

# CURVV

Revision 1.  
Embedded software version 1.

## 1. Features

The device is designed for digital input/output in control systems of accelerators as embedded intelligent controller.

The device includes:

- 8-channel output register with galvanically isolated outputs;
- 8-channel input register with galvanically isolated inputs;
- 8-channel output register with power outputs;
- 24-channel input register with TTL inputs;
- CANBUS interface for interaction with control computer;
- micro-controller.

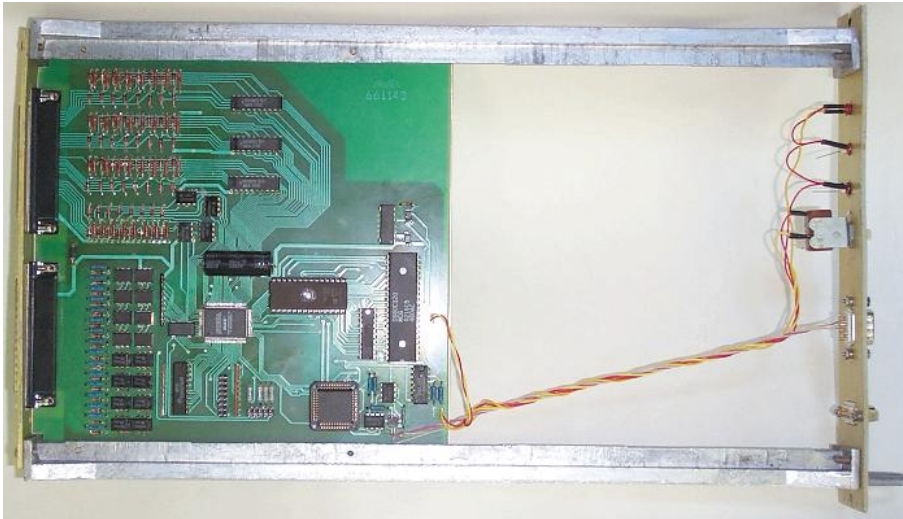


Photo of CURVV.

The device is intended to be embedded in power supply racks. The device requires for proper operation the only powers supply with voltage +5V ( $\pm 5\%$ ).

## 2. Specifications:

1. Channels of output register (with optical isolation) – 8.
2. Maximal voltage for output register- 50V.
3. Maximal current for output register- 10 ma.
4. Channels of input register (with optical isolation) – 8.
5. Voltage for input register- 2.5-6.0V.
6. Input resistance for input register- 510 Ohm.
7. Isolation voltage for registers - 1000 V.
8. Channels of output register (power) – 8.
9. Maximal voltage for output register (power)- 15 V.
10. Maximal current for output register (power)- 300 mA.
11. Channels of input register (TTL inputs) – 24.
12. Voltage for input register (TTL inputs)- 0-6.0 V.
13. Input resistance for input register (TTL inputs)- 1 KOhm (between input and +5 V).

14. CANbus transceiver is galvanically isolated from network and it is in compliance with ISO 11898-24V (chip PCA82C251).
15. Voltage between transmission line and device- 1000V.
16. Hardware implementation allows using both standard and extended CANbus frames. Software implementation is based on standard frames (short identifier).
17. Baud rates- 1000, 500, 250, 125 Kbaud (may be chosen by jumpers).
18. Voltage of power supply- +5V,  $\pm 5\%$ .
19. Power supply current-  $<0.6A$  (typical value- 0.4A).

