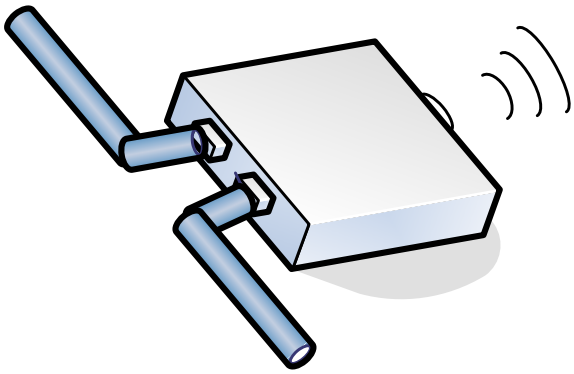
Infrared Transmitter Firmware
UNIV 1.0.5.1

1. Features:

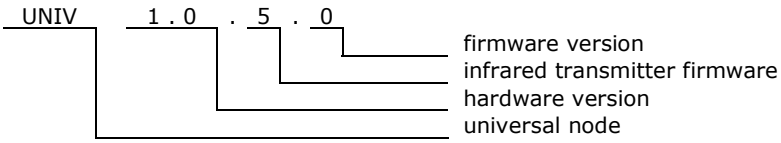


- 5 infrared codes:
  - SIRC 12 bit,
  - SIRC 15 bit,
  - Philips RC5A,
  - Samsung,
  - NEC.
- Allows to set up 24 conditions for receiving bus messages
- Can work as remote controller extender

2. Compatibility:

- Firmware for **UNIV 1.0.5.0 application**.
- Application firmware can be uploaded into devices with bootloader version 2.5 or compatible.

3. Firmware version



4. Operation overview

This is a transmitter of infrared codes: SIRC 12 and 15 bit, Philips RC5A, Samsung and NEC. Module allows setting up 24 conditions for receiving bus message. Can also be configured as remote controller extender. It can react then for codes received by infrared receivers and transmits them.

5. Firmware

Firmware can be uploaded by using HAPCAN Programmer, which can be downloaded from site http://siwilo.com/hapcan/software.

5.1. Infrared transmitter message

Module does not transmit messages to the bus.

5.2. Status request

Module does not respond to the status request.

5.3. Module control

Module can be controlled directly from PC, or indirectly by other modules. Instruction 0xC9 can be used only with indirect controlling.

5.3.1. Control instruction

The table shows instructions executed by module.

Table 1. Instructions coding

Instruction	Instruction code			Description
	INSTR1	INSTR2	INSTR3	
SEND SIRC 12bit CODE	0x00	ADDRESS	COMMAND	Transmits SIRC 12bit code, 32 addresses and 128 commands
SEND SIRC 15bit CODE	0x01	ADDRESS	COMMAND	Transmits SIRC 15bit code, 256 addresses and 128 commands
WYŚLIJ KOD RC5A	0x02	ADDRESS	COMMAND	Transmits Philips RC5A code, 32 addresses and 128 commands
WYŚLIJ KOD Samsung	0x03	ADDRESS	COMMAND	Transmits Samsung code, 256 addresses and 256 commands
WYŚLIJ KOD NEC	0x04	ADDRESS	COMMAND	Transmits NEC code, 256 addresses and 256 commands
SEND RECEIVED CODE	0xC9	0xXX	0xXX	Transmits the code that was received from the bus

0xXX - any data value

5.3.3. Direct control

It is possible to control module by sending DIRECT CONTROL message. The message contains instruction, which will be executed by module. The module can be also controlled from HAPCAN Programmer.

Table 2. DIRECT CONTROL frame (0x10A).

Frame type	Flags	Module	Group	D0	D1	D2	D3	D4	D5	D6	D7
0x10A	0x0	COMP ID1	COMP ID2	0xXX	0xXX	Node Nr	Group Nr	0xXX	INSTR1	INSTR2	INSTR3

0x10A – DIRECT CONTROL frame

COMP ID1 - computer identifier (must be unique on the network)

COMP ID2 - computer identifier (must be unique on the network)

Node Nr - node number of requested module

Group Nr - group number of requested module

INSTR1 - instruction to be executed (byte1)

INSTR2 - instruction to be executed (byte2)

INSTR2 - instruction to be executed (byte3)

0xXX - any value

5.3.4. Indirect control

Indirect control means that module will react to messages sent by other modules on the network. It depends on configuration programmed into the module. It is possible to configure the module, so it can send codes received by infrared receivers. The module can send codes received by one IR receiver or group of receivers.

5.4. Configuration

With this version of firmware parameters below can be configured:

- Module identifier (module number and group number);
- Module description (16 chars);
- Linking device with other modules (indirect control of module)

Configuration process can be done by using HAPCAN Programmer.

5.4.1. Module identifier

Every module on the network must have unique identifier. The identifier is made of two bytes, module number (1 byte) and group number (1 byte). Belonging to particular group might be important when linking devices e.g. some modules can react to messages sent by any node in a particular group.

5.4.2. Module description

Every module can have 16 char description, which makes easier for user (programmer) to distinguish nodes.

5.4.3. Linking devices

The module has 24 memory cells (boxes). Each box can contain information about message sent by other node and instruction which will be executed when that message is received.
This version of application allows flexible programming of conditions for node behaving. The node can react when particular device or group of devices send message. It is possible to choose which data bytes will be matched as condition. The HAPCAN Programmer makes configuration process easier.

This firmware has also feature to set simple conditions of executing instruction. To do so you can use conditional instruction shown in the table below. As an example of simple condition can be situation when light has to be turned on by PIR when someone enters room, but should not be during a day. The HAPCAN Programmer simplifies configuration process.

Table 3. Coding of relay instructions

Instruction	Instruction code			Description
	INSTR1	INSTR2	INSTR3	
ENABLE BOX	0xDD	X	Y	It enables chosen boxes – these boxes will be compared with next received message from the bus.
DISABLE BOX	0xDE	X	Y	It disables chosen boxes – these boxes will be passed when next message arrives from the bus.
TOGGLE BOX	0xDF	X	Y	It toggles boxes – enables when they are disabled and vice versa

INSTR2	Description
0x00	Box 1
0x01	Box 2
…	…
0x17	Box 24

INSTR3	Description
0x00	+ 0 -(and not anyone)
0x01	+ 1 -(and 1 following)
…	…
0x17	+ 23 -(and 23 following)

6. Document version

File	Note	Date
univ_v1-0-5-1a.pdf	Original version	August 2009