Description of OPCARD – 01/100

Single Channel PCI-bus Ultrasonic PR Card
Ultrasonic card with integrated pulser and receiver

OPCARD -01/100 is a complete ultrasonic testing device, suitable for all ultrasonic measurements, and due to many additional inputs and outputs and internal processor, it can be used as a controller for more complicated devices.

The card can work together with following devices:

- 32 channel multiplexer – the card can control it directly;
- Scanner – it has input for incremental encoders;

The card has an implemented one channel pulser & receiver and can be used with one transducer or with two (one is sending and the second receiving).

OPCARD -01/100 is a short PCI card and can be installed in a standard PC.
Technical data

A/D converter:
- Resolution: 8 bits
- Sampling frequency: 12.5, 25, 50 or 100MHz

Analog parameters:
- Input amplifier gain: -20dB - 76dB
- Input attenuator: off or -20dB
- Sensitivity: 0.1 mV - 1Vpp / Div
- Bandwidth: 0.5 MHz - 25 MHz (-3dB)
- Input impedance: 50 Ohm, 10pF
- Filter switchable: .5 – 6MHz, .5 – 10MHz, .5 – 15MHz, .5 – 25MHz

Pulser:
- Pulse Voltage: from off (0V) to 360V
  (positive pulse, short circuit step pulser)
- Fall Time: <= 20 ns
- Pulse Duration: short circuit, bandwidth up to about 50MHz
- Output impedance: < 1 Ohm

Delay time:
Post trigger 256us, measurement accuracy better than 1ns

Data buffer:
256, 512, 1k, 2k, 4k, 8k, 16k, 32k, 64k, 128k samples

DAC (TGC) with arbitrary waveform generator:
- Resolution of time step: 20 ns
- Resolution: 8 bit
- Max. Gain changing: 48dB pro step

Counters / Input for incremental encoder:
- 2 channel, 16-bit

Trigger
- Internal trigger rate: Software
- External trigger rate: TTL Signal
- Multiplexer control: Software
- Trigger output signal: TTL Signal

Additional inputs:
- 2 x digital line: TTL Signal

Output:
- 5 x digital line: TTL Signal

Data bus:
PCI (short card)

Transfer:
Bus Mastering DMA data transfer

1 - software selected
**Signals on the external connectors**

**LEMO:**
- **PE** - Measured input signal (sending/receiving);
- **TT** - Measured input signal (receiving);

**DB15:**
- Pin1, 9 - TTL encoder input signals
- Pin2, 10 - TTL general input signals
- Pin4 - TTL trigger out signal
- Pin5, 6, 7, 12, 13 - multiplexer channels control lines
- Pin8 - +12V
- Pin15 - -12V
- Pin14 - GND

**Caution:**

Upper Lemo connector is connected with the pulser. The maximum voltage of the pulse on this connector can reach about 360 Volts. Due to the short duration it is not dangerous to the human, but can destroy electronic circuits or transducers if not used carefully.

**Signal from Multiplexer must be connect to the TT connector.**

**Characteristic of the card:**

One of the most important features of the card is a precise synchronization between the transmitter trigger signal postTrigger and the moment when the sampling of the input signal starts.

This time (postTrigger) is software programmable in the range of 0 – 255us with a resolution of 1us and has stability within the range of 1ns. It is particularly important in...
the case of scanning devices, since it allows to achieve a very small time skew between different positions (channels). 1ns would correspond to a clock frequency of 1GHz which is much higher than the actual frequency used.

**Characteristic of the pulser on board:**

![Pulser circuit waveforms](image)

Pulser circuit waveforms: The rising edge of the Trigger signal (described as Trigger) initiates the transducer charging process which takes about 3us. After this time a transistor switch which discharges the transducer is turned on (discharge time is about 20ns, but it can be longer, if the capacitance of the transducer is too large; the limit is reached, if the transducer is made from the standard ceramics, is 0.1 mm thick and has 8 mm diameter).

Because of the very low output impedance of the device (<1 Ohm) and short discharge time the pulse generated with this device could be concerned as a real pulse answer for the most transducers. Transducers with a parallel matching inductance should not be used, since they do not allow the transducer to be pre-charged (the inductance causes a short-circuit).