### 3. External connections

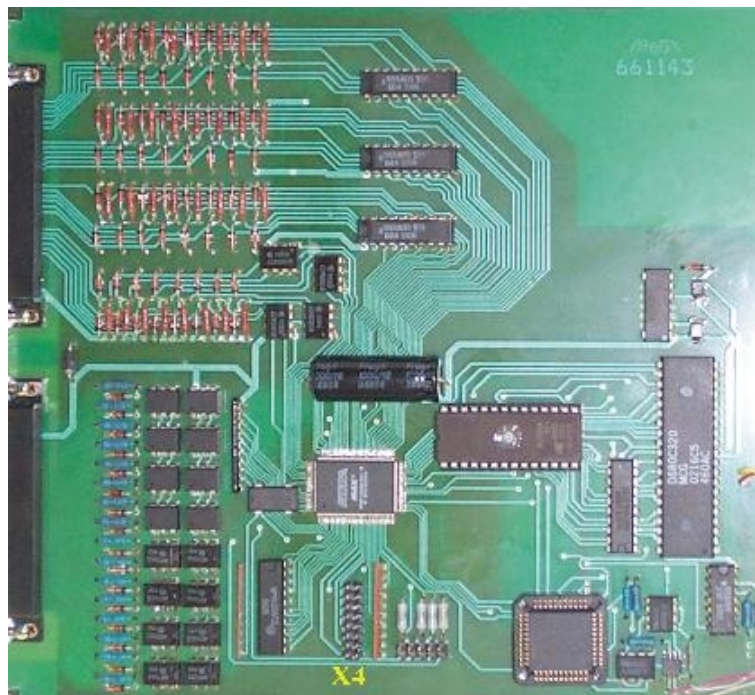
The device CURVV is implemented as module in “WISHNYA” standard. A front panel of the device contains a network connector (DB-9M), RESET button and one LED. The LED is blinking during processing CANbus messages. Connections with external channels of control carry out by DRB connectors on back panel. Power register outputs and TTL inputs of device are connected with pins of X1 connector. Outputs and inputs of registers with optical isolation are connected with X2 connector.

#### 3.1. Jumpers

The device CURVV has the following set of jumpers:

X4 includes 8 jumpers. 6 jumpers define number (address) of device in network (this number is used to compose identifier of messages) and 2 jumpers define baud rate.

Jumpers location is shown on board photo.



Jumpers location on the board.

### Destination of jumpers in X4 group.

Designation	Location	Destination
X4-7	Upper	N5- included in device number (most significant bit)
X4-6	...	N4- included in device number
X4-5	...	N3- included in device number
X4-4	...	N2- included in device number
X4-3	...	N1- included in device number
X4-2	...	N0- included in device number (least significant bit)
X4-1	...	BR1- defines baud rate
X4-0	Lower	BR0- defines baud rate

Jumpers N5...N0 defines logical number (address) of device which is used to compose message identifier for CANbus network (for more detail see PROTOCOL part of this description). An installed jumper should be interpreted as logical 0 and absence of jumper should be interpreted as logical 1.

BR1	BR0	Baud rate
Connected	Connected	1000 Kbit/sec
Connected	Disconnected	500 Kbit/sec
Disconnected	Connected	250 Kbit/sec
Disconnected	Disconnected	125 Kbit/sec

### NOTES:

1. CANbus is bus with multiple access and incorrect baud rate setting may affect on transfer messages of other devices in addition to impossibility of access to this device.
2. In network may exist concurrently devices with identical numbers (addresses). Formally it is permissible, but actually it do cause a lot of problem. Connecting to network devices with identical numbers is strictly not recommended.



### 3.2 Front panel.

A front panel includes:

- Line LED**
- Reset button**
- CANbus connector**

**Line LED** is blinking during processing CANbus messages by onboard processor.

After power-on the device blinks by LED a few times.

**Reset button** is intended for hardware reset. It isn't intended for daily using.

**CANbus connector** (DB-9M) is intended for connection to media. Pin designations follows below in table.

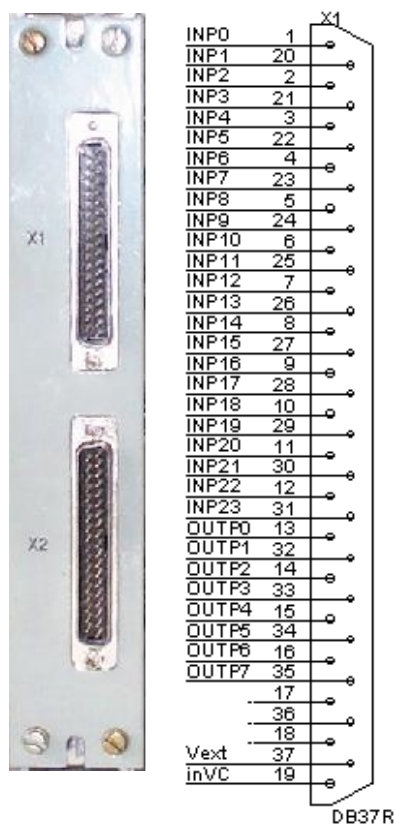
2	CAN-L	One wire in pair
3	GND	Shield of cable
7	CAN-H	One wire in pair

B Shielded twisted pair is used as media. According to the ISO 11898-2 it should has nominal impedance 120 Ohm. Line termination has to be provided through termination resistors of 120 Ohm located at both ends of the line.

