# CPKS8

Revised 16-nov-2011.
Embedded software version 1.

## 1. Features

The device is designed for using in control systems of accelerators. The device generates 8 output PWM signals. Similar devices (CAMAC modules) were widely used in our institute in 70-80 years. The new device (CPKS8) is intended for modernization of obsolete systems.

The device includes:

- 8 channels PWM generator;
- CANbus interface for interaction with control computer;
- Micro-controller.

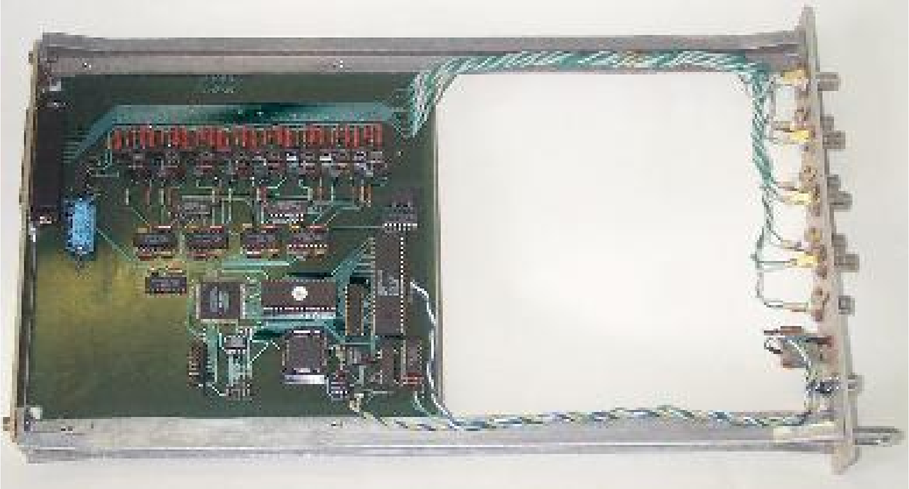


Photo of device

The device may be interpreted as 8 independent PWM generators. All outputs have common "ground" connected with power supply. An user should provide a terminator at the end of output line (75 or 50 Ohm) because the device have current outputs.

The PWM generators are implemented in PLD. As a result, a system traffic and software tricks cannot affect on stability and jitter of output pulses. The device is intended to be embedded in "WISHNYA" racks. The device requires for proper operation the only power supply with voltage +5V (±5%).

## 2. Specifications:

1. Resolution of PWM generator – 16 bits.
2. Output channels – 8.
3. Output voltage ±12÷15 B.
4. Output pulse duration – 300 nS.
5. Terminator resistance 75 Ohm.
6. Time quantum (discreteness) - 100 nS.
7. Accuracy of internal timing - 0.1%.
8. CANbus transceiver is galvanically isolated from network and it is in compliance with ISO 11898-24V (chip PCA82C251).

9. Hardware implementation allows using both standard and extended CANbus frames. Software implementation is based on standard frames (short identifier).
10. Baud rates- 1000, 500, 250, 125 Kbaud (may be chosen by jumpers).
11. Voltage of power supply- +5V, ±5%.
12. Power supply current- <0.5A (typical value- 0.35A).

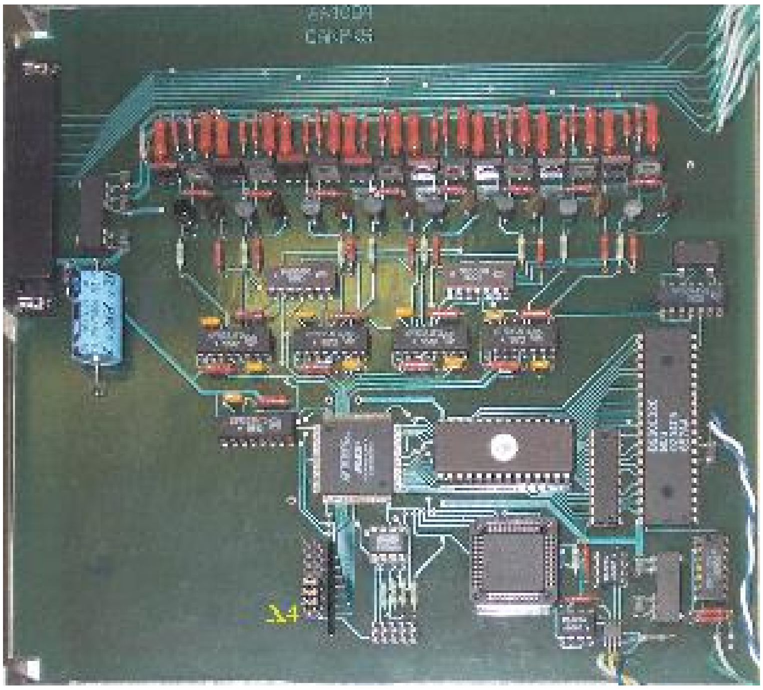
## 3. External connections

The device is implemented as module in “WISHNYA” standard, width is 40 mm. A front panel of the device contains a network connector (DB-9M), RESET button and one LED. The LED is blinking during processing CANbus message. Connection with external channels of control and measurements carry out by DRB-37M connectors on back panel.