**Multiplexer OPCOM-01 (32,16,8 or 4 channels)**

OPCOM-01/100 is particularly well suited for ultrasonic measurements as well as other kinds of measurements which need many channels. Together with the card OPCARD and ultrasonic probes it could be used as complete ultrasonic testing device. Each channel has separate pulser & receiver.

![Picture 1. OPCOM-01/8 channels](image)
Additional caution:
For OPCOM-01/4 channels:  channel 1 has address 8 and 16
channel 2 has address 9 and 17
channel 3 has address 10 and 18
channel 4 has address 11 and 19

For OPCOM-01/8 channels: channel 1 has address 8 and 16
channel 2 has address 9 and 17
channel 3 has address 10 and 18
channel 4 has address 11 and 19
channel 5 has address 12 and 20
channel 6 has address 13 and 21
channel 7 has address 14 and 22
channel 8 has address 15 and 23

Technical data

Channels: 32, 16, 8, or 4
Pulse amplitude range: 50-360V
Max cable length between computer and the box: 30m

Connectors
BNC:
Transducers Trigger input
Signal output

DB15:
Control signals/Power supply

Features and Specifications

Pulser circuit waveforms: The rising edge of the Trig In signal initiates the transducer charging process which takes about 3us. After this time a transistor switch which discharges the transducer is turned on (discharge time is about 20ns, but it can be longer, if the capacitance of the transducer is too large; the limit is reached, if the transducer is made from the standard ceramics, is 0.1 mm thick and has 8 mm diameter). Because of the very low output impedance of the device (<1 Ohm) and short discharge time the pulse generated with this device could be concerned as a real pulse answer for the most transducers.

Software Package

The software (for Windows 95,98/NT 4.0/2000/XP) includes all necessary functions for ultrasonic measurements.
Software Package – how to install and use the software, delivered with the OPCARD

1. Please put the card into appropriate PCI slot.
2. BIOS of the PC should watch the card with following data: Vendor ID 88BB, Device ID 0000 and Class "docking station".
3. The card is Plug-and-Play, Windows should be able to detect OPCARD as “docking station”, and the Add New Hardware Wizard should open automatically as soon as you start the system. Follow the onscreen instructions for the wizard. When you are prompted to select a driver for this device, browse to the DRIVER folder and select the INF file siwdm.inf
4. After restart of system you are to install software in directory "<drive>:\OPCARD" running "setup.exe" from CDROM\INSTALL directory.

OPCARD Card is plug-and-play compatible with Windows 98 and beyond (our recommend Windows 2000 or XP)

Control Panel -> Hardware Manager.

Standard packed software includes:
Driver:
   siwdm.inf
   siwdm.sys
Libraries:
   samplib.h
   samplib.dll
   samplib.lib
**Description of control function OPCARD**

List of functions includes on `samplib.dll`

- `Sample_ReleaseBaddr` Get virtual address for OPCARD
- `Sample_SetTimeout` Set Timeout for DMA channel
- `Sample_GetAddr` Get physical address for DMA channel
- `Sample_InitMeas` Initialize measurement
- `Sample.GetData` Get Sample
- `Sample_SendTGCArray` Send TCG Curve
- `Sample_InitTGC` Initialize TCG curve length
- `Sample_Set20dB` Set attenuator feature
- `Sample_SetDepth` Set length of buffer
- `Sample_SetDelay` Set post trigger
- `Sample_SetFreqTGC` Set TCG curve frequency sampling
- `Sample_SetFreq` Set sampling frequency
- `Sample_SetPulser` Set amplitude of pulser
- `Sample_SetSource` Choose TT or PE mode
- `Sample_SetFiltr` Set Filter feature
- `Sample_SetChannel` Set Multiplexer Channel
- `Sample_SetCounterA` Set divider for external trigger channel A
- `Sample_SetCounterB` Set divider for external trigger channel B
- `Sample_SetCounterAActive` Set active external trigger channel A
- `Sample_SetCounterBActive` Set active external trigger channel B
- `Sample_SetTrigger` Set Trigger Mode
- `Sample_SetSampleTGC` Set TCG curve length of buffer
- `Sample_SetSingleGain` Set constant gain

