# 125 kHz RFID Reader Module

## Internal EEPROM memory
## Storage capacity of 1000 tags

DATASHEET

## Features

- Low-voltage and standard-voltage operation:
  - KA11: VCC= 3V 30mA to 5V 60mA
  - KA11L: VCC= 3V 10mA to 5V 20mA
- 125KHz to 135KHz read frequency
- EM4001, 4102 and TK4001 64-bit RFID tag compatible
- 8 to 12 cm reading range:
  - KA11: 10 to 12cm for standard tags
  - KA11L: 8 to 10cm for standard tags
- Internal EEPROM memory for storage 1000 Tags.
- Adjustable baud rate serial port:
  - 4800, *9600, 14400, 19200, 28800, 38400, 57600, 115200
- Adjustable output data (configurable with serial port)
  1) Optional adjustable Start code [Chr (02) or Chr (58) ": " and more…]
  2) Unique code: [10-digit hex code or (12-digit HEX and XOR code) or *10-digit decimal code printed on tags (0011189641)]
  3) Optional CR code (*chr13) [or Adjustable stop code]
  4) Optional LF code (*chr10) [or Adjustable stop code]
  5) Optional adjustable end code [Chr (03) or Chr (47) "/" and more…]
- Three main card, master card and service cards
- Storage and removal of tags(Main card, Master card, Service card and user cards) via hardware and the serial port
- Main card storage by incoming save pin
  - Save Master card via main card
  - Save service card via master card
  - storage and removal the user cards with the main card and master card
- With special software to setting the device and storage the cards
- Possibility Backup and Restore All cards through the software
- 20 Kbyte EEPROM Memory accessible through the serial port
- Change pins for Internal and External Coil (Antenna)
- Six different operating mode:
  * 1) Working in the internal memory mode (usable in Elevators and home doors)
  2) Working mode in without memory (usable in circuits and Microcontrollers)
  3) Working in without memory and without Relay mode (used in traffic control software and library doors)

---

* Default

4) Working without memory mode - no relays and no beep (used in electronic medical records, parking software, Booking food or electronic-tickets)
5) Working in network mode (with IC RS485) without internal memory and control via a central device or a PC system (used in pools and hotels)
6) Working in network mode (with IC RS485) with internal memory and an independent and non-central (used in pool lockers and hotels)

- Full access the Registry Configuration settings by serial port
- Six- state relay output variable(Selected with the mode pins):

1) Output when the card is present (used in lighting Hotels and start contactor)
2) One second output (used to chain-lock on the door or summon the elevator)
* 3) Three seconds output (usable the no-chain locks on the doors)
4) Five seconds output (used to lock the elevator panels)
5) Eight seconds output (used to lock the elevator panels)
6) Flip-flops output (Security Keys for use in industrial device or Machinery)

- Internal NPN transistor for relay outputs with a current of 300 mA
- Always active and non-active the device with a service card
    - Stored relay Flip-flop Status in the EEPROM (When used the service card)
- Turn on/off relay and beeper by serial port
- Output activator pin for send serial module or wireless transmitter
    - *Before sending data, the going to HI-Level INT-Out pin (8), and send data by TXD (pin3) after 20ms delay and going to Low-Level INT-Out pin after 20ms.
    - Usable to Enable pins on the HMTR TTL - HMTRP - DRF7020-D27, …
    - Usable in Microcontroller interrupts
- Incoming pin for alarms and alerts via serial port
    - Used in warning switch on box or Magnetic sensors on the Hotel doors
- 640 Hz sounder output via NPN transistor (open collector)
- Changeable Exclusive Device ID in RS485 serial network
- 27-bytes Serial Data Bus for RS485 Network with proprietary format
- Package dimensions 6cm * 3cm * 0.9cm

## Description

The KA-11 Kartach RFID reader is an advanced reader for the popular 64-bits EM4001 and EM4102 format 125 KHz to 135 KHz tags. Read ranges of over 12 cm are possible with our external antenna. Advanced features include register setting and control relay or beeper state, storage and removal (main card -master card -service card and user cards), and setting baud-rate or unique data format via serial port. Suitable for home doors, Electronic locker pool, Hotel doors, Attendance device, Electronic tickets, Reservation food, The building elevators, Safety lock devices and industrial machinery, Persons or students traffic control, Electronic software and electronic records, Electronic medical records, Institutional and industrial projects.

# Table of Contents

# Kartach KA-11 / KA-11L Series

Kartach
KA-11
-125khz RFID Reader Module
-with Internal memory for 1000 tags
-power supply: 3v <30mA , 5v <60mA

Package dimensions: 6cm x 3cm x 0.9cm

## 1. Block diagram

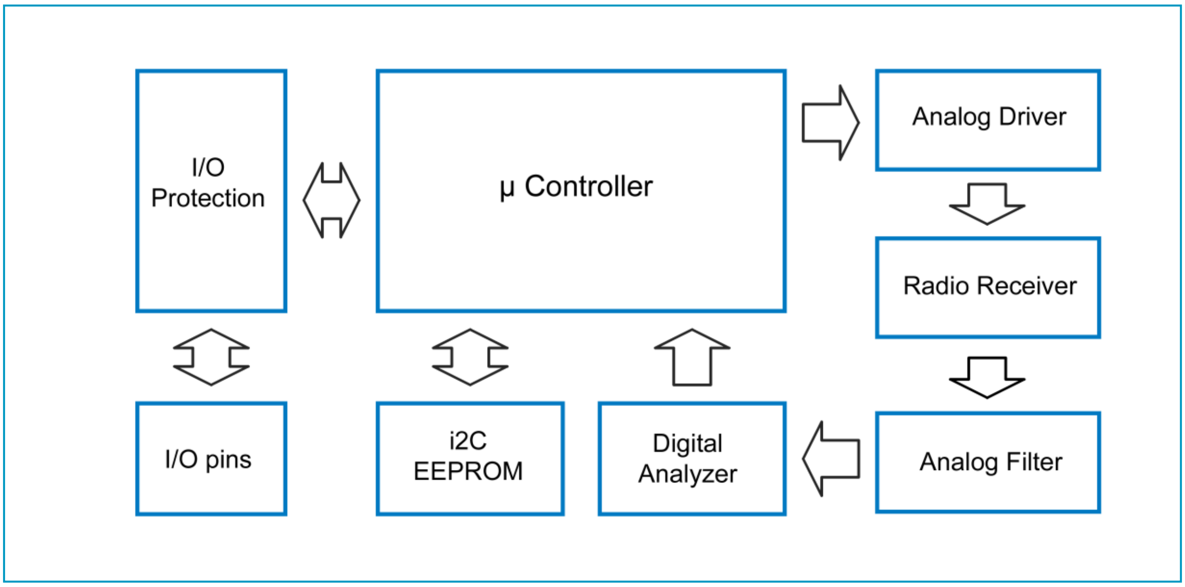


Figure1. Block diagram

## 2. I/O Pins Configuration

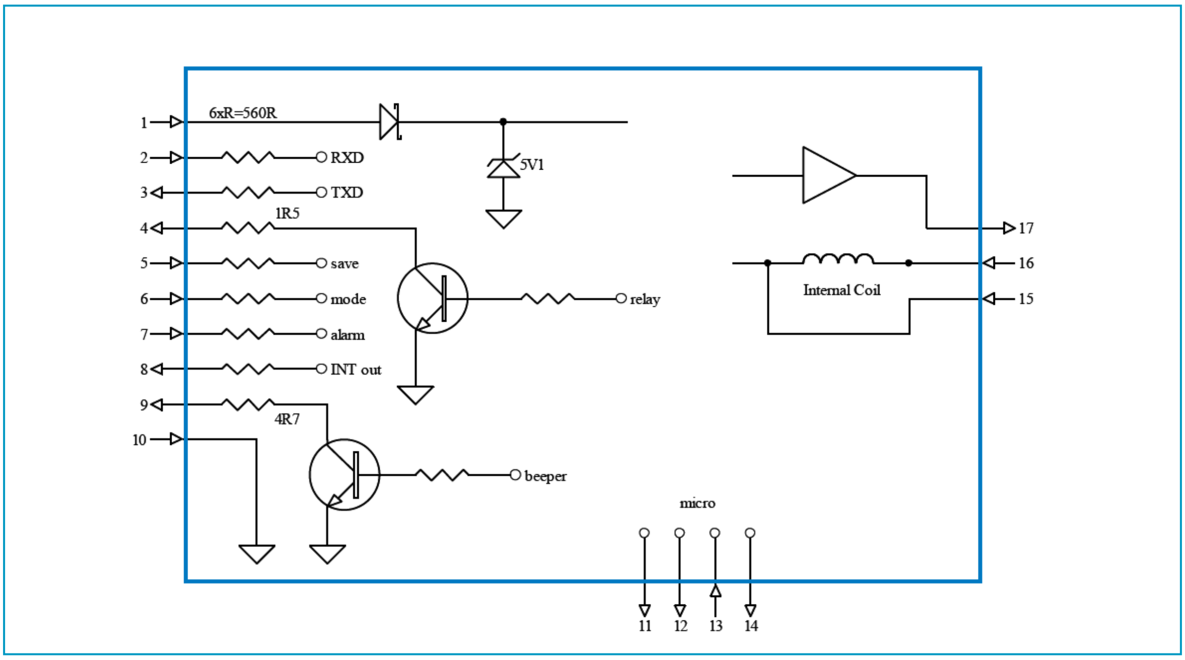


Figure2. Pins Configuration (front view)

- Not to connect the pins to more than 5 volts.
- To protect the Transistor you must use between Relay output and Relay coil one 22Ω Resistor.

