digital input GPI1 or trigger input EXT_Y or CHB input for encoder module ENC1
5	GPI 3	TTL/LVTTL	Digital input GPI3 or CHA input for encoder module ENC2
6	GPO 5	TTL 5V	Digital output GPO 5
7	Reserved		Reserved
8	GPI 5	TTL/LVTTL	Digital input GPI5 or IDX input for encoder module ENC2
9	GPO 1	TTL 5V	Digital output GPO1
10	GPO 3	TTL 5V	Digital output GPO3
11	GPI 0	TTL/LVTTL	Digital input GPI0 or trigger input EXT_X or CHA input for encoder module ENC1
12	GPI 2	TTL/LVTTL	Digital input GPI2 or IDX input for encoder module ENC1
13	GPO 4	TTL 5V	Digital output GPO4
14	GND		Ground
15	GPI 5	TTL/LVTTL	Digital inputGPI4 or CHB input for encoder module ENC2

USB interface:	
Connector type:	USB B socket
Standard:	USB-2.0 High Speed (480MBps)
Vendor ID:	0x0547
Product ID:	0x1003
Hardware version:	2.1 (0x0201)
Power supply:	Bus powered <500mA
USB device class:	Vendor
Available endpoints:	
Endpoint 0 (control register)	Direction: IN/OUT, Type: Control, Packets: max. 64Bytes;
Endpoint 2 (storing TGC curves)	Direction: OUT, Type: Bulk, Packets: max.512Byte;
Endpoint 6 (Reading of measurement data)	Direction: IN, Type: Bulk, Pakckets: max. 512Bytes.

2. List of commands and control registers

OPBOX-2.1 is controlled via USB-2.0 with the help of direct commands and a set of control registers.

Direct commands:		
No	**Name**	**Vendor Request**
1	OPBOX_SN	(0xD0)
2	RESET	(0xD1)
3	RESET_FIFO	(0xD2)
4	DIRECT_SW_TRIG	(0xD3)
5	DIRECT_ACK	(0xD4)
6	DIRECT_DATA_READY	(0xD5)
7	PULSE_AMPLITUDE	(0xD6)
8	USB_MODE	(0xD7)

Control registers:					
No	**Name**	**Address**	**No**	**Name**	**Address**
1	DEV_REV	0x00	33	PDB_START_L	0x40
2	POWER_CTLR	0x02	34	PDB_START_H	0x42
3	PACKET_LEN	0x04	35	PDB_STOP_L	0x44
4	FRAME_IDX	0x06	36	PDB_STOP_H	0x46
5	FRAME_CNT	0x08	37	PDB_REF_VAL	0x48
6	CAPT_REG	0x0A	38	PDB_REF_POS_L	0x4A
7	GP_INPUTS	0x0C	39	PDB_REF_POS_H	0x4C
8	GP_OUTPUTS	0x0E	40	PDB_MAX_VAL	0x4E
9	TRIGGER	0x10	41	PDB_MAX_POS_L	0x50
10	TRG_OVERRUN	0x12	42	PDB_MAX_POS_H	0x52
11	XY_DIVIDER	0x14	43	PDC_START_L	0x54
12	TIMER	0x16	44	PDC_START_H	0x56
13	TIMER_CAPT	0x18	45	PDC_STOP_L	0x58
14	ANALOG_CTRL	0x1A	46	PDC_STOP_H	0x5A
15	PULSER_TIME	0x1C	47	PDC_REF_VAL	0x5C
16	BURST	0x1E	48	PDC_REF_POS_L	0x5E
17	MEASURE	0x20	49	PDC_REF_POS_H	0x60
18	DELAY	0x22	50	PDC_MAX_VAL	0x62
19	DEPTH_L	0x24	51	PDC_MAX_POS_L	0x64
20	DEPTH_H	0x26	52	PDC_MAX_POS_H	0x66
21	CONST_GAIN	0x28	53	ENC1_CTRL	0x68
22	PEAKDET_CTRL	0x2A	54	ENC1_POS_L	0x6A
23	PDA_START_L	0x2C	55	ENC1_POS_H	0x6C
24	PDA_START_H	0x2E	56	ENC1_CAPT_L	0x6E
25	PDA_STOP_L	0x30	57	ENC1_CAPT_H	0x70
26	PDA_STOP_H	0x32	58	ENC1_FILTER	0x72
27	PDA_REF_VAL	0x34	59	ENC2_CTRL	0x74
28	PDA_REF_POS_L	0x36	60	ENC2_POS_L	0x76
29	PDA_REF_POS_H	0x38	61	ENC2_POS_H	0x78
30	PDA_MAX_VAL	0x3A	62	ENC2_CAPT_L	0x7A
31	PDA_MAX_POS_L	0x3C	63	ENC2_CAPT_H	0x7C
32	PDA_MAX_POS_H	0x3E	64	ENC2_FILTER	0x7E