### Basic control functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample_ReleaseBaddr</td>
<td>Get virtual address for OPCARD</td>
</tr>
<tr>
<td>Sample_SetTimeout</td>
<td>Set Timeout for DMA channel</td>
</tr>
<tr>
<td>Sample_GetAddr</td>
<td>Get physical address for DMA channel</td>
</tr>
<tr>
<td>Sample_InitMeas</td>
<td>Initialize measurement</td>
</tr>
<tr>
<td>SampleGetData</td>
<td>Get Sample</td>
</tr>
<tr>
<td>Sample_SendTGCArray</td>
<td>Send TCG Curve</td>
</tr>
<tr>
<td>Sample_InitTGC</td>
<td>Initialize TCG curve length</td>
</tr>
<tr>
<td>Sample_Set20dB</td>
<td>Set attenuator feature</td>
</tr>
<tr>
<td>Sample_SetDepth</td>
<td>Set length of buffer</td>
</tr>
<tr>
<td>Sample_SetDelay</td>
<td>Set post trigger</td>
</tr>
<tr>
<td>Sample_SetFreqTGC</td>
<td>Set TCG curve frequency sampling</td>
</tr>
<tr>
<td>Sample_SetFreq</td>
<td>Set sampling frequency</td>
</tr>
<tr>
<td>Sample_SetPulser</td>
<td>Set amplitude of pulser</td>
</tr>
<tr>
<td>Sample_SetSource</td>
<td>Choose TT or PE mode</td>
</tr>
<tr>
<td>Sample_SetFiltr</td>
<td>Set Filter feature</td>
</tr>
<tr>
<td>Sample_SetChannel</td>
<td>Set Multiplexer Channel</td>
</tr>
<tr>
<td>Sample_SetCounterA</td>
<td>Set divider for external trigger channel A</td>
</tr>
<tr>
<td>Sample_SetCounterB</td>
<td>Set divider for external trigger channel B</td>
</tr>
<tr>
<td>Sample_SetCounterAActive</td>
<td>Set active external trigger channel A</td>
</tr>
<tr>
<td>Sample_SetCounterBActive</td>
<td>Set active external trigger channel B</td>
</tr>
<tr>
<td>Sample_SetTrigger</td>
<td>Set Trigger Mode</td>
</tr>
<tr>
<td>Sample_SetSampleTGC</td>
<td>Set TCG curve length of buffer</td>
</tr>
<tr>
<td>Sample_SetSingleGain</td>
<td>Set constant gain</td>
</tr>
<tr>
<td>DLL Command</td>
<td>Parameter</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------</td>
</tr>
</tbody>
</table>
| Long WINAPI Sample_ReleaseBaddr(long board); | Parameter:  
  • board - slot number of the opcard  
    board slot (0)  
Return: 0 if success |
| long WINAPI Sample_SetTimeout( long board, unsigned long time ) | Parameter:  
  • board - slot number of the opcard  
    board slot (0)  
  • time – timeout from 1 to 10000; for  
    0 timeout disable  
Return: 0 if success |
| long WINAPI Sample_GetAddr( long board, unsigned long *dirAddr[]) | Parameter:  
  • board - slot number of the opcard  
    board slot (0)  
  • pointer for address array  
Return: 0 if success |
| long WINAPI Sample_InitMeas( long board ) | Parameter:  
  • board - slot number of the opcard  
    board slot (0)  
Return: 0 if success |
| long WINAPI Sample_GetData( long board, unsigned long count, unsigned long *data ) | Parameter:  
  • board - slot number of the opcard  
    board slot (0)  
  • count – buffer counter  
  • pointer to array of sample  
Return: 0 if success |
| long WINAPI Sample_SendTGCArray( long board, unsigned long count, unsigned long filtr, unsigned long *data ) | Parameter:  
  • board - slot number of the opcard  
    board slot (0)  
  • count – buffer counter  
  • pointer to array of sample  
Return: 0 if success |
| long WINAPI Sample_InitTGC( long board ) | Parameter:  
  • board - slot number of the opcard  
    board slot (0)  
Return: 0 if success |
| long WINAPI Sample_Set20dB( long board, unsigned long setpreamp ) | Parameter:  
  • board - slot number of the opcard  
    board slot (0)  
  • setpreamp – 0 for OFF  
    1 for –20dB  
Return: 0 if success |
| long WINAPI Sample_SetDepth( long board, unsigned long depth) | Parameter:  
  • board - slot number of the opcard  
    board slot (0)  
  • depth –  
    0 for 256  
    1 for 512  
    2 for 1k  
    3 for 2k  
    4 for 4k  
Return: 0 if success |
<table>
<thead>
<tr>
<th>Function</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
</table>
| long WINAPI Sample_SetDelay( long board, unsigned long hdelay) | Parameter:  
• board - slot number of the opcard  
• hdelay - 0 to 255 (us) | Return: 0 if success |
| long WINAPI Sample_SetFreqTGC( long board, unsigned long freqtgc) | Parameter:  
• board - slot number of the opcard  
• freqtgc - 0 for 50MHz  
• 1 for 25MHz  
• 2 for 12.5MHz  
• 3 for 6.25MHz | Return: 0 if success |
| long WINAPI Sample_SetFreq( long board, unsigned long freq) | Parameter:  
• board - slot number of the opcard  
• freq - 0 for 100MHz  
• 1 for 50MHz  
• 2 for 25MHz  
• 3 for 12.5MHz | Return: 0 if success |
| long WINAPI Sample_SetPulser( long board, unsigned long vpulser) | Parameter:  
• board - slot number of the opcard  
• vpulser - 0 to 15 | Return: 0 if success |
| long WINAPI Sample_SetSource( long board, unsigned long source) | Parameter:  
• board - slot number of the opcard  
• source - 1 - PE mode  
• 2 - TT mode | Return: 0 if success |
| long WINAPI Sample_SetFiltr( long board, unsigned long filtr) | Parameter:  
• board - slot number of the opcard  
• filtr - 0 to 15 | Return: 0 if success |
| long WINAPI Sample_SetChannel( long board, unsigned long channel) | Parameter:  
• board - slot number of the opcard  
• channel - 1 to 32 | Return: 0 if success |
| long WINAPI Sample_SetCounterA( long board, unsigned long counterA) | Parameter:  
• board - slot number of the opcard | Return: 0 if success |