### 3.3 Back panel.

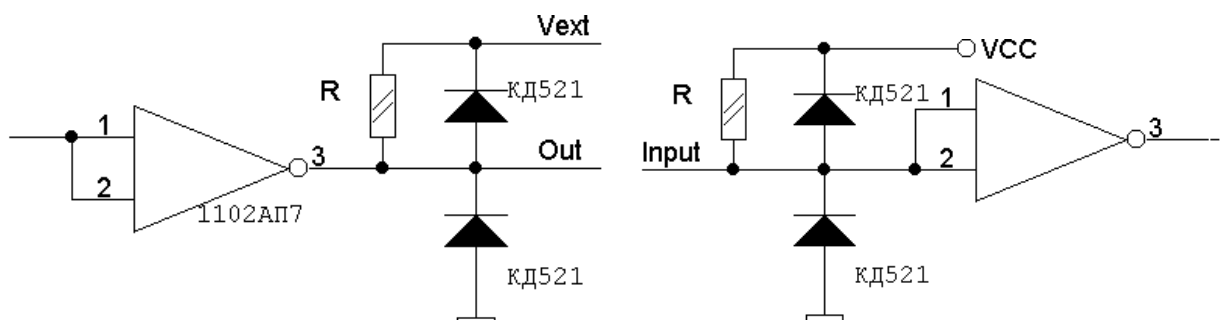
A two connectors are placed on the back panel. Connection external signals with device should be done through these connectors. X1 connector contains power register outputs and TTL inputs and X2 connector includes outputs and inputs of registers with optical isolation and power supply pins.

#### 3.3.1 X1 connector.

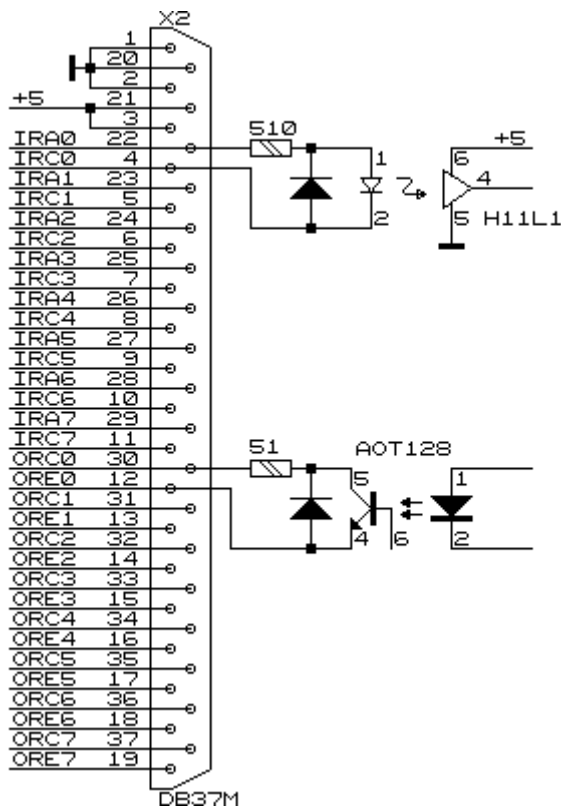


X1 connector provides pins of power output register and TTL input registers. Below is shown schematics of power output and TTL input registers. All outputs of power register have connection with common line by resistor 1,3 KOhm. It is possible to power this line by voltage 5-15 V. All inputs of TTL input register are connected with internal power supply (+5 V) by resistor 1,3 KOhm.

Disconnected input of TTL input register is interpreted as logical 0. Writing 0 in power output register switch-off a current in load (disconnects load).