3. Starting the work with the device

OPBOX-2.1 is powered from USB interface and is conform with USB2.0 standard. According to its requirements there are some limits, concerning power supply (max. start current, max. continuous power). To fulfill this standard requirements OPBOX 2.1 has three independent power supply branches (analogue part, 12V part and ADC/DAC converters part). Each branch has independent signals, switching it and showing error conditions.
After connecting the device to the USB port, all three branches are switched off and only digital part, responsible for communication is working. Before starting to work with the device (triggering acquisition) the user application must switch on the power supply of all remaining branches.
Switching on sequence:

1. Setting bit [0] „Power Enable" in POWER_CTRL register;
2. Checking bit [4] „Power OK" in reg. POWER_CTRL until the value '1' will occur.

If the flag „Power OK" is set, the device is ready to work.
1. In each moment the user can check status flags of all power supply branches: „ANALOG PWR Status", „DC12V Status" and „VREG Status" in POWER_CTRL register.
Switching off analogue branch of the device causes loss of DAC converter settings, thus after each switching on sequence it is necessary to actualize some device settings:

1. PULSE_AMPLITUDE (commend: (0xD6));
2. Amplifier setting – register CONST_GAIN (Address: 0x28).

If the device cannot switch on the power in a correct way (the flag „power ON" is not set during few seconds) it is necessary to control the following:

1. USB cable – if it is not damaged, of poor quality or too long;
2. The cable should be replaced, the condition of the device controlled again;
3. DB15 connector should be disconnected, if used and the device condition should be controlled again;
4. The device should be connected to another computer and checked again.

During data acquisition the device automatically block the measurement command, if any of status flags signals a failure. In the case if the voltage of USB power supply falls significantly during 100ms, the device automatically restarts the power supply sequence, switching off all power supply branches and than switching them on again. During this time all data acquisitions are blocked.
In the header of each acquisition status byte TriggerOverrunSource is written (byte 8) – it contains the information about the source of lost triggers since last completed till actual acquisition. If during this time one or more triggers couldn't be executed due to the failure or overload of power supply, the flag „TrgOvrScr" will be set to „1".

4. Introduction to data acquisition settings.

The data acquisition system in OPBOX 2.1 (firmware rev. 2.1.60) was designed with the goal to reach a real time acquisition speed (PRF) of up to 10000 measurements per second. The device stores the data in internal memory and allows the software to read the data in packets. This allows to reach high medium speed of measurement data for different sizes of measurement windows and limits the necessity to communicate with the device on the level of control functions (inquires, confirmations, readings of results from measurement gates, encoder positions, etc.)

4.1. Basic features of acquisition in FIFO mode:

- Allows to make the acquisition in real time, until the buffer is full, without the need to read the data. This allows to reach high speed of measurements repetition (PRF).
- Large hardware FIFO buffer (256kB) ensures stable PRF and allows to reach the independence from the fluctuation of data speed on the USB interface;
- System controller ensures the priority of acquisition trigger, interrupting the reading during the acquisition time and starting it again, after the measurement is made.
- Each measurement cycles creates in memory a data frame, where first 54 bytes are a

header and remaining part is created by the signal data coming from the ADC;
- The header contains actual information about the given measurement, such as: measurement index, encoder position in the moment of trigger, the results of hardware peak and transition detectors, etc.
- The settings of signal acquisition mechanism allow to adapt flexible buffering according to actual needs of application and hardware possibilities with the goal to reach the required measurement speed in the real time.
- The acquisition mode without storage of signal data („store disabled"), that delivers only the header with results from peak detectors, encoder positions, etc. increases the acquisition speed.

5. Meanings of basic acquisition elements

The description of acquisition in FIFO more requires the definition of some basic elements and assumptions. The list of them is as follow:

5.1. Acquisition

It is a single measuring sequence, having following elements: sending sequence, delay until measurement starts, measurement and creation of a header (register data and measurements results). Acquisition settings include among other: sampling frequency, delay time, size of the measurement window, selection of analogue input, input attenuation, analog filters settings.

5.2. Measurement Window

It defines the amount of samples of analogue signal, obtained from ADC converter, stored in the memory during one acquisition. The amount of measurement data is defined by the parameter DEPTH (register DEPTH_L and DEPTH_H). DEPTH can have a value of 1 to 262090 samples.

5.3. StoreDisabled

The acquisition mode „StoreDisabled" (switched on with bit [9] storeDisable in register MEASURE) causes that data from ADC converter are not stored to the memory of the device. Only acquisition header is stored. This mode can be used to speed up the measurement (lower amount of data to transfer), when measurement data are not necessary (i.e. only the results from peak detectors are needed).

