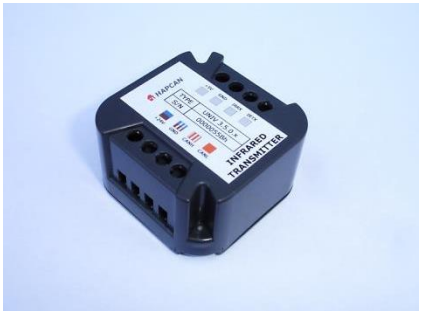
HAPCAN
HOME AUTOMATION

INFRARED RECEIVER / TRANSMITTER FIRMWARE
UNIV 3.5.0.3

1. Features



- Firmware for Infrared Receiver/Transmitter UNIV 3.5.0.x.
- Receives and recognizes infrared codes
  - Sony SIRC 12 bit (32 addresses, 127 commands) – 4k codes
  - Sony SIRC 15 bit (256 addresses, 127 commands) – 32k codes
  - Sony SIRC 20 bit (8192 addresses, 127 commands) – 1M codes
  - Philips RC5 (32 addresses, 127 commands) – 4k codes
  - NEC 16 bit (256 addresses, 256 commands) – 65k codes
  - NEC 24 bit (65280 addresses, 256 commands) – 16M codes
- Emits above codes
- Learns IR codes from original remotes which have carrier frequency between 27 – 49kHz and maximum 335 bits length
- Emits any IR codes with carrier frequency 18 - 500kHz
- It has memory for 100 IR learnt or converted from Pronto format codes
- Allows defining up to 128 CAN messages which can indirectly control the module
- Transmit (42 messages) and receive (42 messages) FIFO buffers

2. Compatibility

- Firmware for **UNIV 3.5.0.x** module
- Firmware can be uploaded into processor with bootloader version 3.1 or compatible.
- **In current firmware version the learnt IR codes must end with a value of 00, so codes scanned in previous versions are not compatible. Please re-scan codes or edit them and the last 7F value replace with 00.**

3. Firmware version

UNIV 3 . 5 . 0 . 3

- 3 – firmware version
- 0 – application version
- 5 – application type (infrared receiver / transmitter)
- 3 – processor version
- UNIV – universal processor

4. Communication Frames (messages)

4.1. Infrared Receiver frame

The module sends information of received infrared code, and another message when infrared transmission stops.

Table 1. INFRARED RECEIVER frame – beginning of receiving the code

Frame type	Flags				Module	Group	D0	D1	D2	D3	D4	D5	D6	D7
0x303	3	2	1	0	Node Nr	Group Nr	0xFF	0xFF	CODE TYPE	CODE1	CODE2	CODE3	0xFF	0xFF

0x302 – universal module frame, relay

3	- - not used flag, read as "0"
2	- - not used flag, read as "0"
1	- - not used flag, read as "0"
0	RE - response flag, flag is equal "1" if node was requested. If flag is equal „0" it means that status of output has just changed.

Node Nr - message sender node number

Group Nr - message sender group number

CODE TYPE
0x03 - received SIRC 12 bit code (32 addresses, 127 commands),
CODE1=address, CODE2=command, CODE3=0xFF
0x04 - received SIRC 15 bit code (256 addresses, 127 commands),
CODE1=address, CODE2=command, CODE3=0xFF
0x05 - received SIRC 20 bit code (8192 addresses, 127 commands),
CODE1=address MSB, CODE2=address LSB, CODE3=command
0x06- received RC5 code (32 addresses, 127 commands),
CODE1=address, CODE2=command, CODE3=0xFF
0x07- received NEC 16 bit code (256 addresses, 256 commands),
CODE1=address, CODE2=command, CODE3=0xFF
0x08 - received NEC 24 bit code, (65280 addresses, 256 commands)
CODE1=address MSB, CODE2=address LSB, CODE3=command

When transmission of infrared signal stops, the module sends another message. The difference between message at the beginning and at the end of transmission is in a D2 byte.