## 3. Device I/O pins

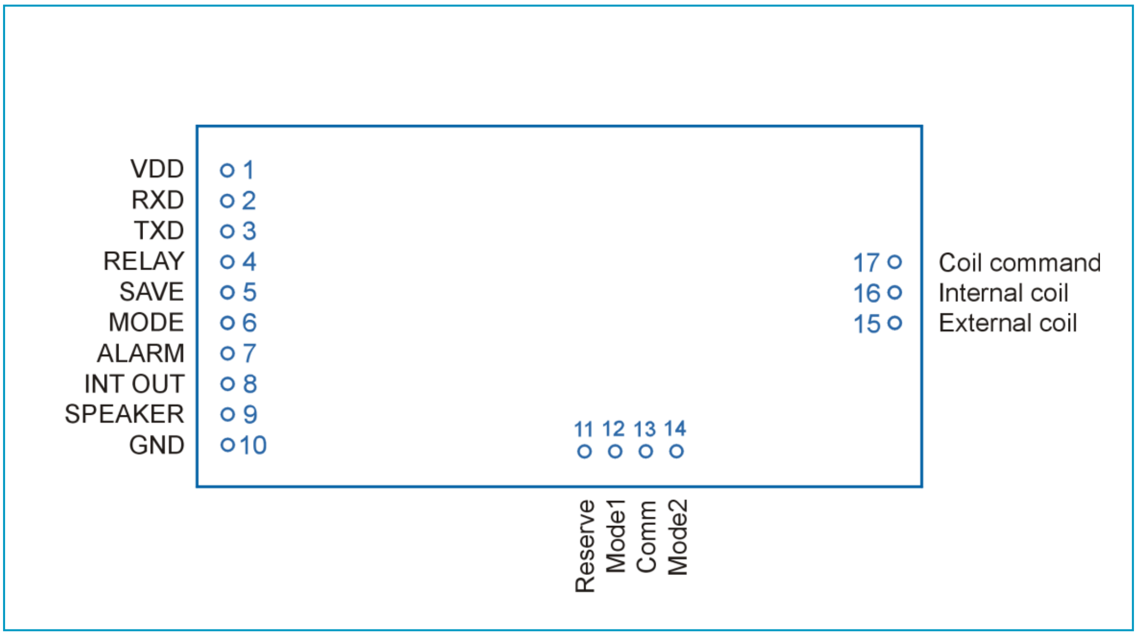


Figure3. Device pins (front view)

## 4. Pin Descriptions

| no | Pin name | Descriptions |
|---|---|---|
| 1 | VDD | Power supply voltage 3V to 5.3V (With internal Zener diode 5.1V) |
| 2 | RXD | Receive external data from micro controller or PC system a TTL standard with variable baud rate, non parity and 1 stop bit. |
| 3 | TXD | Transmitter data for external micro controller or PC system a TTL standard with variable baud rate, non parity and 1 stop bit. |
| 4 | Relay | negative Relay output via internal NPN transistor with a current of 300mA (See Figure 2) for used six-status output. |
| 5 | Save | For storage main-card pull the save-pin in ground voltage and waiting for Two-seconds. |
| 6 | Mode | If this pin is the ground voltage, relay selector pins (12,13,14) is the dual mode |
| 7 | Alarm | If the device works the network mode in the hotel or pool (modes 5 or 6), if this pin connected to ground voltage, send alert with a exclusive ID to the central system. |
| 8 | INT out | To enable the interrupt microcontroller or enable pin on the IC-RS485 and wireless data transmitter activation |
| 9 | Speaker | Speaker output - frequency of 640Hz with open collector NPN transistor. |
| 10 | GND | Power Ground |
| 11 | Reserve | Not use (reserved) |
| 12 | Mode1 | Selector1 (Relay output mode) connect to command-pin13 (*see Table 2*) |
| 13 | Command | Command Mods (connect to mode1 or mode2 pins) |
| 14 | Mode2 | Selector2 (Relay output mode) connect to command-pin13 (*see Table 2*) |
| 15 | External | External antenna input pin |
| 16 | Internal | Internal antenna input pin |
| 17 | Coil out | Coil output - if this pin is connected to pin 16 with a jumper (you can use the internal antenna), or you can connect this pin to the pin 17 through the external antenna, you can use the external antenna *(see Figure 5)* |

Table1 (Pin Descriptions)

## 5. Hardware-based commissioning

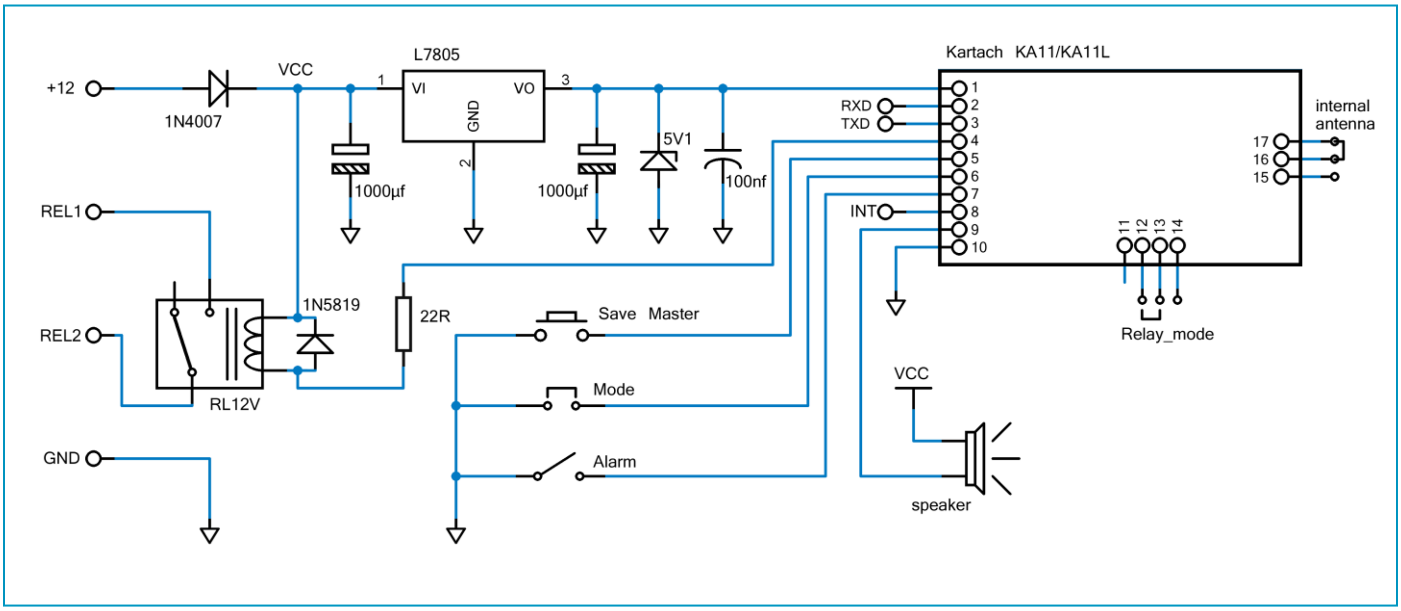


Figure4. Commissioning circuit

## 6. Save Main card

- Hold for two seconds in ground voltage the save pin (5), after hearing the little beep repeatedly place a Tag to front of the module, saved it as a tag for Main-card.
- For storage this cards via serial command*, see Table3 (row4)*.

## 7. Save Master card

- Keep your Maim-card in front of the module, after hearing the 10-small beep, make a continuous beep for 1sec. After the beep and put a tag in front the module, this tag is known as Master-card.
- For storage this cards via serial command, *see Table3 (row6)*.

## 8. Save Service card

- Keep the Master-card in front of the module, after hearing the 10-small beep, make a continuous beep for 1sec. After the beep and put a tag in front the module, this tag is known as Service-card.
- Service Card for permanent on/off relays is used. (Usage on the elevator panel). If the service card is used, the relay status is stored in the EEPROM. (With disconnect and reconnect voltage, the relay status is maintained.)
- For storage this cards via serial command, *see Table3 (row8)*.

## 9. Save User cards.

- Once the main card or the MasterCard card at the front of the module, small-beep continuous in speaker (with period of 300 ms). When you put a tag in front of the module, if the tag has not been previously saved, will be saved and will be deleted if it is stored.
- For storage this cards via serial command, *see Table3 (row10)*.

## 10. External or internal antenna

- If pin16 is connected to pin17 with a jumper, (you can use the internal antenna). Or you connect pin15 to the pin16 through the external antenna; you can use the external antenna.

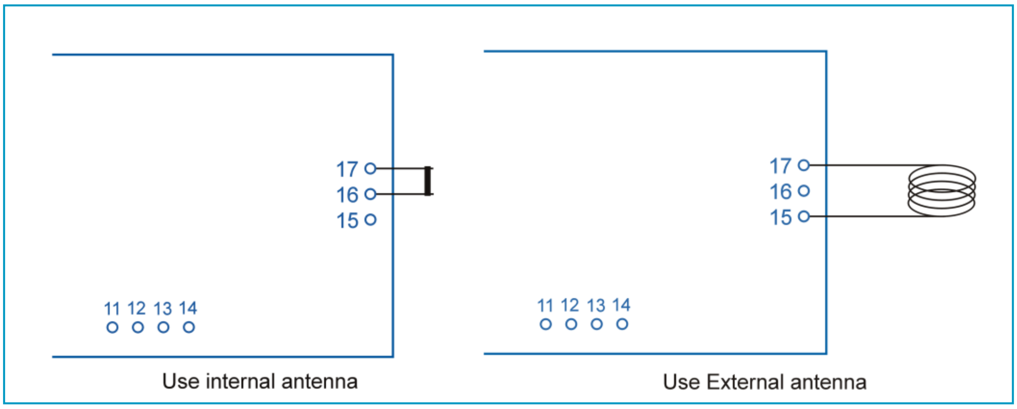


Figure5. External or internal Antenna (front view)

- Variety of external antennas

**Size: 6cm x 3cm**
**Read range: 12cm**

**Size: 3cm x 3cm**
**Read range: 11cm**

**Size: 2cm x 3cm**
**Read range: 9cm**

**Size: 1cm x 3cm**
**Read range: 8cm**

**Diameter: 2.4cm**
**Read range: 8cm**

## 11. Select Relay time Modes

- If the mode pin(6) is connected to ground voltage, you choose the latter modes (*see Table2*).

