First Steps with your AutomationBoard

Thank you for choosing AutomationBoard, a board developed to make the automation system design based on Arduino platform a faster and easier process.

During your development, please always check the board schematics, this way you will be sure of making the right connections and taking advantage of all the resources the board can offer.

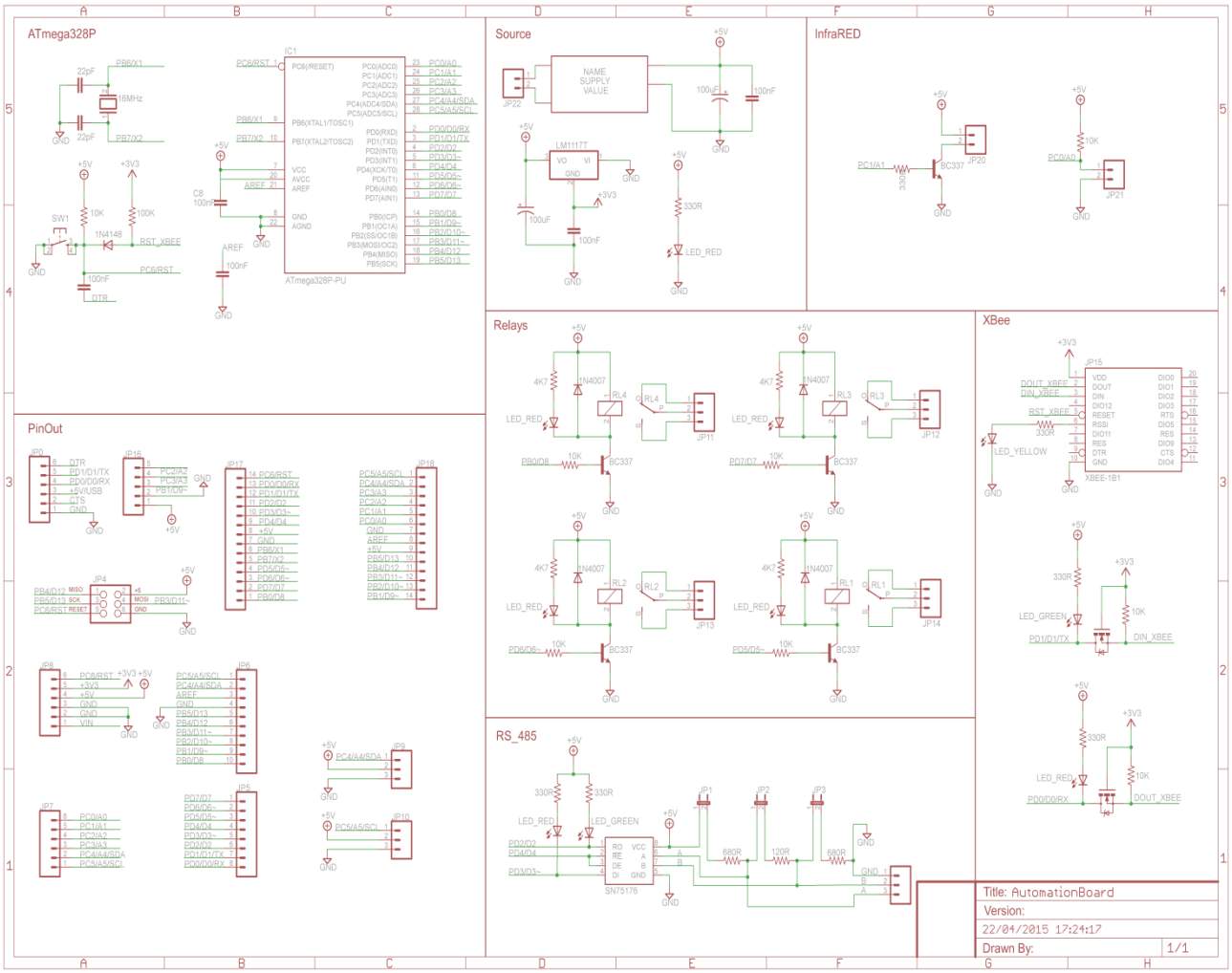


Figure 1: AutomationBoard schematics

Use the following url to download a high definition image of the schematics:

http://www.garagelab.com.br/automation/Images/ABSechmatic.png

Important: Both, AutomationBoard and this article, are intended for developers. You must have basic knowledge on electricity, electronics, programming and Arduino Platform to use them correctly.