## 3.1. Jumpers

The device has a set of jumpers (they are labeled on the photo as X4). 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 below on board photo.



Jumpers location on 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.

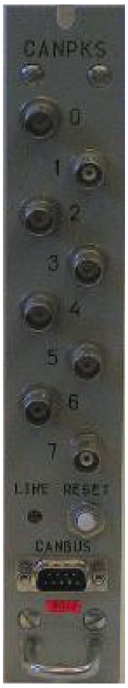
Baud rate defining.

| 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 permissibly, but actually it does 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<br>
**Reset** button<br>
**Output0-Output7** connectors<br>
**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 follow below in table.

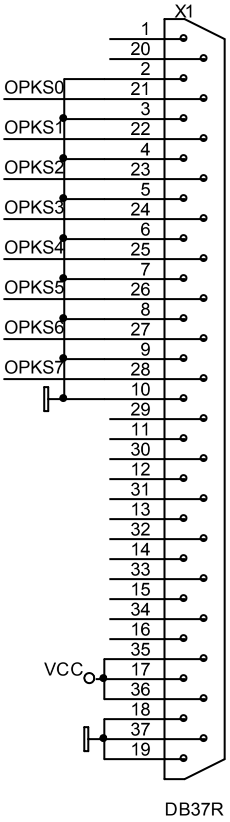
Output connectors **(0-7)** are intended for connecting with controlled devices.

**Connecting the device with CANbus line.**

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

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.



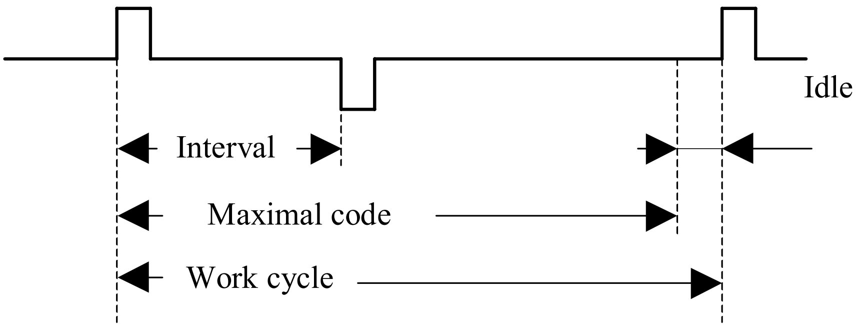
There is connector (DRB-37M) on the back panel. Connection external devices with CPKS8 should be done through this connector. It contains also power supply pins. Controlled devices may be connected with CPKS8 from front panel. An user have a choice which connectors to use. All outputs have common wire connected with "ground" of the device.

## 4. Basics of operations for CPKS-8

As was mentioned above the device includes the following components:

- a multi-channel PWM generator;
- a CANbus interface;
- a micro-controller for integration all parts of the device and interaction with control computer.

After power-up the micro-controller writes zero values into PWM channels and sends to computer a message with attributes. A time interval is coded by two pulses. A positive pulse means start moment of interval and negative pulse means end of interval. It is shown on drawing.



A work cycle consist of maximal interval (65536 quanta- 6,55 msec) and idle interval (32 mcsec). A positive pulse labels a start of work cycle. The positive pulse is followed by negative pulse after defined interval. The device generates positive pulse on all outputs simultaneously.

If there is defined a short interval (0-3 quantum) both pulses may be overlapped. To avoid it the device shortens width of start pulse. If there is defined zero interval then start pulse is disappeared at all. Only stop pulse is present on output in this case.

## 5. A command set for CPKS -8

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).
00 - 07 – write a delay code to CPKS-8 channel 0 - 7
10 - 17 – request delay code from CPKS-8 channel 0 – 7

FE – device status request
FF - device attributes request

On receiving a request for data message the device sends to host a response message. In the response message the first data byte contains the same value as the first data byte in request message. On receiving a write data message the device don’t send any message in reply.

Детализация информационных пакетов для различных типов
(все коды приводятся в 16-ричном виде)

**00 - 07** – (write to delay register of CPKS8 channel 0-7), the following bytes are data bytes.

Example:

| 04 | 12 | 11 |
|---|---|---|
| 4th channel | Byte 0 | Byte 1 |

This message places into 4-th CPKS channel value 2828 (decimal). If time quantum equals 100 nSec then this value defines delay as 282.8 mcSec.
Byte 1 is the most significant byte and byte 1 is the least significant byte.

**10 - 17** – (request code from delay register of CPKS8 channel 0-7). The following bytes will be ignored. In reply the device sends a message with the same code but with data from delay register (bytes 0, 1).

**FE** - request device status. It hasn't any parameters. In reply a device sends the following message:

| | |
|---|---|
| FE | Status |

Status- status of device. There are flags:
b7 – PLD version. For CPKS8 it is 1.

**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 CPKS8 it is equal 7).
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.

The device sends this message on power-on.

### BROADCAST messages

For broadcast messages all devices analyze only field 1 in CANbus identifier. Valid combination is 5. A first byte of date presents a broadcast command. CPKS8 uses the only broadcast command:

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 CPKS-8

There will be described modifications for software versions beyond 1st.

**Version 2.**