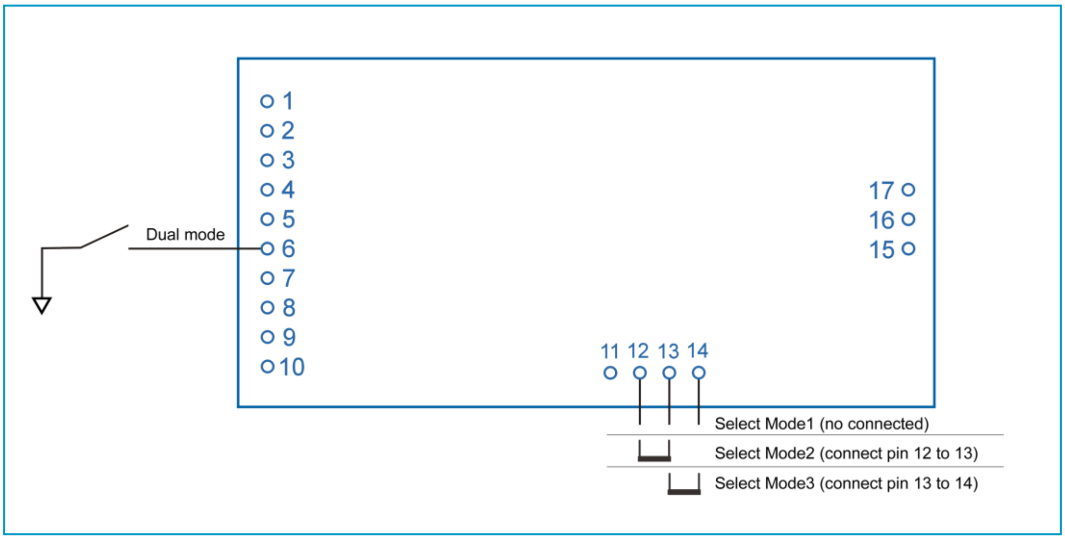

1
2
3
4
5
6
7
8
9
10
Dual mode
17
16
15
11 12 13 14
Select Mode1 (no connected)
Select Mode2 (connect pin 12 to 13)
Select Mode3 (connect pin 13 to 14)

Figure6. Relay time mode pins (front view)

## 12. Select Relay Modes table.

| Mode | Mode pin (pin 6) | Output Relay | Select mode pins (12,13,14) |
|---|---|---|---|
| 1 | NC | Latch (card presence) | No Connected |
| 2 | NC | 1 second (Electronic lock) | Connect pin12 to pin13 |
| 3 | NC | 3 second (Electronic lock) | Connect pin13 to pin14 |
| 4 | GND | Flip-flop (switch) | No Connected |
| 5 | GND | 5 second (elevator) | Connect pin12 to pin13 |
| 6 | GND | 8 second (elevator) | Connect pin13 to pin14 |

Table2. (Relay output modes)

## 13. Relay Output Hardware

- If the relay output used to inductive circuits, you must be used protection diode for feedback voltage.

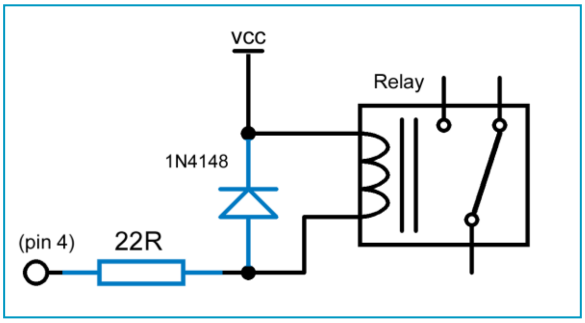


Figure7. Relay feedback Diode (front view)

- If you want use the relay to activate a second relay, if the second relay is DC voltage, the second relay coil is connected to a feedback diode.

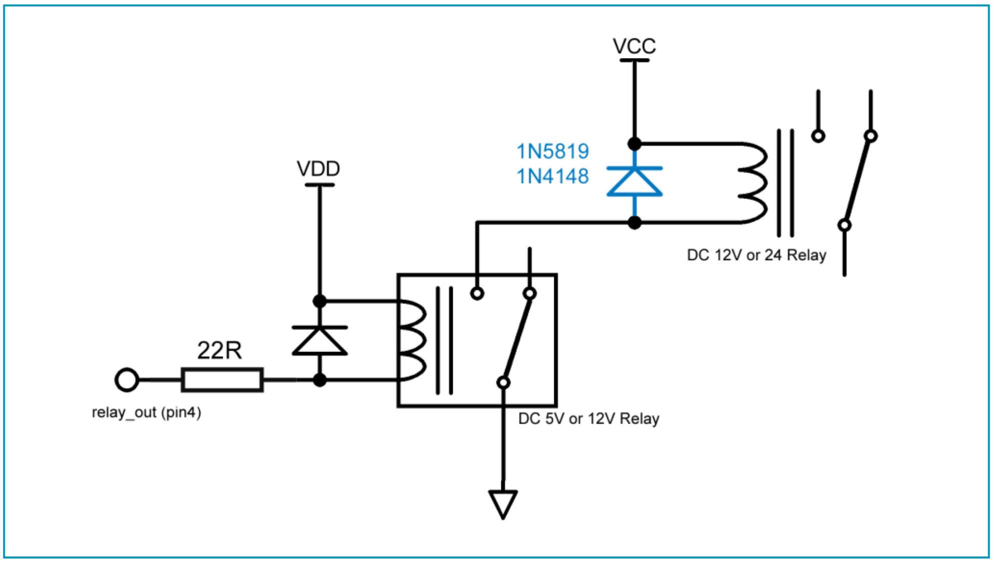


Figure8. Second Relay feedback Diode (DC relay)

- If second relay is induction load, contactor, AC motor or hydraulic / pneumatic electrical switches (AC inductive circuit), you must use a catcher spark circuit in the NC-contact first relay.

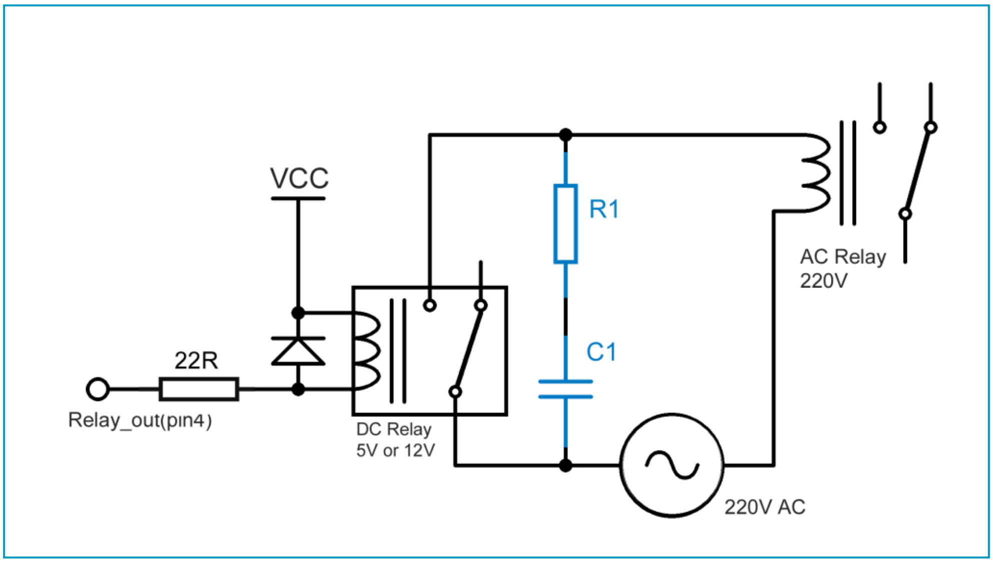


Figure9. Second Relay Snubber circuit (AC relay)

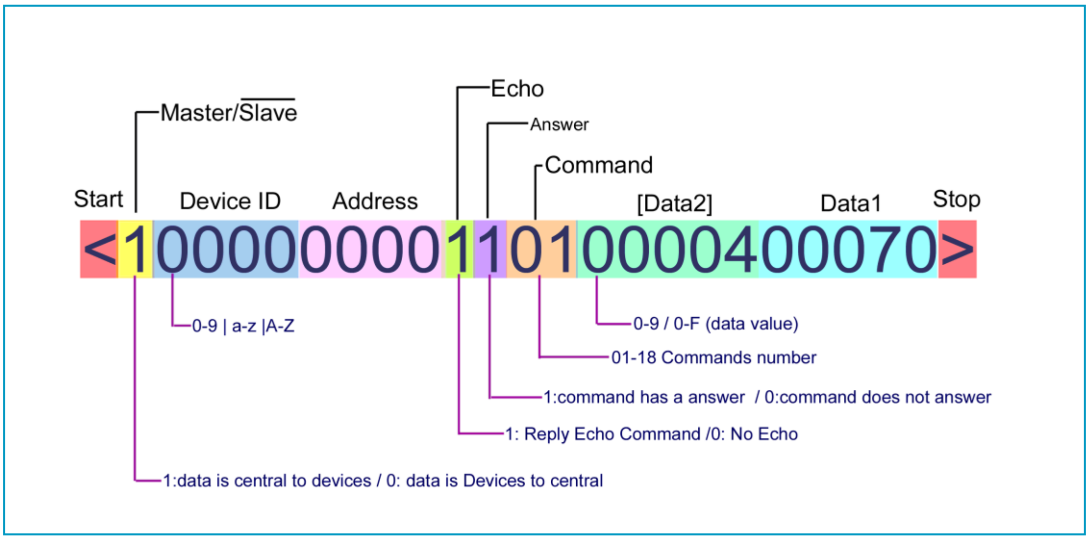
C1 is MKT, polyester or ceramic capacitor, 250V - 47n and R1 is resistor 2watt, 470Ω to 4/7KΩ (depending on the induction load, If a large induction coil resistance should be less.

- If the relay load is a lamp, this circuit is not necessary.

## 14. Serial Port Commands.

- The default serial port settings is "9600,8,n,1". Message length is 27-bytes, 22-bytes data, 2-bytes start(chr60) and stop(chr61), 2-bytes(End characters at the enter(chr13) and end (chr10) must be used CR+LF).