5.4. Header

It has always 54 bytes (constant HEADER_SIZE) and contains the information about each acquisition. Header includes for example acquisition index and information fetched during the trigger moment: encoder positions, state of GPI line, hardware timer and the results from hardware peak detectors. Header format is described in chapter 6.

5.5. Frame

Acquisition frame (short: frame) contains data from one acquisition. The frame is created after each trigger event and has following elements: Header and measurement data. The amount of bytes in the frame is a sum of the length of the header and the amount of measurement data. - FRAME_SIZE [B] = HEADER_SIZE [B] + DEPTH [samples].

5.6. Packet

Acquisition packet (short: packet) is the amount of acquisitions, that the device stores in the internal buffer, until it sets the flag PACKET_READY. The amount of acquisitions in packet defines the register/parameter PACKET_LENGTH. Maximal value of this parameter depends on the amount of measured data in acquisition. PACKET_LENGTH = 1 means the work without the buffer and the device provides the data after each acquisition.

5.7. Acquisition in real time

The meaning of this term, used in this document is as an continuous sequence of acquisitions without changing the device settings, mostly triggered by external signals, changing in the real time, without the control of the software (triggering with hardware timer, encoder positions, etc.)

Elements defined above are shown on the following picture.

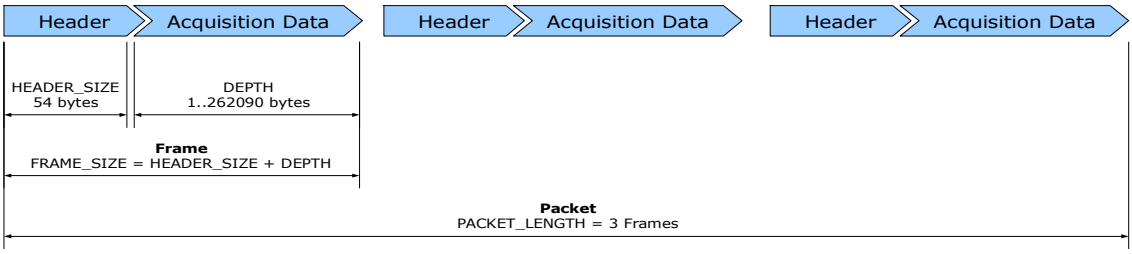

Figure 5.1: Data organization in Header and Packets

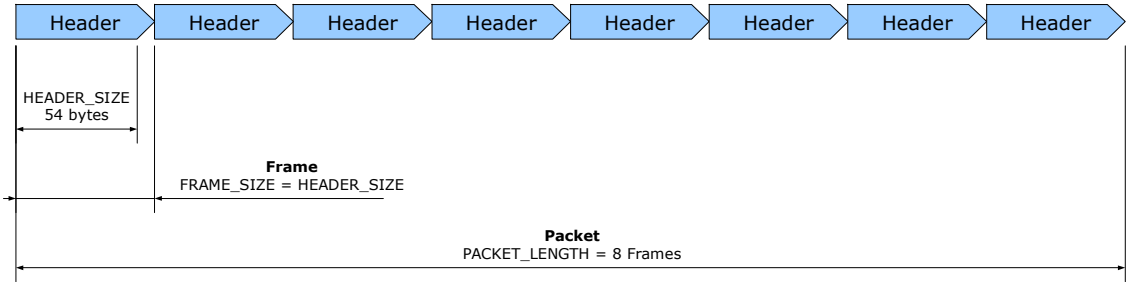

Figure 5.2: Data organization in Header and Packets in „StoreDisabled" mode

6. Header of Acquisition Frame

Acquisition header has 54 bytes, described in details with the following table:

Byte No	Name	Register address	Value range	Description
1	**Start of Frame**	---	**'@' (0x40)**	Beginning character of frame/header ASCII „@”
2	FrameIdx	0x06	[7:0]	Actual value of frame counter – 16-bit counter with automatic reset at overflow
3			[15:8]	
4	TimeStamp	0x18	[7:0]	Register TimeStamp – value of TIMER register captured on trigger event
5			[15:8]	
6	TriggerOverrun	0x12	[7:0]	Register TriggerOverrun – amount of lost triggers since last acquisition (16-bit counter)
7			[15:8]	
8	TriggerOverrunSource	0x0A	[3:0]	Flags of lost triggers sources since last acquisition
9	GPI Captured		[5:0]	State of GPI[5:0] captured on trigger event
10	Encoder 1 Position	0x6E	[7:0]	Encoder 1 position captured on trigger event
11			[15:8]	
12		0x70	[23:16]	
13			[31:24]	
14	Encoder 2 Position	0x7A	[7:0]	Encoder 2 position captured on trigger event
15			[15:8]	
16		0x7C	[23:16]	
17			[31:24]	
18	Peak Detectors Status	0x2A	[7:0]	PeakDetectors control/status register
19			(0x00)	- reserved
20	PDA RefPos	0x36	[7:0]	PDA gate – reference level crossing position
21			[15:8]	- useful bits [17:0]
22		0x38	[23:16]	
23			(0x00)	- reserved
24	PDA MaxVal	0x3A	[7:0]	PDA gate – maximum value of signal in the gate
25			(0x00)	- reserved
26	PDA MaxPos	0x3C	[7:0]	PDA gate – position of maximum value
27			[15:8]	- useful bits [17:0]
28		0x3E	[23:16]	
29			(0x00)	- reserved
30	PDB RefPos	0x4A	[7:0]	PDB gate – reference level crossing position
31			[15:8]	- useful bits [17:0]
32		0x4C	[23:16]	
33			(0x00)	- reserved
34	PDB MaxVal	0x4E	[7:0]	PDB gate – maximum value of signal in the gate
35			(0x00)	- reserved
36	PDB MaxPos	0x50	[7:0]	PDB gate – position of maximum value
37			[15:8]	- useful [17:0]
38		0x52	[17:16]	
39			(0x00)	- reserved
40	PDC RefPos	0x5E	[7:0]	PDC gate – reference level crossing position
41			[15:8]	- useful bits [17:0]
42		0x60	[17:16]	
43			(0x00)	- reserved
44	PDC MaxVal	0x62	[7:0]	PDC gate – maximum value of signal in the gate
45			(0x00)	- reserved
46	PDC MaxPos	0x64	[7:0]	PDC gate – position of maximum value
47			[15:8]	- useful bits [17:0]
48		0x66	[17:16]	
49			(0x00)	- reserved
50	DataCount	0x24	[7:0]	Amount of measurement data (parameter DEPTH)
51			[15:8]	- useful bits [17:0]
52		0x26	[17:16]	
53			(0x00)	- reserved
54	**End of Header**	**---**	**'/' (0x2F)**	Header end character – ASCII „/”
55	Sample 1		[7:0]	First measurement sample
56	Sample 2		[7:0]	Second measurement sample
	[...]			
54+ (Cnt-1)	Sample [DataCount-1]		[7:0]	
54+Cnt	Sample [DataCount]		[7:0]	Last measurement sample

7. Configuration and maintenance of acquisition in real time

Configuration and maintenance of acquisition in real time should be made according to the following schematic:

1. Stop of acquisition, that is going on: blocking all trigger sources (bit TriggerEnable in TRIGGER register).
2. Setting of acquisition parameters:

   - Sender configuration – loading time and voltage level (register PULSE_TIME and PULSE_AMPLITUDE);

   - Analogue channel configuration – choice of input, attenuator, low and high pass filters, amplification (register ANALOG_CTRL, MEASURE, CONST_GAIN);

   - configuration of signal acquisition parameters – delay of measuring window (DELAY), sampling frequency (MEASURE), size of measuring window (DEPTH) and amount of frames in buffer (PACKET_LENGTH);

   - configuration of remaining modules depending on needs – setting of measurement gates PDA, PDB, PDC, encoders ENC1 and ENC2 etc.;

   - configuration of trigger (TRIGGER) – choice of trigger source (TRIGGER register), configuration of block, used for triggering (TIMER, XY_DIVIDER, Encoders).
3. Unblocking of trigger (bit of trigger blocking TriggerEnable in TRIGGER register) – the device is ready to make the acquisition and is waiting for trigger.
4. Receiving acquisition data:

   - the device will set the flag PACKET_READY after the required amount of acquisitions is made (PACKET_LENGTH), trying to read data before will cause failure (reading error, timeout);

   - the time needed to fulfill all acquisitions depends mostly on a chosen trigger source and the size of Measurement Window;

   - the actual amount of made and stored acquisitions can be checked reading the register (FRAME_CNT)

   - if packet is available (PACKET_READY=1), it is possible to read the whole Acquisition Packet;

   - the amount of bytes, that could be read depends on acquisition parameters:
     PACKET_SIZE = (HEADER_SIZE+DEPTH)*PACKET_LENGTH [bytes];

   or in the mode without storing measurement data (StoreDisabled):

     PACKET_SIZE = (HEADER_SIZE)*PACKET_LENGTH [bytes];

   - after the data was received it is not necessary to send a confirmation, the device is automatically freeing the memory.
5. Blocking the acquisition – receiving the remaining frames

   After stopping the acquisition in a real time (blocked trigger source) it can happen, that the buffer of the device is filled with some acquisition data. In such a situation remaining frames can be ignored by resetting the memory controller. This is possible in some ways:

   - sending the command FIFO_RESET;

   - changing the length of measuring window (writing to DEPTH register);

   - increasing the length of the Packet (writing to PACKET_LEN register a value, that is larger than actual value, writing a smaller value do not cause the reset – details will follow).
6. The procedure or reading the remaining frames from the memory is as follow:

   - blocking the triggering of acquisition (TRIGGER register);

- receiving all packets until the moment as the flag PACKET_READY will have zero value.