5 for 8k  
6 for 16k  
7 for 32k  
8 for 64k  
9 for 128k
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
<th>Parameters</th>
<th>Return</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Sample_SetCounterB</code></td>
<td>Sets counter B</td>
<td><code>long board, unsigned long counterB</code></td>
<td>0 if success</td>
</tr>
<tr>
<td><code>Sample_SetCounterAActive</code></td>
<td>Activates counter A</td>
<td><code>long board</code></td>
<td>0 if success</td>
</tr>
<tr>
<td><code>Sample_SetCounterBActive</code></td>
<td>Activates counter B</td>
<td><code>long board</code></td>
<td>0 if success</td>
</tr>
<tr>
<td><code>Sample_SetTrigger</code></td>
<td>Sets trigger</td>
<td><code>long board, unsigned long trigger</code></td>
<td>0 if success</td>
</tr>
<tr>
<td><code>Sample_SetSampleTGC</code></td>
<td>Sets sample TGC</td>
<td><code>long board, unsigned long divtgc</code></td>
<td>0 if success</td>
</tr>
<tr>
<td><code>Sample_SetSingleGain</code></td>
<td>Sets single gain</td>
<td><code>long board, unsigned long filtr, unsigned long singleg</code></td>
<td>0 if success</td>
</tr>
</tbody>
</table>
## Standard software