Table 2. INFRARED RECEIVER frame – end of the IR transmission

Frame type	Flags	Module	Group	D0	D1	D2	D3	D4	D5	D6	D7
0x303	0x1	Node Nr	Group Nr	0xFF	0xFF	CODE TYPE +0x80	CODE1	CODE2	CODE3	0xFF	0xFF

4.2. Learning IR codes

The module enables learning and saving infrared codes into memory. Scanning should be performed in two steps: scanning the carrier frequency signal and then scanning the signal content. The current version of the HAPCAN Programmer scans only the content of the signal and assigns the default 38kHz carrier frequency.

After receiving instructions "*Learn code - the carrier frequency of the transmitted IR signal from the remote controller*" (control instructions) and correct receiving IR code, the module sends carrier frequency frame. If the module does not receive the IR code, then sends the error frame (Table 5)

Table 3. LEARNT IR SIGNAL CARRIER FREQUENCY frame

Frame type	Flags	Module	Group	D0	D1	D2	D3	D4	D5	D6	D7
0x303	0x1	Node Nr	Group Nr	0x00	CARRIER	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF

0x3031 – universal module, infrared receiver frame

Node Nr - message sender node number

Group Nr - message sender group number

0x00 - frame data type (0x00 – IR carrier frequency frame)

CARRIER - received frequency codded: f = 4608/(CARRIER+1) [kHz]

After receiving instructions "*Learn code - the body of transmitted IR signal from the remote controller*" (control instructions) and correct receiving IR code, the module sends 96 frames containing IR signal body. If there is any problem with code learning, the module sends error frame (Table 5)

Table 4. LEARNT IR SIGNAL CODE frames

Frame type	Flags	Module	Group	D0	D1	D2	D3	D4	D5	D6	D7
0x303	0x1	Node Nr	Group Nr	FRAME No	CODE0	CODE1	CODE2	CODE3	CODE4	CODE5	CODE6

0x3031 – universal module, infrared receiver frame

Node Nr - message sender node number

Group Nr - message sender group number

FRAME No - number of frame with IR code (1-96)

CODE0 - IR signal carrier frequency taken from *IR signal carrier frequency frame* (Table 1)

CODE1-672 - data of received IR (671 bytes sent in 96 frames)
Bit <7> of each byte defines if this is IR burst (bit <7> = „1") or space (bit <7> = „0") in IR code
The length of the burst or space is equal: 40us * byte_value (7 bit value -without 7th bit)

If there is any problem with code learning, the module sends error frame.

Table 5. LEARNT CODE ERROR frame

Frame type	Flags	Module	Group	D0	D1	D2	D3	D4	D5	D6	D7
0x303	0x1	Node Nr	Group Nr	0xF0	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF

0x3031 – universal module, infrared receiver frame

Node Nr - message sender node number

Group Nr

0xF0 - frame data type (0xF0 – learnt code error frame)

4.3. Status request

Module doesn't respond to status request.

4.4. Uptime request

Table 6. UPTIME REQUEST (0x113).

Frame type	Flags	Module	Group	D0	D1	D2	D3	D4	D5	D6	D7
0x113	0x0	COMP ID1	COMP ID2	0xXX	0xXX	Node Nr	Group Nr	0xXX	0xXX	0xXX	0xXX

0x1130 – UPTIME REQUEST 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
0xXX - inessential data

Table 7. Response to UPTIME REQUEST (0x113).

Frame type	Flags	Module	Group	D0	D1	D2	D3	D4	D5	D6	D7
0x113	0x1	Node Nr	Group Nr	0xFF	0xFF	0xFF	0xFF	UPTIME3	UPTIME2	UPTIME1	UPTIME0

0x1131 – Response to UPTIME REQUEST frame

Node Nr - node number on the network
Group Nr - group number of the node on the network
UPTIME - (UPTIME3*256^3+UPTIME2*256^2+UPTIME1*256^1+UPTIME3*256^0) in seconds

4.5. Health check request

Table 8. HEALTH CHECK - STATUS REQUEST (0x115).

Frame type	Flags	Module	Group	D0	D1	D2	D3	D4	D5	D6	D7
0x115	0x0	COMP ID1	COMP ID2	0x01	0xXX	Node Nr	Group Nr	0xXX	0xXX	0xXX	0xXX

0x1150 – HEALTH CHECK REQUEST frame

COMP ID1 - computer identifier (must be unique on the network)
COMP ID2 - computer identifier (must be unique on the network)
0x01 - status request
Node Nr - node number of requested module
Group Nr - group number of requested module
0xXX - inessential data

As response the module will send two frames (Table 7).