Step 1 – Turning the board on

The board includes an ATmega328 microcontroller programmed with the same Arduino UNO bootloader. However, it does not do anything yet, it is necessary to program it with your own code.

At this point you can power it and check if its on-board power supply is working properly. You can do it checking if the LED PWR is on as explained below.

The on-board power supply is responsible to power all circuits on the board, like the digital circuits, relays and the XBee module, which uses the regulated 3.3V line. So you do not have to worry about powering any circuit on the board using external batteries or extra power supplies.

Warning!!!
Risk of electric shock!
AutomationBoard uses mains. The voltage level used is danger and can kill.
DO NOT touch any part of the board if it is connected to mains.
DO NOT place the board on metallic or conductive surfaces to prevent short circuit.

Use an appropriate AC power cord and attach it to the 110/220VAC screw terminal on the board (Figure 2). After that, connect the plug to mains socket. The LED PWR must turn on (Figure 3).

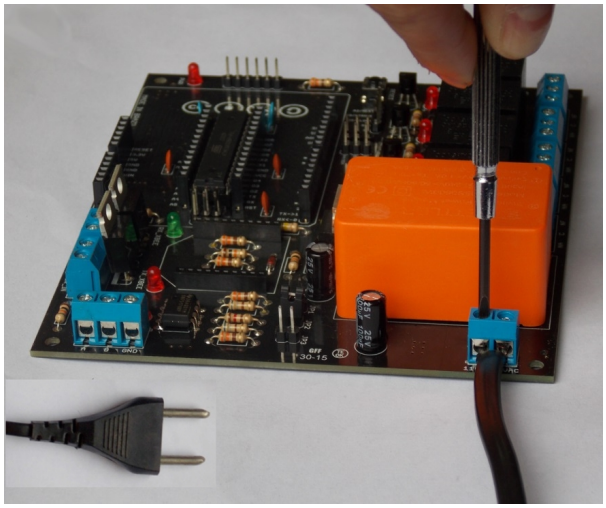


Figure 2 – Connecting an AC power cord

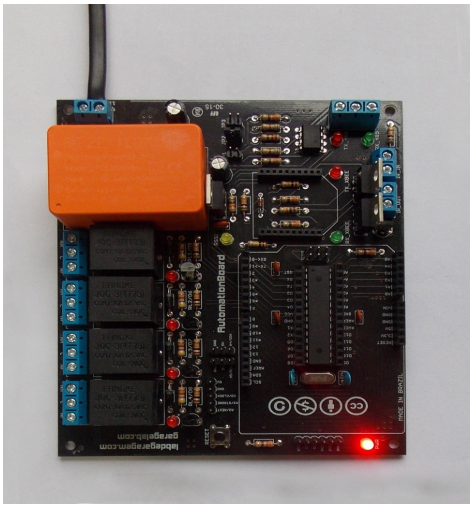


Figure 3 – LED PWR on

Step 2 – USB/Serial converter

You will need the USB/Serial converter to program your board. Check the Figure 4 and be sure your module is on the right position.

First, connect the USB cable (not included) to the module, and then connect the module to the board. After that, connect the USB cable to your computer.