- the device buffer can contain a fragmentary acquisition packet (not the whole PACKET_LEN), the amount of remaining frames should be read from the register FRAME_CNT;

- if FRAME_CNT = 0 it means, that all frames was received and the reading was correctly finished.

- if FRAME_CNT is not equal zero (i.e N) the setting PACKET_LEN should be changed to exactly the amount of frames, that is remaining in the buffer of the device (N should be written to the register PACKET_LEN). Only in this situation the change of PACKET_LEN register to the smaller one do not reset the frame buffer in the device. After changing the PACKET_LEN setting the device will allow to read N frames, changing the flag PACKET_READY to '1';

- to confirm it, it is possible to check the flag PACKET_READY and if it is equal '1' it is possible to read the new acquisition packet (N frames);

- after reading the packet with new length, the acquisition sequence was finished with correct reading of all data. To prepare the next acquisition the PACKET_LEN should be changed to the previous value or new one.

8. Comments to the acquisition settings.

8.1. Calculating maximal packet length

Hardware acquisition buffer allows to store maximally 256kB samples (single measurement data – exactly 262144). Maximal value of packet length (PACKET_LEN), means the amount of full frames, that the device can store in internal memory depends on the length of measurement window (DEPTH) and can be calculated from following formulas:

PACKET_LEN_MAX = INTEGER PART[262144 / FRAME_SIZE] =
= INTEGER PART[262144 / (DEPTH + 54)]

In the mode without writing measurement data (StoreDisabled):

PACKET_LEN_MAX = INTEGER PART[262144 / HEADER_SIZE] = **4854**

Example:

For measurement window 10us @ 100MHz (DEPTH = 1000 samples):

PACKET_LEN_MAX = INT. PART[262144 / (1000+54)] ≈ INT. PART[248,71] = **248**

8.2. Packet length and PRF

The choice of optimal setting of PACKET_LEN depends on some factors:

- required PRF;
- required repeatability and fluency of data readings from following acquisitions;
- duration of each acquisition (depend on DEPTH and DELAY);
- PC parameters (size of RAM, type of processor, type of USB Host);
- actual load on USB (amount of devices connected to USB);
- other (operational system, DLL libraries).

Factors influencing throughput of data from the device are:

- frequent calling of control function (including checking the flag PACKET_READY);
- large amount of reading data packets;
- not optimal construction of checking PACKET_READY loop.

Additionally, USB interface is not guarantying fixed bandwidth for Bulk transfers, which is used by OPBOX, but it is guarantying the correct delivery of all data (confirmation, check sum). For Bulk data transfers USB Host is offering only the time, that is not used by other transfers on the bus, and this is causing that the transfer speed can vary. To achieve large throughput of measurement data from the device to the PC one should try to increase the length of the packet, that allows to diminish the amount of calls of data check function (PACKET_READY) and to make the readings of large amount of data at one time.

Available throughput in the acquisition mode in real time is about 10...15 MB/s. This value can be used for estimation of maximal available PRF that can be reached, depending on the length of chosen length of measurement window (DEPTH) dividing for example 10MB/s (10485760 B/s) through the frame length (HEADER_SIZE+DEPTH).

Experimental measurements of throughput have shown, that it increases strongly if the size packet reaches about 8kB and can be about 8MB/s. Further, even large increase of packet size causes only small increase of maximal throughput. Packet size of about 8kB can be treat as a compromise between the throughput and fluency of data transfer.

8.3. Changing the acquisition settings

During making changes of acquisition settings one must always remember, that the buffer can contain some amount of acquisitions made with older settings. After changing the settings, acquisitions made according to new settings will be received after older acquisitions are read, that can cause analysis errors, delayed reaction, etc.