Table 9. Response to HEALTH CHECK - STATUS REQUEST (0x115).

Frame type	Flags	Module	Group	D0	D1	D2	D3	D4	D5	D6	D7
0x115	0x1	Node Nr	Group Nr	0x01	RXCNT	TXCNT	RXCNTMX	TXCNTMX	CANINTCNT	RXERRCNT	TXERRCNT

0x1151 – Response to HEALTH CHECK REQUEST frame

Node Nr - node number on the network
Group Nr - group number of the node on the network
0x01 - frame 1 (current values)
RXCNT - current level of receive FIFO buffer
TXCNT - current level of transmit FIFO buffer
RXCNTMX - maximum level of receive FIFO buffer since power up
TXCNTMX - maximum level of transmit FIFO buffer since power up
CANINTCNT - number of CAN interface restarts since power up
RXERRCNT - current receive errors register
TXERRCNT - current transmit errors register

Frame type	Flags	Module	Group	D0	D1	D2	D3	D4	D5	D6	D7
0x115	0x1	Node Nr	Group Nr	0x02	0xFF	0xFF	RXCNTMXE	TXCNTMXE	CANINTCNTE	RXERRCNTE	TXERRCNTE

0x1151 – Response to HEALTH CHECK REQUEST frame

Node Nr - node number on the network
Group Nr - group number of the node on the network

0x02 - frame 2 (maximum values saved in eeprom memory)

RXCNTMXE - maximum ever level of receive FIFO buffer

TXCNTMXE - maximum ever level of transmit FIFO buffer

CANINTCNTE - maximum ever number of CAN interface restarts

RXERRCNTE - maximum ever receive errors

TXERRCNTE - maximum ever transmit errors

To clear maximum values saved in eeprom memory the frame shown in Table 8 must be sent. There is no response to this message.

Table 10. HEALTH CHECK - CLEAR REQUEST (0x115).

Frame type	Flags	Module	Group	D0	D1	D2	D3	D4	D5	D6	D7
0x115	0x0	COMP ID1	COMP ID2	0x02	0xXX	Node Nr	Group Nr	0xXX	0xXX	0xXX	0xXX

0x1150 – HEALTH CHECK REQUEST frame

COMP ID1 - computer identifier (must be unique on the network)
COMP ID2 - computer identifier (must be unique on the network)

0x02 - clear request

Node Nr - node number of requested module

Group Nr - group number of requested module

0xXX - inessential data

5. Module control

The module can be controlled directly from PC, or indirectly by other modules

5.1. Control instruction

The table below shows all instructions, which can be executed by the module. Some of them can be executed only with direct control and other with indirect control (through other modules).