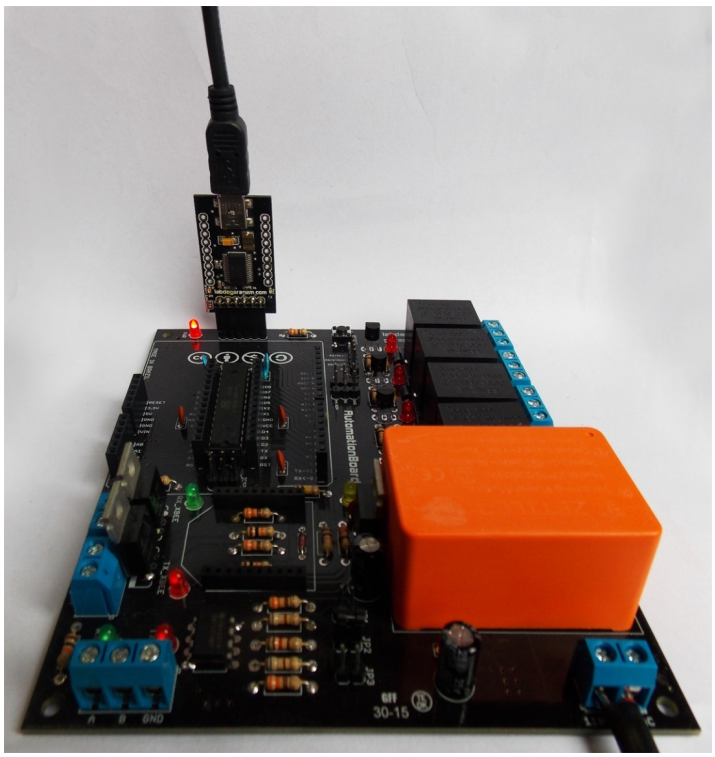


Figure 4 – Connecting the USB/Serial converter to the board

Step 3 – Uploading your first code

Open the Arduino IDE Software to upload your code and turn on one of the relays.

By the board schematics you can verify that the D5 digital pin is responsible to turn the Relay 1 on and off. So, to turn it on we must write a “HIGH” value to the pin.

The code below turn the Relay 1 on and off every 3 seconds.

```
void setup()
{
  pinMode(5, OUTPUT);
}
void loop()
{
  digitalWrite(5, HIGH);
  delay(3000);
  digitalWrite(5, LOW);
  delay(3000);
}
```

Copy and paste the code above on your Arduino IDE.

Click on the “Tools” tab, and select “Board” and then “Arduino UNO”.

While still on “Tools” option, select “Serial Port” and then choose the available serial.

Click “Upload”.

After the upload process finishes you will see the LED below Relay 1 turning on and off every 3 seconds. You also will hear a click inside the relay, what indicates it is also working (Figure 5).

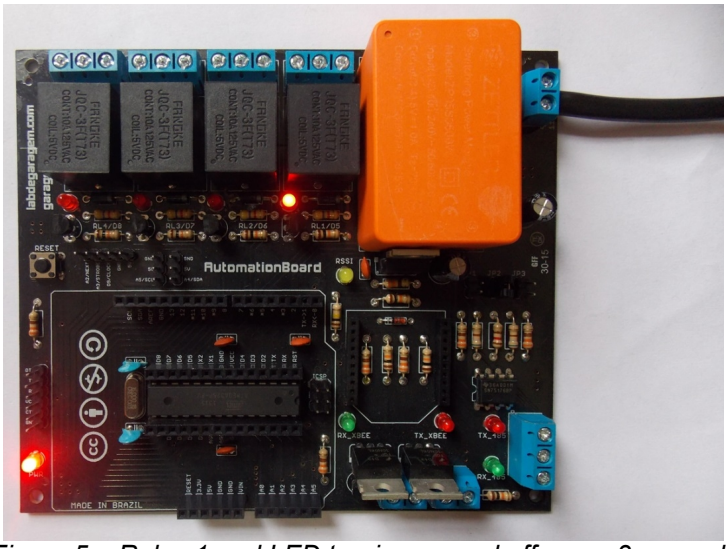
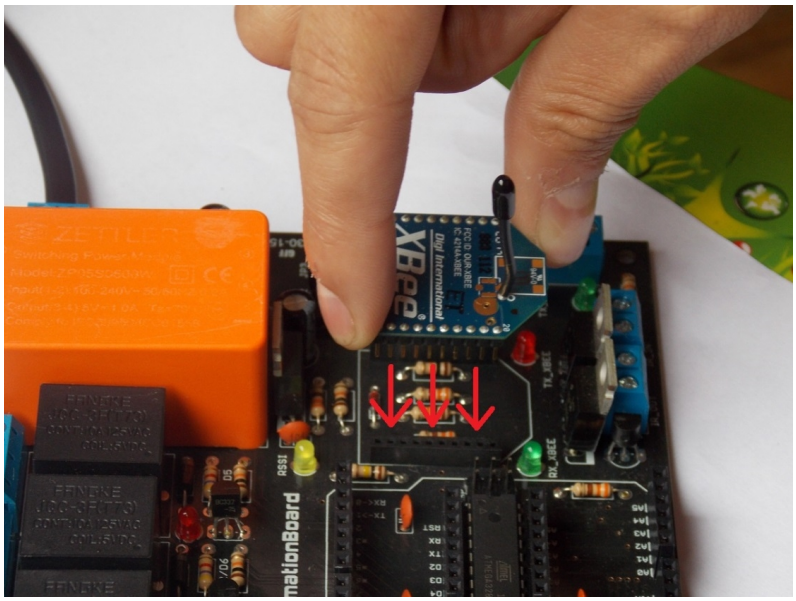