<10000000011010000400070> + (chr13) + (chr10)


Master/$\overline{\text{Slave}}$
Echo
Answer
Command
Start
Device ID
Address
[Data2]
Data1
Stop
<10000000011010000400070>
0-9 | a-z |A-Z
0-9 / 0-F (data value)
01-18 Commands number
1:command has a answer / 0:command does not answer
1: Reply Echo Command /0: No Echo
1:data is central to devices / 0: data is Devices to central

Command data Format

## 15. Commands table1 (send from PC to module).

| No | Command | Sample | Results |
|---|---|---|---|
| 1 | **Set Buzzer** | < 1 0000 0000 1 0 01 00004 00120 > | 4 beep for 120 ms duration<br>data1: 001 to 999 data2: 0001 to 9999 |
| 2 | **Set relay** | < 1 0000 0000 1 0 02 0000000053 > | Relay to turn-on for 53 seconds<br>data = 0000 to 9999 |
| 3 | **Relay Off** | < 1 0000 0000 1 0 03 0000000000 > | Turn off relay |
| 4 | **Save Main-Card** | < 1 0000 0000 1 1 04 0006119354 > | Save number 0006119354 for Main |
| 5 | **Remove Main-Card** | < 1 0000 0000 1 1 05 0000000000 > | Remove Main card |
| 6 | **Save Master-Card** | < 1 0000 0000 1 1 06 0006119354 > | Save number 0006119354 for Master |
| 7 | **Remove Master-Card** | < 1 0000 0000 1 1 07 0000000000 > | Remove Master card |
| 8 | **Save Service-Card** | < 1 0000 0000 1 1 08 0006119354 > | Save number 0006119354 for Service |
| 9 | **Remove Service-Card** | < 1 0000 0000 1 1 09 0000000000 > | Remove Service card |
| 10 | **Save User-Card** | < 1 0000 0000 1 1 10 0006119354 > | Save number 0006119354 for User |
| 11 | **Remove User-Card** | < 1 0000 0000 1 1 11 0000000000 > | Remove User card |
| 12 | **Clear user-Cards** | < 1 0000 0000 1 0 12 0000000000 > | Removal of all user cards |
| 13 | **Select mode** | < 1 0000 0000 1 0 13 0000000006 > | Select the device working mode: 6<br>data= 1 to 6 |
| 14 | **Write to EEPROM** | < 1 0000 0000 1 0 14 20000 00255 > | Write 255 to EEPROM at address 20000<br>data1= 00001 to 20000 data2= 000 to 255 |
| 15 | **Read from EEPROM** | < 1 0000 0000 0 1 15 01483 00015 > | Reading 15 byte from the address 1483<br>data1= 00001 to 20000 data2= 000 to255 |
| 16 | **Set serial Baud-rate. 9600=11059200/16/ _ 9600 – 1 = 71** | < 1 0000 0000 1 0 16 0000000071 > | Set the baud rate to 9600<br>For Example: 4800=143; 9600 = 071; 14400=047; 19200=035; 28800=023; 38400=017; 57600=011; 115200=005 |
| 17 | **Set data format** | < 1 0000 0000 1 0 17 0288131003 > | Set data out for hex: 1D0053C4A3<br>02=' '; 03=' '; 88=10-Digit-HEX; 72= XOR 85=10Digit-Decimal unique code printed on the tag. 58=':' , all ASCII character |
| 18 | **Set device ID** | < 1 0000 0000 1 0 18 0000001234 > | Set device ID to 1234 id is 0000 to 9999<br>When you use a shared ID'0000' in the network, Echo-bit and answer-bit must be 0. |
| 19 | **Write card to address** | < 1 0000 0637 1 1 19 0006119354 > | Write the 0006119354 tag for user card at address 637 address=0000 to 1000 |
| 20 | **Read card from Addr.** | < 1 0000 0002 1 1 20 0000000000 > | Get Card number stored in the Address: 2<br>address=0000 to 1000 |

Table3. (Commands from PC to the device)

- If the Master/Slave-bit (bit 1) is '1', the message is sent from central device or PC to the module. Or if the Master/Slave-bit is '0', this message is sent from module to central device or PC.
- To use all features of the device, you must select mode 6 in the module.
- If a command requires a result, the value of Answer-bit should be 1.
- If you want to be sure that message delivery was successful, the Echo-bit should be 1.
- If you using multiple modules in network with IC RS485, Device ID not should be used common id '0000'.
- If use a common command in the network, you should be Answer-bit and Echo-bit value is set to 0.

## 16. Commands table2 (send from module to PC).

| No | Command | Sample | Results |
|---|---|---|---|
| 1 | **Set Buzzer** | < 0 0000 0000 1 0 01 00004 00120 > | Buzzer command was successfully |
| 2 | **Set relay** | < 0 0000 0000 1 0 02 0000000053 > | Turn on the Relay for 53 seconds. |
| 3 | **Relay Off** | < 0 0000 0000 1 0 03 0000000000 > | Relay OFF successfully |
| 4 | **Save Main-Card** | < 0 0000 0000 0 1 04 0006119354 > | Saved the 0006119354 card to main-card |
| 5 | **Remove Main-Card** | < 0 0000 0000 0 1 05 0000000000 > | Main-card was successfully deleted |
| 6 | **Save Master-Card** | < 0 0000 0000 0 1 06 0006119354 > | Saved the 0006119354 card to Master |
| 7 | **Remove Master-Card** | < 0 0000 0000 0 1 07 0000000000 > | Master-card was successfully deleted |
| 8 | **Save Service-Card** | < 0 0000 0000 0 1 08 0006119354 > | Saved the 0006119354 card to Service |
| 9 | **Remove Service-Card** | < 0 0000 0000 0 1 09 0000000000 > | Service-card was successfully deleted |
| 10 | **Save User-Card** | < 0 0000 0000 0 1 10 0006119354 > | Saved the 0006119354 card for Users |
| 11 | **Remove User-Card** | < 0 0000 0000 0 1 11 0000000000 > | Main card successfully deleted |
| 12 | **Clear user-Cards** | < 0 0000 0000 1 0 12 0000000000 > | All user cards successfully deleted |
| 13 | **Select mode** | < 0 0000 0000 1 0 13 0000000006 > | Operate mode6 of the device was Selected |
| 14 | **Write to EEPROM** | < 0 0000 0000 1 0 14 20000 00255 > | Write 255 to EEPROM at address 20000 |
| 15 | **Read from EEPROM** | < 0 0000 0000 0 1 15 01483 00015 > | Reading 15 byte from the address 1483 |
| 16 | **Set serial Baud-rate** | < 0 0000 0000 1 0 16 0000000071 > | Set the baud rate to 9600 |
| 17 | **Set data format** | < 0 0000 0000 1 0 17 0288131003 > | Set data out for hex: 1D0053C4A3 |
| 18 | **Set device ID** | < 0 0000 0000 1 0 18 0000001234 > | Set device ID to 1234 id is 0000 to 9999 |
| 19 | **Write card to address** | < 0 0000 0637 1 0 19 0006119354 > | Write the 0006119354 tag for user @ 637 |
| 20 | **Read card from Addr.** | < 0 0000 0002 0 1 20 0000000000 > | Get Card number saved in the Address 2 |
| 21 | **Placing user card** | < 0 0000 0000 0 1 21 0006119354 > | Placed a user card to front the module (mode6) |
| 22 | **Placing Unidentified card** | < 0 0000 0000 0 1 22 0006119354 > | Placed a Unidentified card to front the module |
| 23 | **Pacing cards** | < 0 0000 0000 0 1 23 0006119354 > | Placed a card to front the module (mode5) |
| 24 | **Going to save mode** | < 0 0000 0000 0 1 24 0006119354 > | Goto saving mode with the Main card |
| 25 | **Going to save mode** | < 0 0000 0000 0 1 25 0006119354 > | Goto saving mode with the Master card |
| 26 | **Select service mode** | < 0 0000 0000 0 1 26 0006119354 > | Select service mode with the Service card |
| 27 | **Saved user card** | < 0 0000 0000 0 1 27 0006119354 > | The card saved to memory (Manually) |
| 28 | **Deleted user card** | < 0 0000 0000 0 1 28 0006119354 > | The card was removed from memory. (Manually) |
| 29 | **Alarm input** | < 0 1234 0000 0 1 29 0000000000 > | Alarm Input (pin 6) in the Module was enabled. |
| 30 | **Card is stored** | < 0 0000 0000 0 1 30 0006119354 > | This card is already saved. |
| 31 | **This card was not found** | < 0 0000 0000 0 1 31 0006119354 > | This card was not found for removal. |
| 32 | **This Card is services** | < 0 0000 0000 0 1 32 0006119354 > | This card is one of the services cards. |
| 33 | **This Card is Users** | < 0 0000 0000 0 1 33 0006119354 > | This card is one of the user cards. |
| 34 | **Saved main card** | < 0 0000 0000 0 1 34 0006119354 > | Saved the card for main. (Manually) |
| 35 | **Saved master card** | < 0 0000 0000 0 1 35 0006119354 > | Saved the card for master. (Manually) |
| 36 | **Saved service card** | < 0 0000 0000 0 1 36 0006119354 > | Saved the card for service. (Manually) |

Table4. (Command from the Device to PC)

- If in Command 20 the reserve value is 3000, this message is warning for read completed.

## 17. Sample commands for serial port settings and tag storage.

After a message, must be use the following CR+LF command (chr13, chr10).

### 1) Buzzer Command.

- Once beep of 400 ms duration.
  `<10000000010010000100400>+Chr(13) +Chr(10)`

- Four beep of 50 ms duration.
  `<10000000010010000400050>+Chr(13) +Chr(10)`

### 2) Relay Control.

- Relay to turn-on for 2 seconds.
  `<10000000010020000000002>+Chr(13) +Chr(10)`

- Relay permanent clarify.
  `<10000000010020000000000>+Chr(13) +Chr(10)`

### 3) Relay Off.

- Turn off relay.
  `<10000000010030000000000>+Chr(13) +Chr(10)`

### 4) Save Main-Card.

- Save the Tag 0005944374 for the Main card. This code is 10-digit number printed on the tag or card.
- If the answer bit is 1, the result of the operation is returned.
  `<10000000011040005944374>+Chr(13) +Chr(10)`

### 5) Remove Main-Card.

- This command will remove the Main Card.
  `<10000000011050000000000>+Chr(13) +Chr(10)`

### 6) Save Master-Card.

- Save the Tag 0005944374 for the Master card.
  `<10000000011060005944374>+Chr(13) +Chr(10)`

### 7) Remove Master-Card.

- This command will remove the Master Card.
  `<10000000011070000000000>+Chr(13) +Chr(10)`

### 8) Save Service-Card.

- Save the Tag 0005944374 for the Service card.
  `<10000000011080005944374>+Chr(13) +Chr(10)`

### 9) Remove Service-Card.

- This command will remove the Service Card.
  `<10000000011090000000000>+Chr(13) +Chr(10)`