### 3.3.2 X2 connector.



X2 connector provides pins of input and output registers. Powering of the device should be done through pins of X2. The device requires the only external power supply with voltage +5V ( $\pm 5\%$ ). There is shown a location of signals on the connector's pins. A fragment of circuitry is included to show an implementation of input and output registers. Both registers are designed with galvanic isolation that is provided by optocouplers.

Design of input register is based on a chip H11L1. It is intended for detection external voltage or current. Input voltage range is from 3V to 12V. Input current range is from 4mA to 20mA. Unconnected input (no current in LED) is interpreted as logical 0.

Output register design is based on transistor optocoupler and it is able to provide output current up to 10 mA. Output voltage may be up to 30V.

## 4. Basics of operations for CURVV

The CURVV contains 6 registers of digital input/output. Two registers intended for output data and four registers may be used for input of digital data. One input register and one output register have an optical isolation. Other registers are connected with common "ground" of device. Power output register may be used for controlling relay. After power-up on-board micro-controller writes zero into outputs registers. It means "No current" in external load. Disconnected pins of input registers are interpreted as logical 0.

Registers with optical isolation are identical to registers in other CANbus devices. This identity are hardware (connectors pin-out) and software (these registers are controlled by standard messages). New registers are connected with pins of X1 connector and controlled by new message.

## 5. A command set for CURVV

### Identifier bits distribution

Identifier bits	ID10...ID08	ID07...ID02	ID01...ID00
Bit field	Field 1	Field 2	Field 3
Destination	Priority	Address	Reserve

Comments to bit distribution:

Field 1 – priority field (type field):

Code 5 – a broadcast message (field 2 is ignored).

Code 6 – ordinary (address) message.

Code 7 – response (reply for type 6 message).

Code 0 is forbidden, other combination is not used (they are reserved for future extensions).

Field 2 – a physical address field. It defined address device (this address is defined by jumpers on-board).

Field 3: User should set zero in this field. The device can send messages with different values in this field.

Any device on receiving address message interprets an information by its content. If received message requires a reply, the device sends required information by message with code 6 (response type message). A broadcast messages should be received by all devices simultaneously and required actions should be done in all devices.

An interpretation of data fields:

On receiving message a device interprets data in following way: a first byte (byte 0) is descriptor of message, the other bytes are an additional information.

There is a list of message descriptors (codes are hexadecimal).

E8 - request data from new registers

E9 - write to new power register

F8 - request for data from input and output registers with optical isolation

F9 - write to output register with optical isolation

FF - device attributes request

### **Detail description of messages for CURVV (all codes are hexadecimal)**

**E8** - request data from TTL registers, no additional parameters. As a response the device sends the following message:

E8	DO0	0	DI0	DI1	DI2	0
----	-----	---	-----	-----	-----	---

Here:

DO0 – data written to power output register (bits OUTP0-OUTP7);

DI0 – data from input TTL register (low byte, bits INP0-INP7);

DI1 – data from input TTL register (middle byte, bits INP8-INP15);

DI2 – data from input TTL register (high byte, bits INP16-INP23).

**E9** - write data to output power register.

Byte 1 contains data to be written into output power register (bits OUTP0-OUTP7).

**Message F8** – request data from registers. This message has not additional information. In reply a device sends a message with output register byte and input register byte.

F8	Output Register Data	Input Register Data
----	----------------------	---------------------

**Message F9** – write data to output register.

Byte 1 contains information to be written into output register. The device don't respond on this message.

F9	Output Register Data
----	----------------------

**Message FF** – device attribute request. There is not additional parameters. In reply a device sends the following message:

FF	Device Code	HW version	SW version	Reason
----	-------------	------------	------------	--------

Device Code- device type (for CURVV it is equal 10).

HW version- hardware version of device.

SW version- software version of device.

Reason- reason of sending this message:

- 0- After power-up.
- 1- After reset by button on front panel.
- 2- On request by address message with code FF.
- 3- On request by broadcast message (who is here?).
- 4- On restart by Watchdog timer.
- 5- On busoff recovery.

### **BROADCAST messages**

This device recognizes the only broadcast message- FF.

FF- request “Who is here”. On this broadcast request all devices on-line must send into network message with their attributes (and identifier).

## **6. Software versions for CURVV**

There will be described modifications for software versions beyond 2<sup>nd</sup>.

### **Version 2.**

- 1.