Figure 5 – Relay 1 and LED turning on and off every 3 seconds

Step 4 – Using an XBee module

If you are familiar with XBee modules and intend to use them to control your AutomationBoard wirelessly, place your module on the board as showed on Figure 6. Observe the diagonal lines indicating the correct position.

The board has on-board logic level converters and 3.3V regulator, so you do not have to worry about those things to use your module.


Figure 6 – Placing an XBee module

The relays are not controlled by the XBee module directly, instead, the XBee module communicates with the ATmega328 through a serial communication, Arduino pins RX/TX (D0 and D1), and then, the ATmega328 alone controls the relays.

We provide an example code below. You can use an XBee module connected to your computer to send characters to the module on the board.

```
void setup(){
  Serial.begin(9600);
  pinMode(5, OUTPUT);
  pinMode(6, OUTPUT);
  pinMode(7, OUTPUT);
  pinMode(8, OUTPUT);
}

void loop(){
 if (Serial.available()>0)
   {
     char a = 0;
     a = Serial.read();
     if (a=='5') { PORTD ^= (1<< 5); }
     if (a=='6') { PORTD ^= (1<< 6); }
     if (a=='7') { PORTD ^= (1<< 7); }
     if (a=='8') { PORTB ^= (1<< 0); }
     Serial.write(a);
   }
}
```

With the code above the microcontroller receives characters via a serial communication with the module. Everytime it receives the characters “5”, “6”, “7” or “8”, the corresponding relay will change its state on/off (toggle).

This article is NOT intended for XBee modules use learning. There are excellent tutorials on the internet. We like to recommend this: http://examples.digi.com/category/get-started/

Step 5 – Using Shields

You can also use Arduino or Garagino shields with your AutomationBoard. But it is very important that you check the pins your shield uses and if they are available on the board. So please, before connecting your shield check the following table:

Pin	**Function**
D0	Serial Communication (RX) / XBee DOUT
D1	Serial Communication (TX) / XBee DIN
D2	RS485 RO
D3	RS485 DI
D4	RS485 RE/DE
D5	Relay 1
D6	Relay 2
D7	Relay 3
D8	Relay 4
D9, D10, D11, D12, D13	Available
A0	Infrared in
A1	Infrared out
A2, A3, A4, A4	Available

Step 6 – Using Sensors

You can directly connect sensors to the Arduino or Garagino headers around the microcontroller, but the best way is using the 2 rows of 3 pins showed on Figure 7.