### 10) Save User-Card.

- Save the Tag 0005944374 for the User Cards.
  `<10000000011100005944374>+Chr(13) +Chr(10)`

## 11) Remove User-Card.

- This command will remove the User Card.
  <10000000011110000000000>+Chr(13) +Chr(10)

## 12) Clear User-Cards.

- This command will clear all User Cards.
  <10000000010120000000000>+Chr(13) +Chr(10)

## 13) Select the device working mode. (default mode: 1)

- This device has 6 operating modes are different.
- The device is operating in mode 1 (internal EEPROM and Master Card...).
  <10000000010130000000001>+Chr(13) +Chr(10)

- The device is operating in mode 1 (No internal EEPROM or Master Card).
  <10000000010130000000001>+Chr(13) +Chr(10)

## 14) Save data to the device EEPROM (20-Kbyte accessible internal EEPROM).

- Saving byte (205) to the address (1350).
  <10000000010140135000205>+Chr(13) +Chr(10)

## 15) Read data from the device EEPROM.

- Reading 8 bytes of address (23).
  <10000000011150002300006>+Chr(13) +Chr(10)
  Answer this command is:
  <00000000001154002300255>+Chr(13) +Chr(10)
  <00000000001150002400007>+Chr(13) +Chr(10)
  <00000000001150002500255>+Chr(13) +Chr(10)
  <00000000001150002600018>+Chr(13) +Chr(10)
  <00000000001150002700107>+Chr(13) +Chr(10)
  <00000000001150002800255>+Chr(13) +Chr(10)

## 16) Set baud rate of the serial port. (default: 9600)

- Set the baud rate to 19200.
  <10000000010160000000035>+Chr(13) +Chr(10)
  35 = 11059200 / 19200 / 16 - 1

- Set the baud rate to 9600.
  <10000000010160000000071>+Chr(13) +Chr(10)
  71 = 11059200 / 9600 / 16 - 1

Note. After changing the Baud rate, you must send the next message with the new baud rate.

## 17) Output data setting. (default output data: 0011021576 [85,13,10,00,00] )

Data exclusive naming:
88: 10-Digit Card HEX Code.
85: 10-Digit Card ASCII Code (printed on the tags)
72: 1-Byte XOR code
13: CR
10: LF
58: ':' All Characters ASCII Code. For example: chr(40) is '(' or chr(64) is '@' or chr(36) is '$' .

- Set data output to the 10-digit unique code printed on the tag [0006130464]. (No enter-code and end-code).
  <10000000010178500000000>+Chr(13) +Chr(10)
  Data output for tag number 4 reads:
  0006130464000613046400061304640006130464

- Set data output to the 10-digit unique code printed on the tag [0006130464]. (with CR+LF code).
  <1000000001017851310000>+Chr(13) +Chr(10)
  Data output for tag number 4 reads:
  0006130464
  0006130464
  0006130464

- Set data output to the 10-digit unique code printed on the tag [0006130464]. (with CR Enter code).
  <1000000001017851300000>+Chr(13) +Chr(10)
  Data output for tag number 4 reads:
  0006130464
  0006130464
  0006130464

- Set data output to the 10-digit HEX code [1E005D8B20] and CR+LF enter and end code.
  <1000000001017881310000>+Chr(13) +Chr(10)
  Data output for tag number 4 reads:
  1E005D8B20
  1E005D8B20
  1E005D8B20

- Set data output to the 10-digit HEX code [1E005D8B20] and XOR code (EM-18 RFID reader module data).
  <1000000001017887200000>+Chr(13) +Chr(10)
  Data output for tag number 4 reads:
  1E005D8B20E81E005D8B20E81E005D8B20E81E005D8B20

- Set data output to standard RFID code (RF01D id3, RF01D mango, ID-12, ID-20 data).
  Start (chr02) + 10-digit HEX code + CR enter code (chr13) + LF end code (chr10) + stop (chr03).
  <1000000001017028813100З>+Chr(13) +Chr(10)
  Data output for tag number 9 reads:
  ▯1E005D8B20
  ▯▯1E005D8B20
  ▯▯1E005D8B20

---

## 18)Set the device ID in the Network RS485. (default ID is 0000)

For send a command to the modules, ID (0000) is acceptable for all ID.

- When you use a shared ID'0000' in the network, Echo-bit and answer-bit should be '0'.
  <1000000000020000000002>

- Set the module ID to '1234 '.
  <1000000001018000000 1234>+Chr(13) +Chr(10)

- Save the tag 0005944374 for User-card in the Device ID code is 1234.
  <1123400001100005944374>+Chr(13) +Chr(10)

- 2 seconds turn-on relay on the module in network, the Device ID code is 1234.
  <1123400001102000000002>+Chr(13) +Chr(10)

- Set the module ID code from 1234 to 2563.
  <1123400001118000000 2563>+Chr(13) +Chr(10)

---

## 19) Write card to address

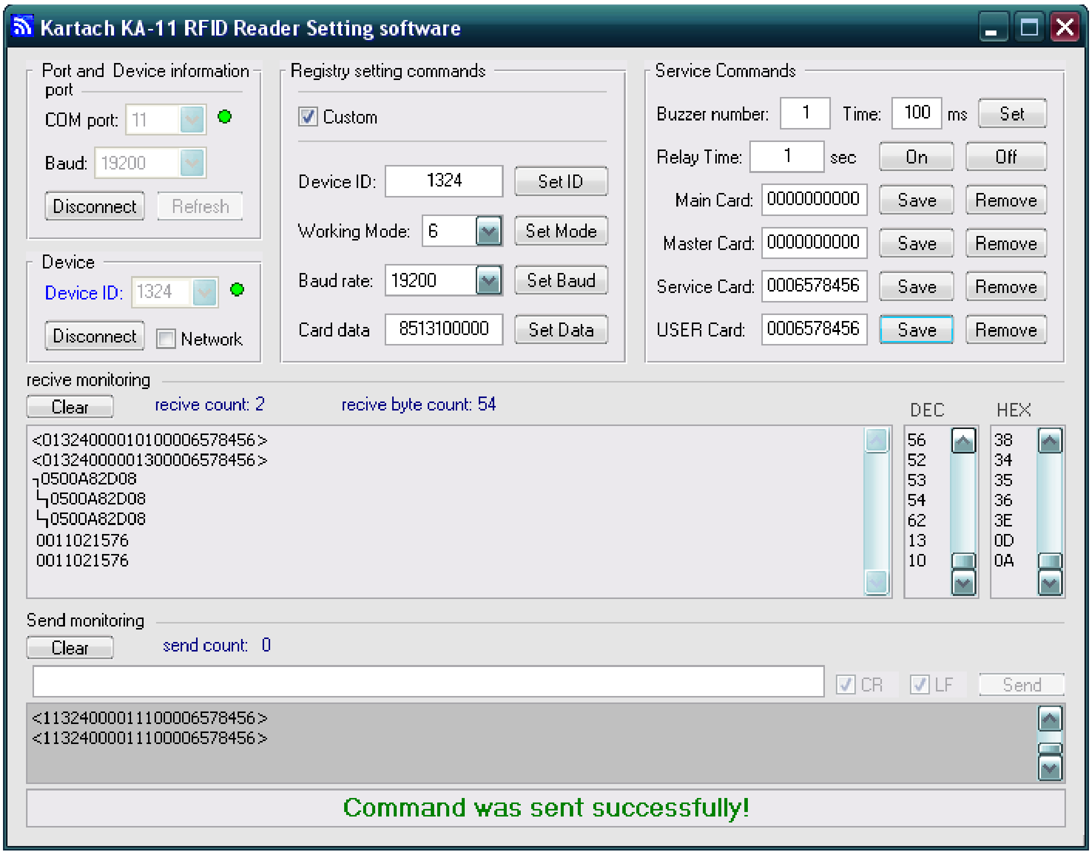
- Write the 0006119354 Tag for user card at address 637 (address=637 to 1000).
  <1000006370119000 6119354 >+Chr(13) +Chr(10)

---

## 20) Read card from address

- Get Card number stored in the Address 2.
  <1000000201200000000000 >+Chr(13) +Chr(10)
  Get Card number stored in the Address: 2 address=0000 to 1000

## 18. Device Configuration Software.



Kartach KA-11 RFID Reader Setting software

Port and Device information port

COM port: 11

Baud: 19200

Device

Device ID: 1324

☐ Network

Registry setting commands

☑ Custom

Device ID: 1324

Working Mode: 6

Baud rate: 19200

Card data 8513100000

Service Commands

Buzzer number: 1 Time: 100 ms

Relay Time: 1 sec

Main Card: 0000000000

Master Card: 0000000000

Service Card: 0006578456

USER Card: 0006578456

recive monitoring

recive count: 2

recive byte count: 54

```
<01324000010100006578456>
<01324000001300006578456>
┐0500A82D08
└0500A82D08
└0500A82D08
0011021576
0011021576
```

| DEC | HEX |
|---|---|
| 56 | 38 |
| 52 | 34 |
| 53 | 35 |
| 54 | 36 |
| 62 | 3E |
| 13 | 0D |
| 10 | 0A |

Send monitoring

send count: 0

☑ CR ☑ LF

```
<11324000011100006578456>
<11324000011100006578456>
```

Command was sent successfully!

Software for the Device Configuration or Storage and Removal Cards

## 19. PC hardware connections.

- For using an RS232 COM port is required a MAX232-IC. Capacitors should be connected in the over long distances.

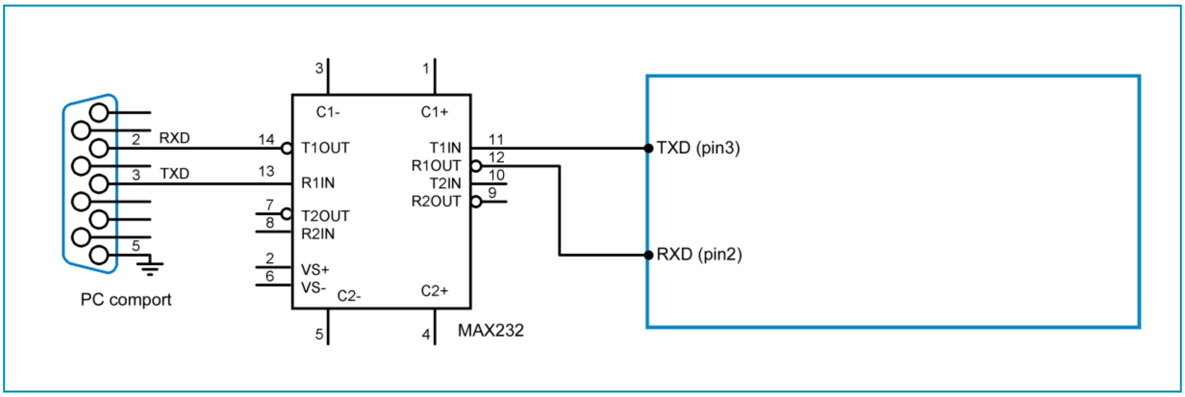


Figure10. Connect To PC via PC via Comport

- The use of a PC USB ports, a converter IC (USB to Serial-TTL) is required. In the circuit one 1000µf capacitor on the power supply (VDD and GND) should be used.