In the case of changing such parameters as filters settings, amplification, delay the result can be a delay in introducing such changes. But in the case of change of Measurement window length (DEPTH) and packet length (PACKET_LENGTH) it could cause wrong data reading (reading of different length of Frame/Packet as stored in the buffer). This is the reason, why the memory controller in OPBOX-2.1 automatically detects and reacts to such situations, to avoid reading errors during changes of critical parameter settings. The description of behavior of OPBOX-2.1 during changes of DEPTH and PACKET_LENGTH is described below:

8.3.1. Change of the measurement window setting (DEPTH)

Changing of DEPTH setting causes that it is necessary to calculate new parameters of acquisition buffering (PACKET_LEN_MAX, placing of frames in relation to memory of TGC curve, etc.). Changing DEPTH parameter automatically changes the packet length too. **For this reason each change of the length of measurement window (writing to DEPTH register) causes the reset of acquisition buffer.** All acquisitions that are stored in the buffer are removed.
To avoid the loss of data stored in the acquisition buffer it is recommended to change the DEPTH setting according to the schematic below:
1.Stopping the acquisition by blocking the trigger (i.e. using the bit „Trigger Enable" in TRIGGER register);
2. Receive all frames from the buffer until the flag PACKET_READY = 0 (no full packet in the buffer);
3. Check if the buffer do not contain partial acquisition packets (read the register FRAME_CNT);
4. If FRAME_CNT is different than zero, it is required to set the length of packet to the value read from FRAME_CNT;
5. Read the last partial acquisition packet;
6. Change DEPTH setting to the required value (at this point it is possible to change also other parameters of acquisition – if this is necessary, among other also the length of acquisition packet, that was changed before);
7. Unblock triggering (TRIGGER register).

During writing new settings to DEPTH registers, the device checks automatically if the new packet length is not larger than the size of acquisition buffer (256kB). New length of the packet is calculated based on the new DEPTH setting and actual PACKET_LEN value. If this is larger than 256kB the device **automatically changes the setting of PACKET_LEN to PACKET_LEN_MAX** – maximal possible with new DEPTH setting.

8.3.2. Changing of the length of acquisition packet (PACKET_LEN)

Changing the length of acquisition packet, similar as in the case of DEPTH parameter, requires the actualization of many buffering parameters – **this is the reason, why the change of PACKET_LEN is causing the reset of acquisition buffer.** It is not happening only in cases, when following conditions are fulfilled:

- buffer contains partial acquisition packet (flag PACKET_READY is equal zero and the amount of frames stored in the buffer is lower than PACKET_LEN);
- new value of PACKET_LEN is **lower** than actual.

If any of above conditions is not fulfilled, the change of the PACKET_LEN parameter causes automatic reset of the device buffer.
Above exception is used to stop the acquisition in real time with receiving all data stored in acquisition buffer. During writing to PACKET_LEN register the device is automatically checking the value written. If it is zero, it is changing the value to minimum possible (PACKET_LEN=1). If written value is larger than maximal possible with actual DEPTH setting, the device is automatically limiting the setting of PACKET_LEN to PACKET_LEN_MAX.

8.3.3. Change of the remaining acquisition settings

Change of other than above described settings (i.e. Input channel, amplification, filter) can be made without stopping the acquisition, but it can cause delay of „reaction to change", caused by fact, that the buffer can contain many acquisitions made with earlier settings, that will be read first. If such behavior do not cause wrong behavior of data analysis algorithms, changes of of acquisition parameters can be made „on fly" without stopping the acquisition in the real time.

8.4. Blocking of acquisition triggering

OPBOX cannot make the acquisition although trigger signal is present in following cases:

a) the acquisition in progress
   In this case the new acquisition will be not started, and trigger impulse will be counted by the module of lost triggers counter (TrgOverrun) and the bit TrgOvrScr_A in CAPT_REG will be set. The value of this counter is written to the acquisition header immediately after it is finished. This value shows the amount of triggers that was lost since the previous acquisition was finished – during the actual acquisition.
b) the frequency of trigger pulses is larger than 10kHz (their period is smaller than 100μs)
   OPBOX-2.1 has internal module limiting maximal repetition speed to 10kHz (trigger period 100μs). Trigger pulses coming quicker than 100μs since previous trigger are ignored and their amount is counted by TrgOverrun module and bit TrgOvrScr_H in CAPT_REG is set);
c) the memory of internal buffer is full:
   If the buffer has no place to store the next acquisition frame, all trigger pulses are counted by the TrgOverrun counter and the TrgOvrScr_F in register CAPT_REG is set). This register are written to the header after the next acquisition is made (after freeing the place in the buffer);

d) the power supply control system has informed about the power supply failure
   OPBOX-2.1 has a power control system fulfilling the needs of USB-2.0 interface specification and to secure the system for the case of overload of power supply. POWER_CTRL register contains the description of status flags of the power supply module. If on trigger event any of this flags shows error, the acquisition is not made and the pulse is counted by the TrgOverrun counter and the bit TrgOvrScr_P in CAPT_REG is set). It prevents acquisitions with potentially erroneous or unstable settings (for example voltage level).