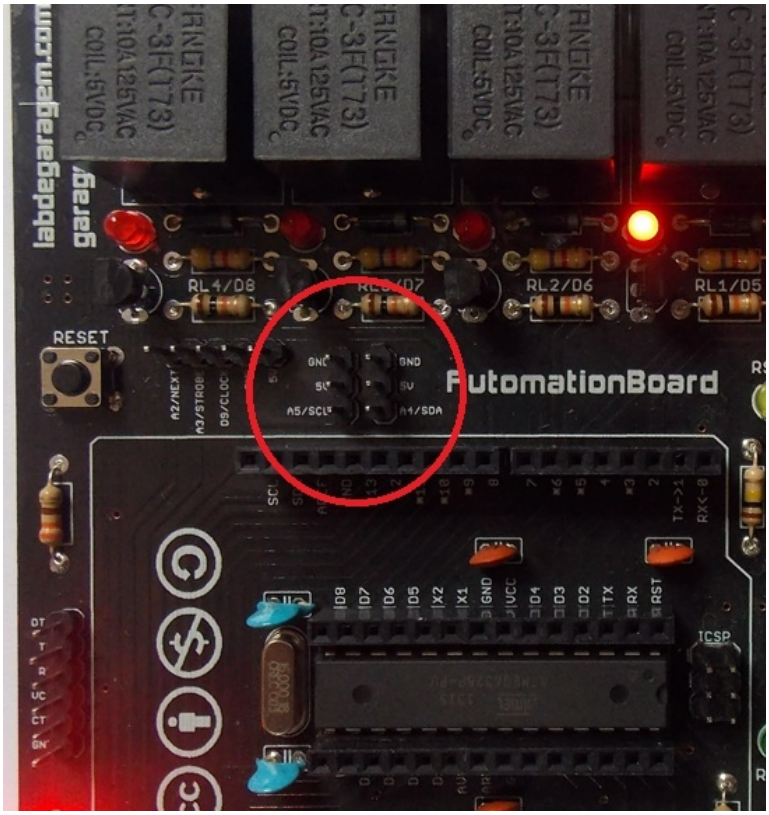


Figure 7 – Pins for sensor connections

Through those pins you can use an female connector to connect analog or digital sensors. The I2C bus is also available.

Step 7 – Infrared

AutomationBoard can also be used to send and receive IR signals, acting like or being controlled by a remote control.

Figure 8 shows the circuit block for IR.

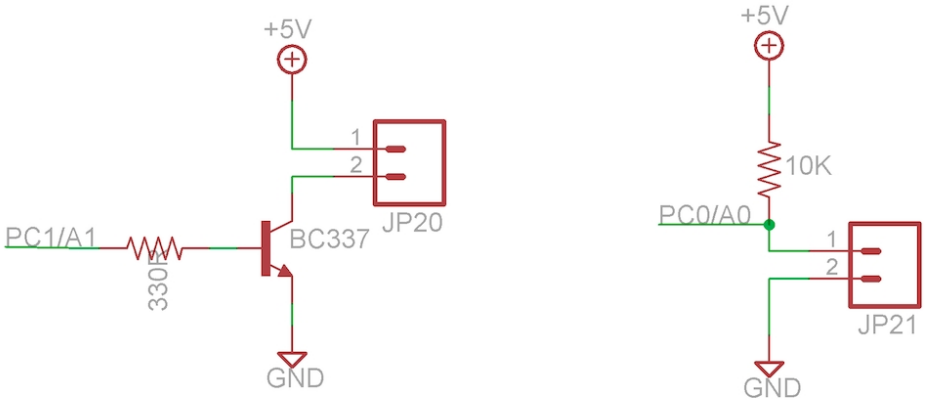


Figure 8 – Circuit on the board for IR LED and phototransitor

JP20 is the IR_OUT screw terminal, where a transistor is used to send signals. This way you will have an extra current to reach a higher power level to activate an IR emitter. JP21 is the IR_IN screw terminal, it can be connected to as IR phototransistor. The pin PC0/A0 can be used to capture the signal received.

Step 8 – RS-485

The RS-485 standard was included on AutomationBoard so it can be used on places with some electromagnetic interference and for a wired long range communication. The chip used on the board is the SN75176, a TIA/EIA-485-A standard transceiver.

Datasheet: http://www.ti.com/lit/ds/symlink/sn65176b.pdf

Figure 9 shows the circuit on the board.

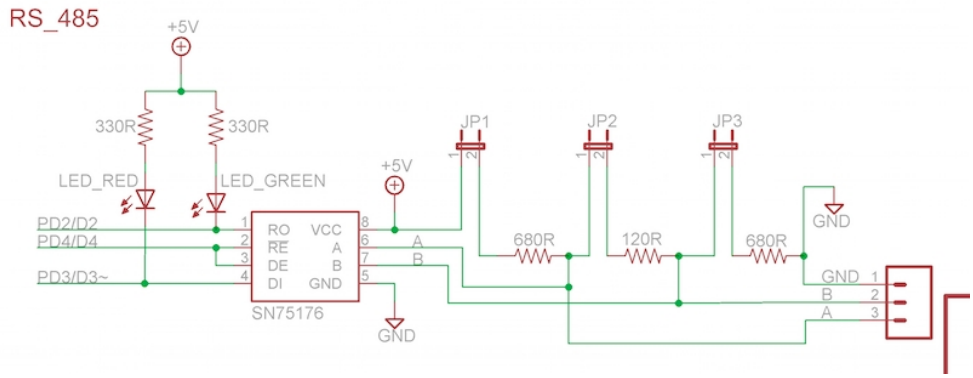


Figure 9 – RS-485. Circuit of the SN75176

You can find the screw terminal corresponding to the circuit above on the right lower corner of the board, the pins are named as A, B and GND. The LEDs RX_485 and TX_485 are in the same area, so are the connectors JP1, JP2 and JP3.