### Overview of Hotkey Selection:

<table>
<thead>
<tr>
<th>Hotkey</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td>STOP/START enables / disables selected acquisition mode – start / stop measurement</td>
</tr>
<tr>
<td>F2</td>
<td>Load Settings Load the configuration/settings file</td>
</tr>
<tr>
<td>F3</td>
<td>Save Settings Saving the configuration/settings file</td>
</tr>
<tr>
<td>F4</td>
<td>Print Screen This option will create a hardcopy of the screen into graphics mode compatible printers. The printout will include everything currently on the screen.</td>
</tr>
<tr>
<td>F5</td>
<td>Load data Load in a previously saved data/measurement file and restore the capture settings as they were when the data was saved</td>
</tr>
<tr>
<td>F6</td>
<td>Save data Saving the current measurement/data and current configuration/settings to the file</td>
</tr>
<tr>
<td>F7</td>
<td>Pattern</td>
</tr>
<tr>
<td>F8</td>
<td>Protocol</td>
</tr>
<tr>
<td>F9</td>
<td>Info Short information about the OPTEL Sp. z o.o</td>
</tr>
<tr>
<td>F10</td>
<td>Exit</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Pulser Ampl</strong></td>
<td>Set impulse voltage from off to 360V range in 13 levels</td>
</tr>
<tr>
<td><strong>Mode</strong></td>
<td>PE – Measured input signal (sending/receiving); TT – Measured input signal (receiving)</td>
</tr>
<tr>
<td><strong>Channel</strong></td>
<td>This function allows choosing the channel, if multiplexer is used.</td>
</tr>
<tr>
<td><strong>Filtr [MHz]</strong></td>
<td>Set bandwidth pass filter: 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;</td>
</tr>
<tr>
<td><strong>Sampling</strong></td>
<td>Set sampling frequency: 100MHz, 50MHz, 25MHz or 12.5MHz</td>
</tr>
<tr>
<td><strong>Trigger</strong></td>
<td>Set Trigger Mode – internal/ext. Internal – software trigger</td>
</tr>
<tr>
<td><strong>Active Count.</strong></td>
<td>A - Counter for Inc. Encoder Channel 1</td>
</tr>
<tr>
<td><strong>Counter A</strong></td>
<td>B - Counter for Inc. Encoder Channel 2</td>
</tr>
<tr>
<td><strong>Counter A</strong></td>
<td>A - Set divider for counter for incremental encoder – for encoder channel 1</td>
</tr>
<tr>
<td><strong>Counter B</strong></td>
<td>B - Set divider for counter for incremental encoder – for encoder channel 2</td>
</tr>
<tr>
<td><strong>Attenuator</strong></td>
<td>Set attenuator; off / -20dB</td>
</tr>
<tr>
<td><strong>GainMode</strong></td>
<td>Set gain mode (switching between DAC and constant gain and freehand DAC)</td>
</tr>
<tr>
<td><strong>Gain [dB]</strong></td>
<td>Set constant gain / amplification from -29 to +67[dB]</td>
</tr>
<tr>
<td><strong>Exponent</strong></td>
<td>Set DAC curve using exponential function</td>
</tr>
<tr>
<td><strong>Offset</strong></td>
<td>Allows to set the voltage offset</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td>This function allows to average the signal (using the mean worth of many measurements). Averaging of 2, 4, 8, 16 and 32 signals is possible.</td>
</tr>
<tr>
<td><strong>Window</strong></td>
<td>Set the length of the</td>
</tr>
</tbody>
</table>
**Delay** - The time distance between the pulse (trigger) and the beginning of the window, that is visible on the screen can be set.

**Measurement Gates**

Using cursors it is possible to set two measurement gates in which different functions can be performed.

In the part of the main screen, visible here the positions of this gates and the maximum amplitude, attenuation, phase of signal is shown.

Control change the background color depending of color of the measurement gate.

**Colors settings**

Using the control it is possible to set personal color of the background of display, trace, grid, measurement gates, trace of the memories.

**On the top of main panel**

Choose between RF Signal and Detector (absolute, Positive and Negative) display in upper window.

The software allows storing a signal and showing it (for reference purposes) on the display. Three such memories are available. Click on the square in the upper right area of the window causes, that the signal actually shown is shown on the main window in a frozen form in the color of the memory square.

Special function for measuring thickness, sound velocity i.e. when transducer is with delay lines. Please see below **Sample 1**.

If the square is selected, then only default-white measurement gate are working and the velocity measurement is calculate from the trigger to the signal between the gate limits.

Choose units axis Y between [dB], [%], [V]
### In the middle of main panel

<table>
<thead>
<tr>
<th>Measuring thickness, sound velocity and time of flight</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Thickness [mm]</strong> 0.00</td>
</tr>
<tr>
<td>Ref. multipl. 1</td>
</tr>
</tbody>
</table>

This function works on following basis: the user chooses, which signals should be used for thickness and/or velocity measurement (for example the reflection from first and second wall of a sample with parallel walls). This allows to use this software with almost any kind of samples, containments etc. For people using this software it is necessary to have some knowledge about such kind of measurements and physics of ultrasounds.