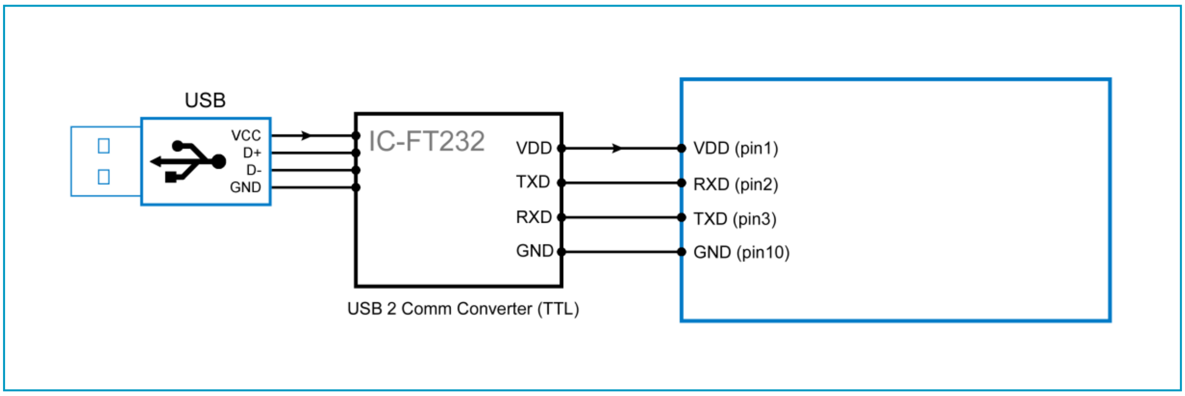


Figure11. Connect to PC via USB port

- RS485 Transmission protocol for long distances.

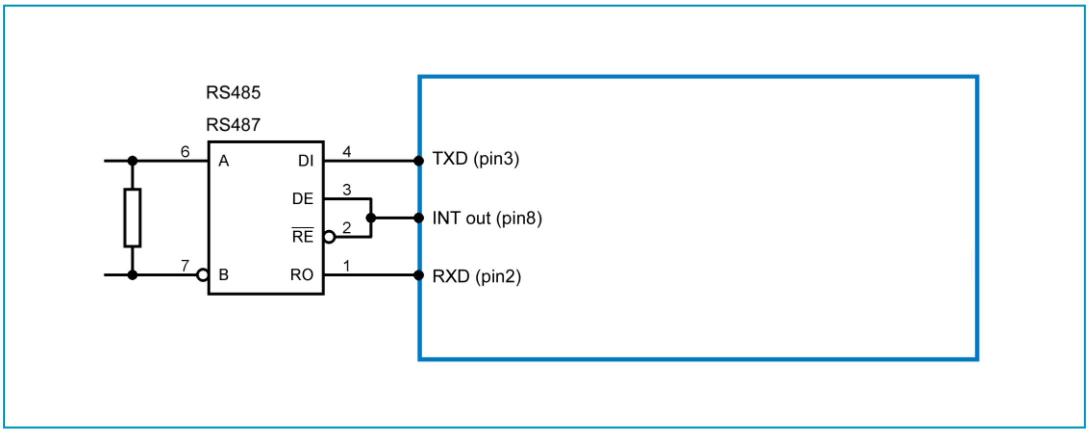


Figure12. Support of IC RS422/RS485/ MAX485/ MAX487

- Network these modules through the rs485 protocol to use in the hotels and swimming pools.

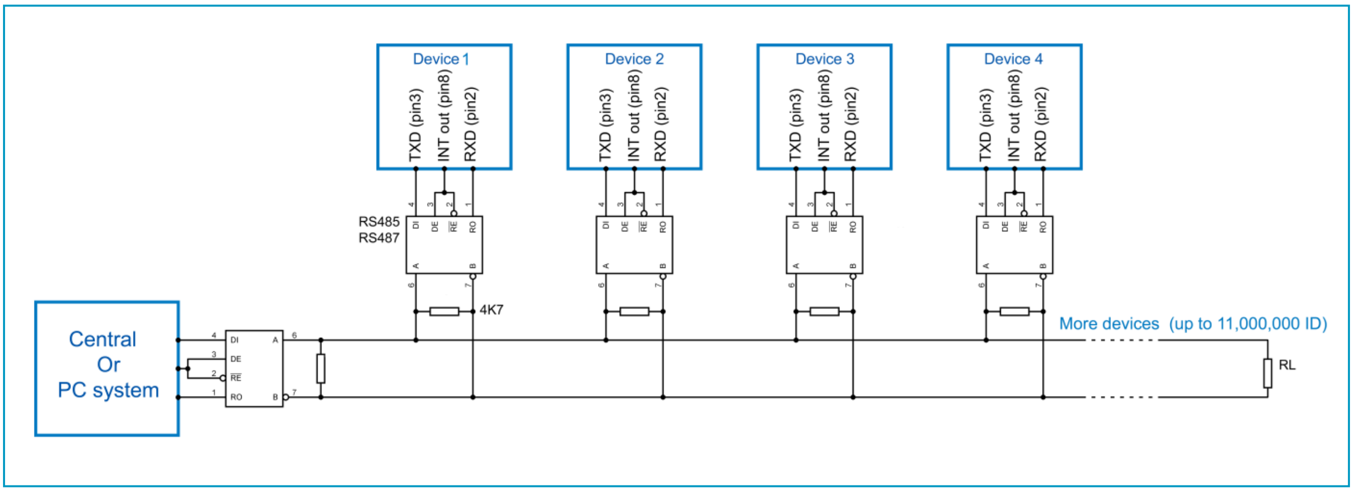


Figure13. Network with rs485 protocol

## 20. Example for Send Commands with several type programming environments.

- Example for BASCOM AVR compiler: *(Download source)*

```
'Save a tag for a user-card in the network on the module ID is '0016'.

   $regfile = "M8DEF.dat"
   $crystal = 11059200
   $baud = 9600

   Dim _data As String * 10 , _id As String * 4 , _command As String * 2

   '0016 is device id, 04 is save command, 0005944374 is tag unique code.
   _id = "0016" : _command = "04" : _data = "0005944374"

   Print "<1" ; _id ; "000011" ; _command; _data ; ">"
   'send data is: <10016000011040005944374>
   Wait 1

   'Or
   Print "<10016000011040005944374>"

   Do : Loop

End
```

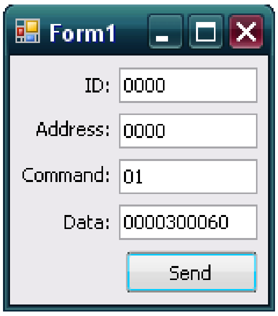
Schedule1. For example, sending data with BASCOM-AVR

- Example for Codevision AVR compiler: *(Download source)*

```
//Save a tag for a user-card in the network on the module ID is '0016'

#include <mega8.h>
#include <stdio.h>
#include <string.h>
#include <delay.h>
//04 is save command, 0005944374 is tag unique code.
char _id[]="0000", _command[]="04", _data[]="0005944374";

void main(void){

//USART setting (crystal:11059200Hz)
UCSRA=0x00; UCSRB=0x08; UCSRC=0x86; UBRRH=0x00; UBRRL=0x47;

//0016 is device id,
strcpyf(_id,"0016");

while (1){

    printf("<1%s000011%s%s>\r\n",_id,_command,_data);
    delay_ms(1000);

    //Or
    printf("<10016000011040005944374>\r\n");

    for(;;);
};

}
```

Schedule2. For example, sending data with BASCOM-AVR

- Example for C# programming environment: *(Download source)*



ID: 0000

Address: 0000

Command: 01

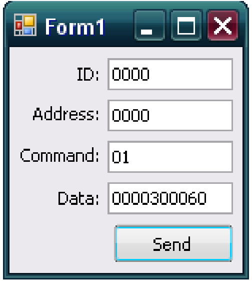
Data: 0000300060

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.IO.Ports;

namespace Visual_C_Sharp
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            //configuring the serial port
            serialPort1.PortName = "COM11";
            serialPort1.BaudRate = 9600;
            serialPort1.DataBits = 8;
            serialPort1.Parity = Parity.None;
            serialPort1.StopBits = StopBits.One;
        }

        private void button_Send_Click(object sender, EventArgs e)
        {
            //opening the serial port
            serialPort1.Open();
            string output;

            output = string.Format("<1{0:d4}", Convert.ToUInt16(Text_ID.Text)) +
               string.Format("{0:d4}", Convert.ToUInt16(Text_Address.Text)) + "01"
               + string.Format("{0:d2}", Convert.ToUInt16(Text_Command.Text)) +
               string.Format("{0:d10}>\r\n", Convert.ToInt32(Text_Data.Text));

            //write data to serial port
            serialPort1.WriteLine(output);

            MessageBox.Show (output);

            serialPort1.WriteLine("<10000000001010000200150>\r\n");

            //close the port
            serialPort1.Close();
        }
    }
}
```

Schedule3. For example, sending data with BASCOM-AVR

- Example for VB.Net programming environment: *(Download source)*

```
Public Class Form1

    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
     System.EventArgs) Handles MyBase.Load
        'configuring the serial port
        SerialPort1.PortName = "COM11"
        SerialPort1.BaudRate = 19200
        SerialPort1.DataBits = 8
        SerialPort1.Parity = IO.Ports.Parity.None
        SerialPort1.StopBits = IO.Ports.StopBits.One
    End Sub