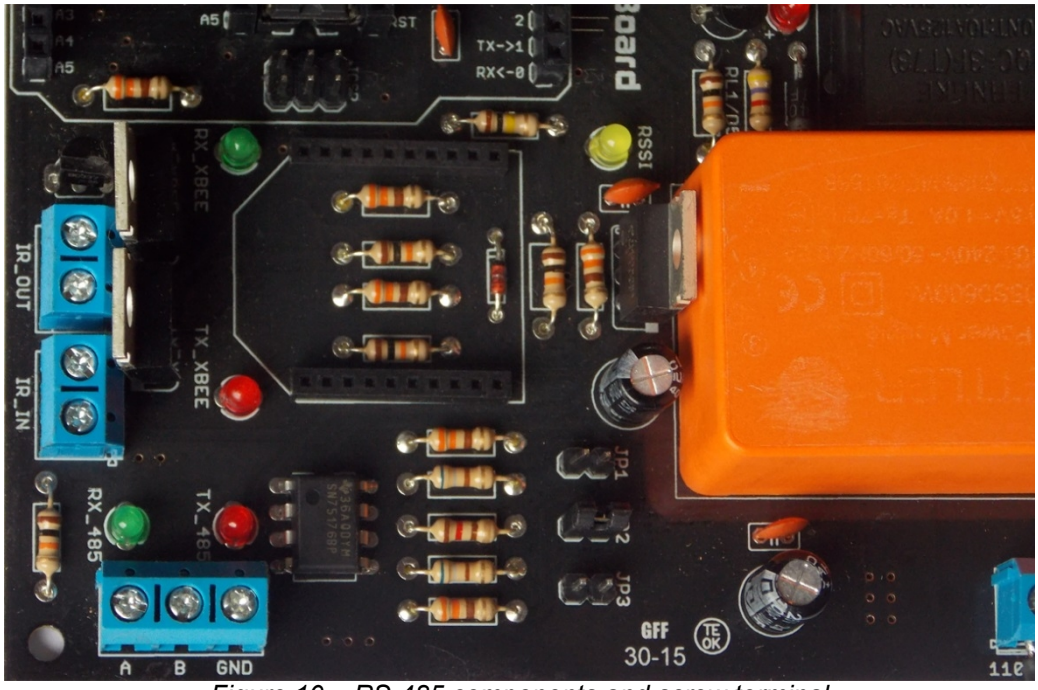


Figure 10 – RS-485 components and screw terminal

Step 9 – SideKick board

If you chose to buy your SideKick along with AutomationBoard, you can control 2 extra relays via XBee modules. The SideKick board also have a power supply onboard, allowing it to be powered directly from the AC mains. The AC power cord must be connected to the board as showed on Figures 11 and 12.