The next step is to choose (toggle button "V"), if thickness or sound velocity should be measured. Than it is necessary to mark, which value should be used as constant. The remaining value will be calculated.

The information from the window Ref. Multipl. (Reflection multiplication) is used for this calculation – it informs the software between how many reflections the gates are set.

Time of flight shows the time of the signal in the first gate.

If the square is selected, then only default- white measurement gate are working and the velocity measurement is calculate from the trigger to the signal between the gate limits.

---

### On the bottom of main panel

<table>
<thead>
<tr>
<th>Choose units axis X between [us], [sample], [mm]</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Start</strong> 32.95</td>
</tr>
</tbody>
</table>

On the bottom window are two measurement (depending on setting units axis X cursors showing: frequency / time / distance)

**Cursor** - Using the control it is possible to set personal color of the third cursor on the upper window.

Bottom window can be used to show many different results. Its modes can be selected by click on the beam, placed on it right bottom side.
**Bottom window features**

**zoom**

Upper window – yellow trace of the measurement signal – with two measurement gates (white and blue) and the third red cursor for the zoom with is displayed on the bottom window.

Bottom window – with zoom of the measurement signal choosing by red cursor on the upper window

**FFT windowed**
This function shows the real part of the FFT (with Hamming window applied) of the whole signal, shown on the upper window:

Additional features are available:
**Frequency [MHz]** 2.44 - Domain frequency is shown.
**Filter** – Software/digital bandwidth filter can be used. Bandwidth is set by cursors. Default - white cursor is for low frequency limit and default – the blue cursor is for high frequency limit. If the square “filter” is selected, a digital bandwidth pass filter is applied to the measured signal.

**Ring slide** – for adjustment the axis Y ratio.

**FFT cursors**

This function shows the real part of the FFT (with Hamming window applied) of the signal, shown on the upper window in the region, embraced by the gate, that appears on the upper window in this mode (red on the above picture).

Additional features are available:

**Frequency [MHz]** - Domain frequency is shown.

**Filter** – Software/digital bandwidth filter can be used. Bandwidth is set by cursors. Default - white cursor is for low frequency limit and default – the blue cursor is for high frequency limit. If the square “filter” is selected, a digital bandwidth pass filter is applied to the measured signal.

**Ring slide** – for adjustment the axis Y ratio.

**FFT power spectrum**

It shows power spectrum of FFT of the signal from upper window, calculated with following formula:

\[
\text{Power Spectrum} = \frac{|\text{FFT}(X)|^2}{n^2}
\]
Additional features are available:

**Frequency [MHz]** - Domain frequency is shown.

**Filter** – Software/digital bandwidth pass filter can be used. Bandwidth is set by cursors. Default – white cursor is for low frequency limit and default – the blue cursor is for high frequency limit. If the square “filter” is selected, a digital bandwidth pass filter is applied to the measured signal.

**Ring slide** – for adjustment the axis Y ratio.

**Amplification of the received signal and gain curve (TGC function)**

The OPCARD has not only a simple amplifier, but a special function, allowing amplifying the signal depending on the time, passed from the start (trigger). This function is called *time-gain compensation* (TGC).

OPCARD allows setting the curve, which controls the amplifier during measurement using arbitrary waveform generator. Upper window allows controlling it, if it is used in gain curve mode. Additionally it is necessary to set (upper right part of the screen) the gain mode. It is possible to select following modes:

1. **Constant**: Gain is not changed during the time.
2. **Curve**: gain is changing exponentially. It will compensate for exponential signal decay (typical for sound signals) by amplifying the signal by an exponential factor that depends on elapsed time. Exponent is defined by the exponent setting.
3. **Free hand**: Amplification depends on the curve, that can be freely drawn using the upper window:

Red cursor cross in the upper window allows drawing the curve in the way that is wished by the user – after click on button: Acquire
To draw the curve, the user must click on **Acquire**, after the curve is ready on **Done** and **Set**, and the transfer of the curve to the card is fulfilled with set.