    Private Sub Button_send_Click(ByVal sender As System.Object, ByVal e As
     System.EventArgs) Handles Button_send.Click
        Dim output As String
        output = String.Format("<1{0:0000}", Convert.ToInt32(TExt_ID.Text)) +
        String.Format("{0:0000}", Convert.ToInt32(TExt_Address.Text)) + "01" +
        String.Format("{0:00}", Convert.ToInt32(TExt_Command.Text)) +
        String.Format("{0:0000000000}>", Convert.ToInt32(Text_Data.Text)) +
        Chr(13) + Chr(10)
        'opening the serial port
        SerialPort1.Open()
        SerialPort1.WriteLine(output)
        MessageBox.Show(output)
        SerialPort1.WriteLine("<1000000000101000020015 0>" + vbCrLf)
        SerialPort1.Close()
    End Sub

End Class
```

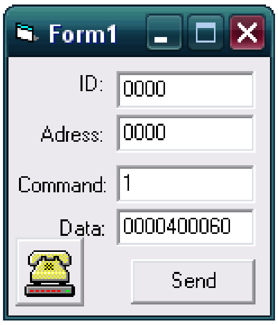


ID: 0000
Address: 0000
Command: 01
Data: 0000300060

Schedule4. For example, sending data with BASCOM-AVR

- Example for VB6 programming environment: *(Download source)*

```
Private Sub Form_Load()
    If MSComm1.PortOpen = True Then MSComm1.PortOpen = False
    'configuring the serial port
    MSComm1.Settings = "9600,n,8,1"
    MSComm1.CommPort = 11
End Sub

Private Sub Command_send_Click()
    Dim S As String
    'opening the serial port
    MSComm1.PortOpen = True

    S = "<1" & Format(Text_ID.Text, "0000") & Format(Text_addres.Text, "0000") &
    "01" & Format(Text_Command.Text, "00") & Format(Text_data.Text, "0000000000") &
    ">" & Chr(13) & Chr(10)
    MSComm1.Output = S
    MsgBox S
    MSComm1.Output = "<1000000000101000020015 0>" & Chr(13) & Chr(10)