After receiving the frame, it is possible to check in the header the amount of lost triggers since the start of previous acquisition and the reason for this event (CAPT_REG register). Each of flag TrgOvrScr_A, TrgOvrScr_H, TrgOvrScr_F and TrgOvrScr_P in CAPT_REG register informs about the fact, if at least one trigger event took place, that was missed for some concrete reason. This flags can be helpful in quick diagnosis of problems with the device, measurement speed, etc. It is possible to set more than one flag. Reasons for loosing trigger events could overlap, for example in the case, if the next trigger event occurrs during on going acquisition and earlier than 100μs after its beginning both flags TrgOvrScr_A and TrgOvrScr_H will be set.

8.5. Resetting of Acquisition buffer

Acquisition buffer of the device can be reset at each moment with the help of the **DIRECT_RESET_FIFO** (0xD2) command. This command causes that all acquisitions will be removed from the buffer. **Resetting acquisition buffer do not reset the acquisition settings!** Such parameters as PACKET_LENGTH, DEPTH remaining unchanged. Resetting all settings is possible with the global reset command: DIRECT_RESET (0xD1)). Global reset resets acquisition buffer too.

8.6. Time gain compensation curve (TGC)

Configuring the device to work with the generation of arbitrary TGC curve during acquisition in FIFO mode requires appropriate preparation of the table of TGC curves with the size, that is equal the size of the buffer for this curves (TGC memory). The size of this buffer is equal the size of the acquisition buffer (256kB)
The table of TGC curves must include PACKET_LEN_MAX amount of copies of a single amplification curve with the length DEPTH bytes, that must be placed in the whole table with the distance of FRAME_SIZE bytes and shifted by HEADER_SIZE=54 bytes in the relation to the begin of each frame. First 54 bytes should be written with the first value of amplification from the TGC curve.
The detailed structure of the TGC curves table is shown on the following diagram:

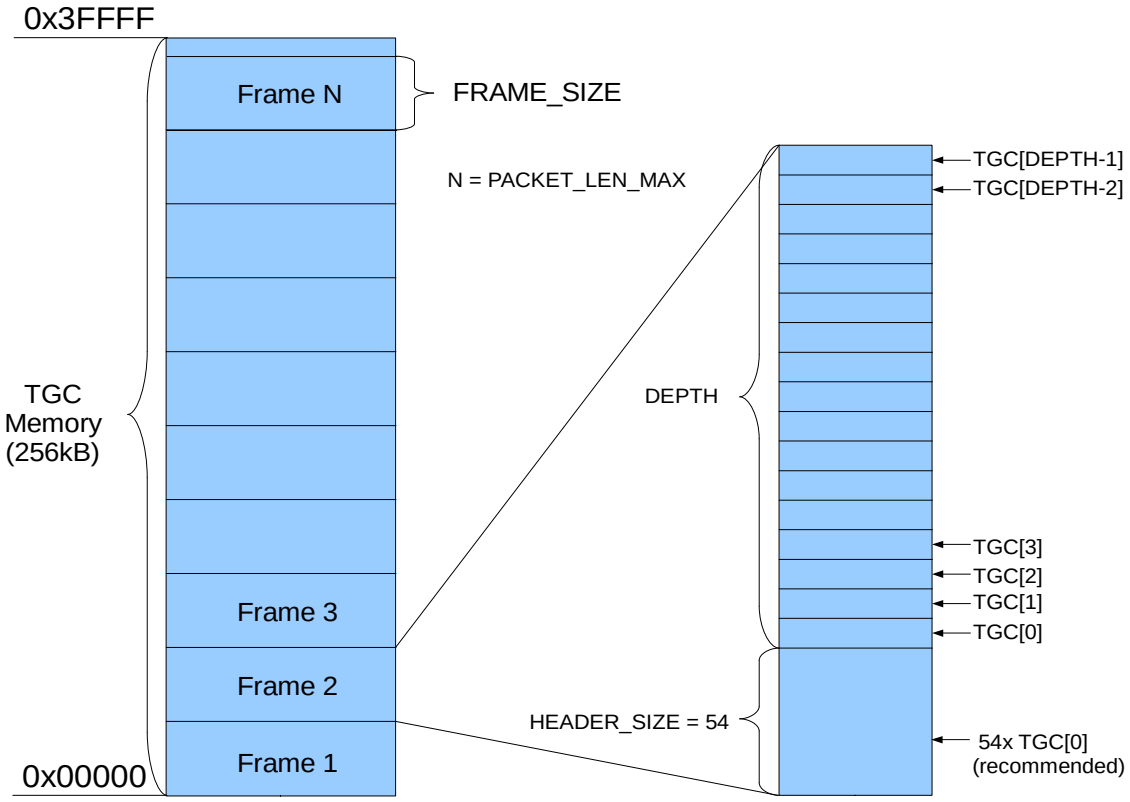

Figure 8.1: Organization of TGC curves in the Table of Amplification Curves

The sending of amplification curves to the device should always include the transfer of the whole 256kB table of data, needed to write the whole TGC curves buffer. Partial writing of data to this buffer can cause wrong function of TGC function during some acquisitions and cause problem with following writing curves to the memory (shift of hardware data pointer, etc.). If necessary it is possible to make a reset of TGC curves controller using the commands FIFO_RESET (0xD2) or RESET (0xD1).
Not correctly prepared table of TGC curves causes wrong function of acquisition – each acquisition can be made with different (not correct or time shifted) time gain curve as the previous one.
During transmission of the TGC curves table to the device it is necessary to block the triggering of acquisitions. It is recommended to use for this purpose the zeroing of bit [4] Trigger Enable in TRIGGER register (address: 0x10) according to the following schedule:

- blocking the trigger events (zeroing bit „Trigger Enable" in TRIGGER register);
- sending the TGC curves table (256kB) to the device;
- unblocking the device.

If the device during the transfer of the TGC curves table is not blocked, it could cause the triggering of acquisition during memory storage of TGC curves and cause errors in the acquisition data.

9. Maintenance of Peak Detectors modules

OPBOX 2.1 has three hardware peak and level detector modules – gates PDA, PDB i PDC. They have following parameters:

- three independent gates with peak detectors and level comparators;
- configurable start and stop point for each gate;
- configurable level for level detector (comparator);
- configurable function modes: „Level", „Rising", „Falling" and „Transition";
- information about the moment of level transition;
- information about the position and level of maximal signal value in the gate.

Gates PDA, PDB and PDC must be configured in a following way:

- activate the gate by the setting of bit „PDx enable" in PEAKDET_CTRL register;
- set the required comparator mode of – bits „PDx_mode"
- write the Start and Stop position of gates – register PDx_START_L (_H) and PDx_STOP_L (_H);
- write the level of comparator activation – register PDx_REF_VAL.

Measurement gates comparator can be configured in one of the following four modes:

1. Mode „Level" - detecting signal level – first sample has a value that is higher or equal to PDx_REF_VAL causes the activation of comparator and writing the position of this sample in the register PDx_REF_POS_L (and _H);
2. Mode „Rising" - detects rising slope – first event occurs when one sample is lower than PDx_REF_VAL and the next sample is higher or equal this level causes the activation of a comparator and writing the level of the second sample;
3. Mode „Falling" - detection of the falling slope – first event occurs when one sample is higher than PDx_REF_VAL and the next sample is lower or equal this level causes the activation of a comparator and writing the level of the second sample;
4. Mode „Transition" - detecting the level transition – first event „Rising" or „Falling" causes the activation of comparator and writing the position of the second sample;

After configuring and unblocking the gates are working parallel with each acquisition analyzing data from ADC after hardware processing (RF or absolute representation). After the acquisition is finished the results of work of each gate are written together with measurement data in the acquisition header. After each acquisition the results of the gates are automatically reset and the gates are prepared for the next acquisition. Measurement gates PDA, PDB i PDC are working also in the mode without storing measurement data to the memory (mode „Store disabled").

10. Maintenance of the incremental encoders modules

OPBOX 2.1 has two encoder counters modules ENC1 and ENC2 that are independent able to work with two incremental encoders having quadrature inputs CHA and CHB and index input IDX.
Parameters of encoder modules:

- two independent position counters;
- configurable work modes, working with one or all encoder inputs;
- large position counter – 32-bits;
- catching of position counter value in the moment of acquisition trigger event;
- triggering the acquisition on defined positions;
- storing the position trigger in each acquisition header.

Configuration of encoder modules:

1. Disconnect USB cable from the device;
2. Connect encoder using CONTROL connector according to the table below;
3. Connect USB cable to the device;
4. Connect the encoder power supply (external);
5. Set the configuration bits of the module: work mode, negation, input filter in the ENCx_CTRL register of chosen encoder module;
6. Set bit [0] „Encoder X Enable” in ENCx_CTRL register;

Each encoder module has the possibility to trigger acquisition on defined positions. More information in the description of the control register of the device.
Modules ENC1 i ENC2 have configurable digital noise filters. The user must configure them according to the expected frequency of the encoder signal.
Encoder modules ENC1 and ENC2 are using digital inputs GPI[5:0]. All this inputs are working in 5V TTL standard and have 10kOm resistors parallel to the device ground.
The user must ensure, that the outputs of the encoder used are compatible to the inputs of OPBOX 2.1.

List of encoder modules inputs :

Description	Encoder 1 (ENC1)	Encoder 2 (ENC2)
CHA	GPI0 (pin 11)	GPI3 (pin 5)
CHB	GPI1 (pin 4)	GPI4 (pin 15)
IDX	GPI2 (pin 12)	GPI5 (pin 8)