Table 11. Module instructions

Instruction	Instruction Codding								Note	Control	
	INSTR1	INSTR2	INSTR3	INSTR4	INSTR5	INSTR6	INSTR7	INSTR8		Direct	Indirect
LEARN CODE (carrier frequency)	**0x00**	**0x00**	0xXX	0xXX	0xXX	0xXX	0xXX	0xXX	Module waits about 6s for IR signal. When signal is properly received, the module sends CAN message (Table 3) containing carrier frequency. (Needs IR diode connected to the module)	√	
LEARN CODE (code body)	**0x00**	**0x01**	0xXX	0xXX	0xXX	0xXX	0xXX	0xXX	Module waits about 6s for IR signal. When signal is properly received, the module send CAN messages (Table 4) containing learnt IR code. (Needs IR receiver TSOP4838 connected to the module)	√	
SEND LEARNT CODE	**0x01**	**CODE No**	0xXX	0xXX	0xXX	0xXX	0xXX	0xXX	Emits IR code which is saved in module memory. The number of the code 1-100 (0x01-0x64) must be given in INTR2 byte.	√	√
SEND RECEIVED CODE	**0x02**	0xXX	0xXX	0xXX	0xXX	0xXX	0xXX	0xXX	Sends code which was received by other IR receiver and passed over the bus. IR receivers, as control sources, can be defined when linking devices		√
SEND SONY SIRC 12bit	**0x03**	**ADR**	**COMM**	0xXX	0xXX	0xXX	0xXX	0xXX	Sends SIRC 12 bit code (32 addresses, 127 commands). The code address must be given in NSTR2 byte, and command in INSTR3 byte	√	√
SEND SONY SIRC 15bit	**0x04**	**ADR**	**COMM**	0xXX	0xXX	0xXX	0xXX	0xXX	Sends SIRC 15 bit code (256 addresses, 127 commands). The code address must be given in NSTR2 byte, and command in INSTR3 byte	√	√
SEND SONY SIRC 20bit	**0x05**	**ADRH**	**ADRL**	**COMM**	0xXX	0xXX	0xXX	0xXX	Sends SIRC 20 bit code (8192 addresses, 127 commands). The code MSB address must be given in INSTR2 byte, LSB addres in INSTR2 byte and command in INSTR4	√	√
SEND PHILIPS RC5	**0x06**	**ADR**	**COMM**	0xXX	0xXX	0xXX	0xXX	0xXX	Sends Philips RC5 code (32 addresses, 127 commands). The code address must be given in INSTR2 byte, and command in INSTR3 byte	√	√
SEND NEC 16bit	**0x07**	**ADR**	**COMM**	0xXX	0xXX	0xXX	0xXX	0xXX	Sends NEC 16 bit code (256 addresses, 256 commands). The code address must be given in INSTR2 byte, and command in INSTR3 byte	√	√
SEND NEC 24bit	**0x08**	**ADRH**	**ADRL**	**COMM**	0xXX	0xXX	0xXX	0xXX	Sends NEC 24 bit code (65280 addresses, 127 commands). The code MSB address must be given in INSTR2 byte, LSB address in INSTR2 byte and command in INSTR4	√	√
ENABLE BOX	**0xDD**	**BoxX**	**BoxY**	0xXX	0xXX	0xXX	0xXX	0xXX	It enables chosen boxes – these boxes will be compared with next received message from the bus.		√

DISABLE BOX	**0xDE**	**BoxX**	**BoxY**	0xXX	0xXX	0xXX	0xXX	0xXX	It disables chosen boxes – these boxes will be passed when next message arrives from the bus.		√
TOGGLE BOX	**0xDF**	**BoxX**	**BoxY**	0xXX	0xXX	0xXX	0xXX	0xXX	It toggles boxes – enables when they are disabled and vice versa		√

0xXX – inessential data

BoxX	Note
0x00	- from Box 1
0x01	- from Box 2
…	
0x7F	- from Box 128

BoxY	Note
0x00	+ 0 -(and not anyone else)
0x01	+ 1 -(and 1 following)
…	
0x7F	+127 -(and 127 following)

5.2. Direct control

It is possible to control module by sending DIRECT CONTROL message. The message contains instruction, which will be executed by the module.

Table 12. DIRECT CONTROL frame (0x10A).

Frame type	Flags	Module	Group	D0	D1	D2	D3	D4	D5	D6	D7
0x10A	0x0	COMP ID1	COMP ID2	INSTR1	INSTR2	Node Nr	Group Nr	INSTR3	INSTR4	INSTR5	INSTR6

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

INSTR1-6 - instruction to be executed (6 bytes)

5.3. 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 boxes (memory cells).
This firmware has feature to set simple conditions of executing instruction. To do so, you can use blocking instruction (0xDD – 0xDF) shown in the table 11. The HAPCAN Programmer simplifies configuration process.

6. Configuration

Parameters that can be configured with this firmware:
- Module identifier (module number and group number);
- Module description (16 chars);
- Learnt IR codes
- Text notes;
- Linking device with other modules (indirect control of module).

Configuration process can be done using HAPCAN Programmer.

6.1. 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). Identifier of the Ethernet Interface can be changed in HAPCAN Programmer in software settings.

6.2. Module description

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

6.3. Learnt IR code

The firmware has 100 memory cells to store up to 100 learnt or manually entered (in Pronto format) infrared codes. Some IR codes might be longer and they can be saved in max 3 memory cells.

6.4. Text notes.

Up to 1024 characters can be written into processor's memory.

6.5. Linking devices

The module has 128 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.

7. License

8. Document version

File	Note	Date
univ_3-5-0-3a.pdf	Original version	April 2014