   'Close the port
    MSComm1.PortOpen = False

End Sub
```

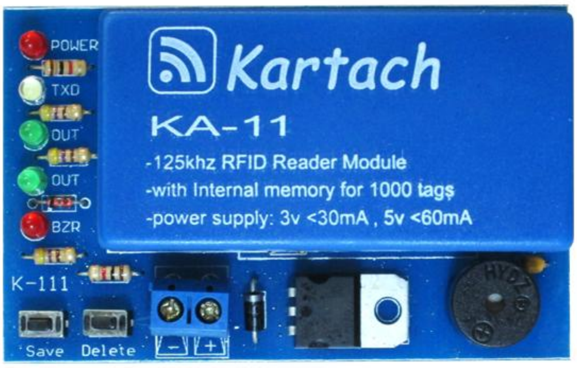


ID: 0000
Adress: 0000
Command: 1
Data: 0000400060

Schedule5. For example, sending data with BASCOM-AVR

## 21. Driver circuits.

K-111 (Module tester board) voltage: 9 to 15 volts


Kartach
KA-11
-125khz RFID Reader Module
-with Internal memory for 1000 tags
-power supply: 3v <30mA , 5v <60mA
POWER
TXD
OUT
OUT
BZR
K-111
Save
Delete
-
+

Size of Board: 7.6cm x 4.8cm

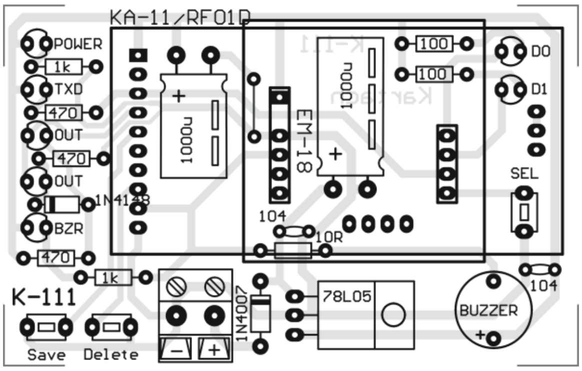


Figure14. K-111 Profile Components

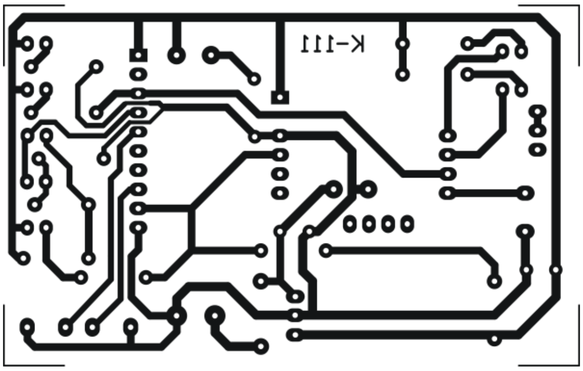


Figure15. K-111 PCB map (Scale 1:1 Printable on board)

## K-112 (Module and pc Interface circuit (USB to Com Converter)

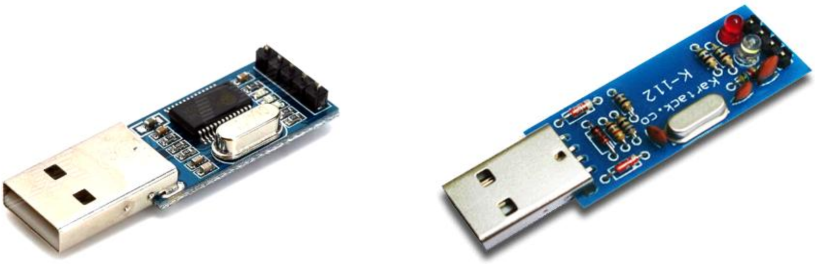
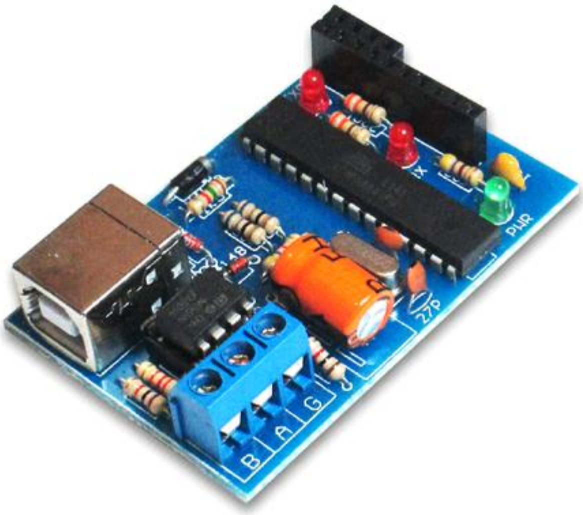


Figure16. K-112 (USB to Com Converter)

## K-113 (Interface between the PC and the module (USB to RS485 Converter)



Size of Board: 5.8cm x 4cm

Kartach
KA-11
-125khz RFID Reader Module
-with Internal memory for 1000 tags
-power supply: 3v <30mA , 5v <60mA
TX
RX
12M
27P
PWR
B
A
G

## K-114 (USB Com-Port RFID)

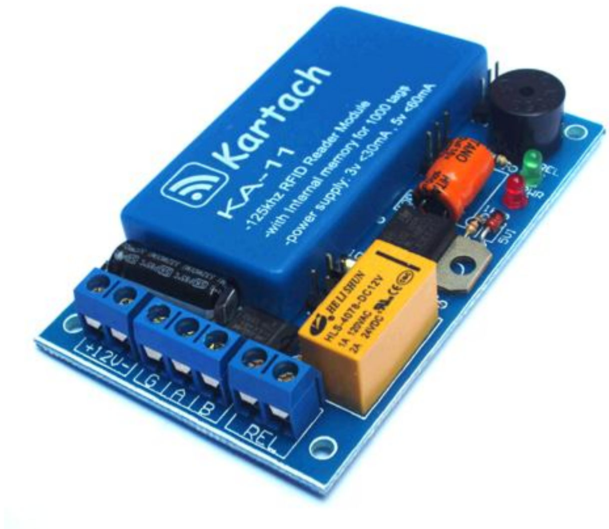
Size of Board: 6.3cm x 4.1cm

## K-115 (USB HID Keyboard RFID)

Size of Board: 6.3cm x 4.1cm

Kartach
KA-11L
-125khz RFID Reader Module
-with Internal memory for 1000 tags
-power supply: 3v <10mA , 5v <20mA
K115

K-116 (12V DC driver board with relay output and RS485 port)


Kartach
KA-11
-125khz RFID Reader Module
-with Internal memory for 1000 tags
-power supply: 3v <30mA , 5v <60mA
+12V-
G
A
B
REL

Size of Board: 8.4cm x 5.3cm x 1.4cm

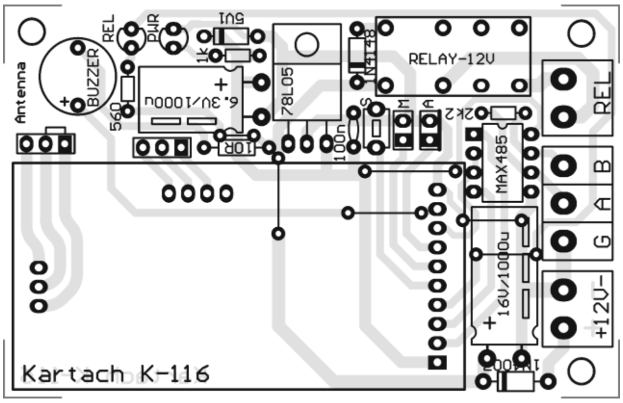


Figure17. K-111 Profile Components

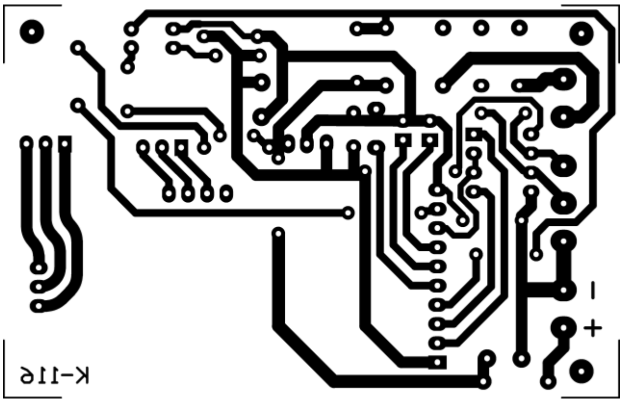
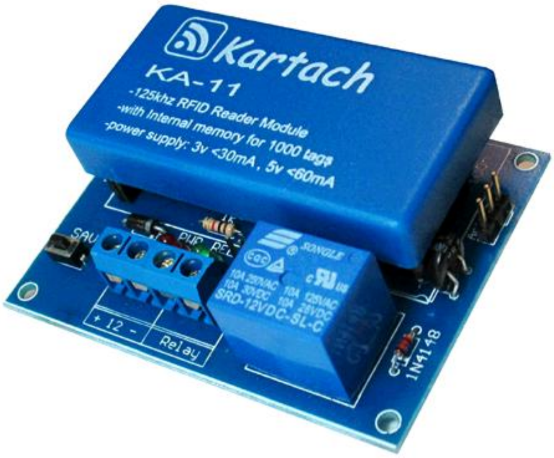


Figure18. K-111 PCB map (Scale 1:1 Printable on board)

## K-117 (Driver Board 12V DC with relay output)



Size of Board: 7.3cm x 5.4cm x 2cm

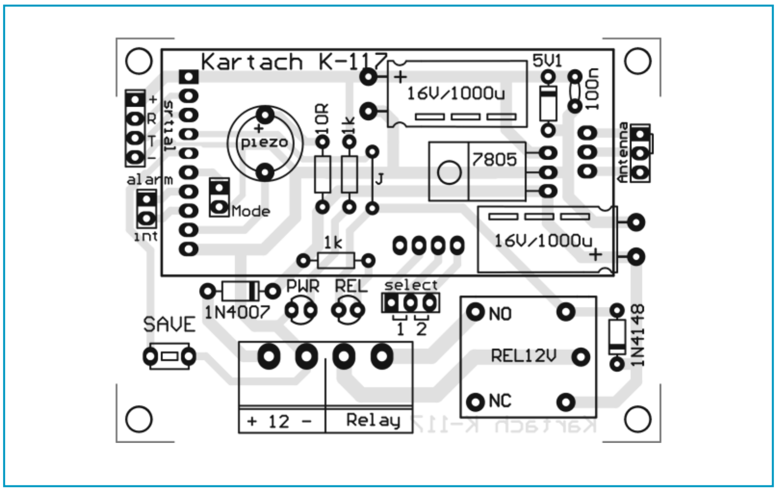


Figure19. K-111 Profile Components

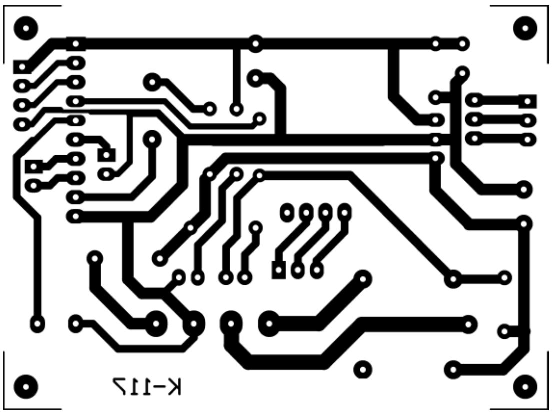
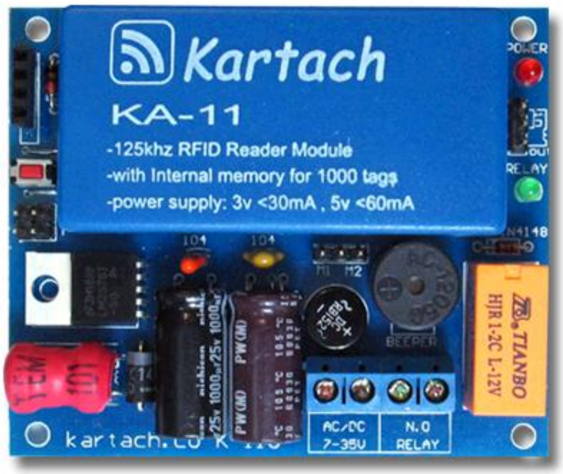


Figure20. K-111 PCB map (Scale 1:1 Printable on board)

## K-118 (Driver Board 7-35V AC/DC with relay output)


Kartach
KA-11
-125khz RFID Reader Module
-with Internal memory for 1000 tags
-power supply: 3v <30mA , 5v <60mA
POWER
RELAY
1N4148
104
104
M1
M2
BEEPER
AC/DC
7-35V
N.O
RELAY

Size of Board: 7.8cm x 6.5cm x 1.4cm

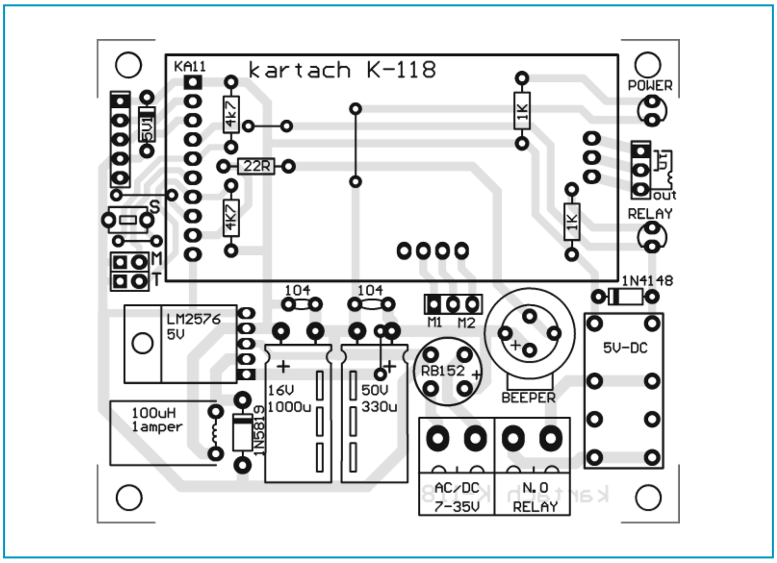


Figure21. K-111 Profile Components

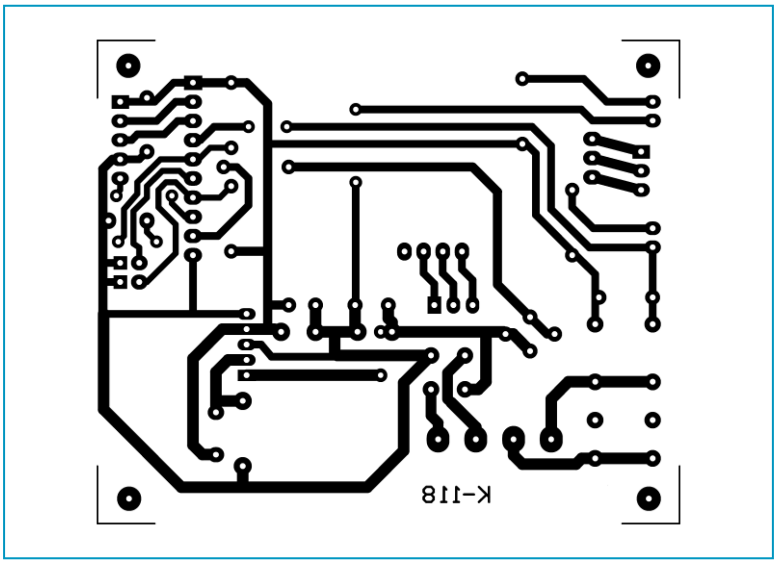
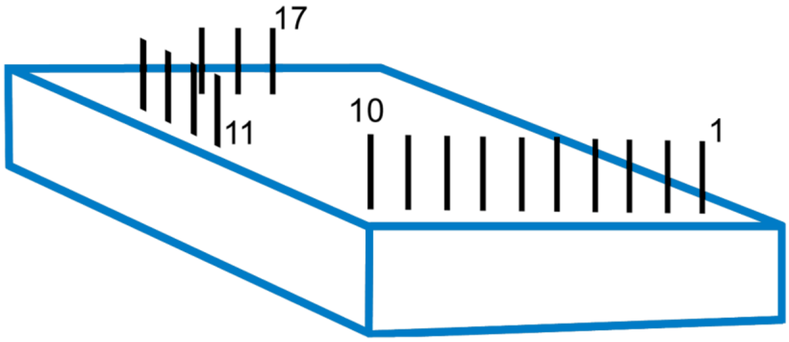
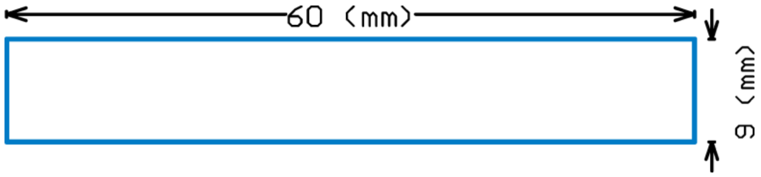
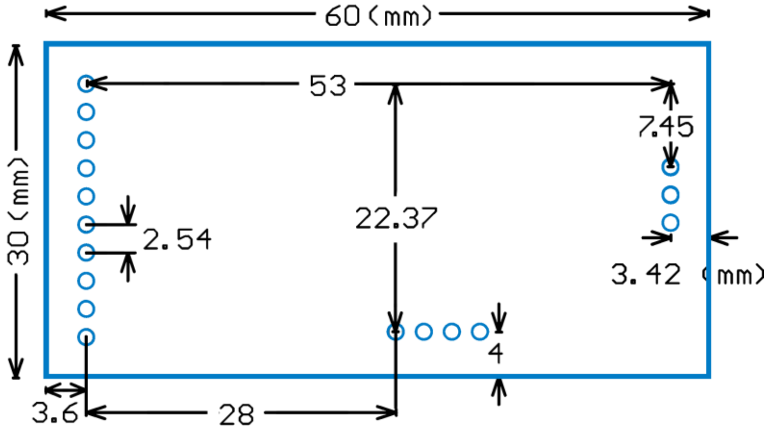


Figure22. K-111 PCB map (Scale 1:1 Printable on board)

## 22. Packaging Information.






60 (mm)
53
7.45
30 (mm)
2.54
22.37
3.42 (mm)
4
3.6
28