**Protocol Panel – under F8 Protocol on Main Panel**
Overview of Hotkey Selection:

<table>
<thead>
<tr>
<th>Hotkey</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1 Run</td>
<td>Periodical Measurement with adjusted interval and cycle length</td>
</tr>
<tr>
<td>F2 Run</td>
<td>Standard measurement with speed what fast computer can</td>
</tr>
<tr>
<td>F5 Sequence</td>
<td>It is also possible to decide, if the information from different channels should be added (to produce for example one spectrum information from many channels) or treated as a sequence (each channel is separated).</td>
</tr>
<tr>
<td>F6 Save *.csv</td>
<td>Save as csv - Saving all results to the file cvs</td>
</tr>
<tr>
<td>F8 Reset</td>
<td>Clear all measurement results</td>
</tr>
<tr>
<td>F10 Back</td>
<td>Back to the main panel</td>
</tr>
</tbody>
</table>

**Settings on the protocol panel**

It is possible to store the results as the whole measurements (the whole wave information - raw data) or as measurement results (thickness, wave velocity) or as FFT

- **Header:** "4 lines of text - new line [CTRL+ENTER]"
  - Measurement Protocol
  - Additional comments can be added to the csv file.

- **To file:**
  - Wave
  - Measured results
  - FFT/Spectrum
  - Choose what kind of result must be saved to the files.

**[F1] - Periodical Measurement**

- **Interval:** 1 seconds
- **Cycle length:** 13 seconds
- **Remaining Time:** 00:01:13
- **Remaining Meas:** 13
- **No. of Meas:** 13

**Settings for Periodical measurement.**
- Interval between measuring.
- Length of the measurement cycle.

**F5 Sequence panel**
**Exit from the OPCARD software:**

![Exit screen](image)

When the OPCARD software is to be closed last panel is displayed. If the settings should be saved then OPCARD software will start with this settings.

**EXAMPLE OF MEASUREMENTS WITH THE OPCARD**

For better understanding we have prepared an example, showing the use of OPCARD. Especially with the goal to explain such functions as: One cursor measurement, pattern use.

![Measurement of polystyrol](image)

Fig.1 Measurement of a piece of polystyrol, ca. 39 mm length.
Fig. 2 Measurement of parameters: time of flight, sound speed, distance, using pulse echo (PE) method.

First such parameters as window, delay, gain – depending on the measured object must be set and naturally the mode (PE). Now we must set the gates so, that the signal reflected from the wall is inside (first and the second reflection) – as shown on the picture 2. In the field “thickness” the correct thickness must be given. The sound speed in the measured material is shown in the field “Velocity”.

Fig. 3 Measurement of parameters: time of flight, sound speed, distance, using PE method and one cursor

First such parameters as window, delay, gain – depending on the measured object must be set and naturally the mode (PE) and the function One Cursor Measurement. Now we must set the white gate (as on the Fig. 3 (Gate1 – white) on the signal reflected from the sample wall. Thickness must be set correctly. The result of the sound speed measurement is visible in the field Velocity. The measurement with one cursor is not so exact as with two, because it includes hardware delay (in electronics and transducer used).
First such parameters as window, delay, gain – depending on the measured object must be set and naturally the mode (TT). Now we must set the gates so, that the signal transmitted and reflected from the wall is inside (first and the second reflection) – as shown on the picture 4. In the field “thickness” the correct thickness must be given. The sound speed in the measured material is shown in the field “Velocity”.

The measurement with one cursor is not so exact as with two, because it includes hardware delay (in electronics and transducer used).
This is possible with the function "pattern". It allows to exclude automatically the transmission time through the delay line from the calculations. For this purpose it is first necessary to measure time of flight through the delay line. Than the measured signal should be stored as pattern. This is done with pressing F7. Now the gate should be set so, that the signal from the measured object is inside. This function can be used both in TT and PE mode.
Fig. 8 Signal measured for the object as shown on the Fig. 7. Please note, that the signal coming from the object is between the signals coming from the delay line (transmission and first reflection). Gates shown here are not giving correct results. It is necessary to set the gate as on the fig. 9.

Fig. 9 Measurement of parameters: time of flight, sound speed, distance, using TT method and with delay line. White gate is set on the stored pattern, and blue on the signal going through the object and delay line.

Some additional remarks.
All modes (PE, TT, with one or two cursors) are working perfectly and are giving real results only in the case, when gates are set correctly on proper signals. This requires some experience, especially in the case of complicated objects.

If you have any question, please do not hesitate to contact us.