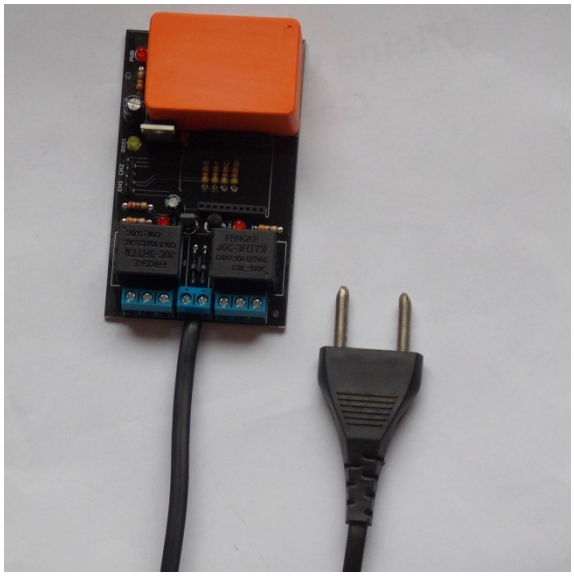


Figure 11 - SideKick board - top

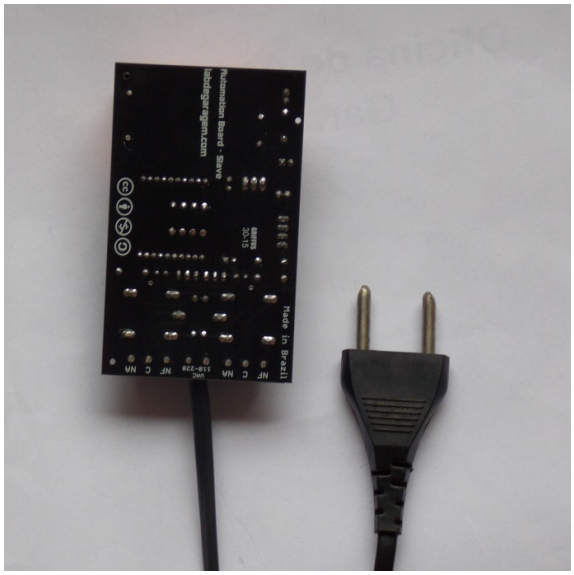


Figure 12 - SideKick board - bottom

The 3 pins screw terminals on each side are connected directly to each relay on the board. On the bottom silk you can check the name of each pin.

Below it is shown the SideKick circuit.

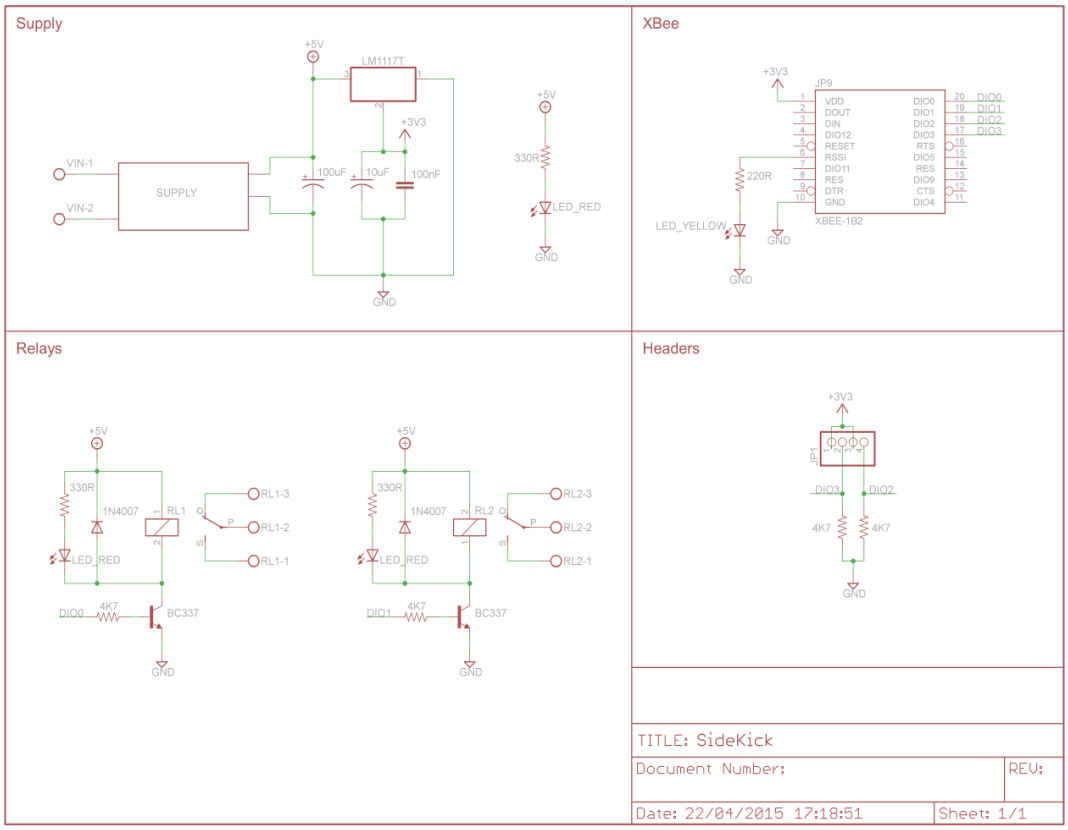


Figure 13 – SideKick schematics

http://www.garagelab.com.br/automation/Images/Sidekick.png

You can find the XBee modules pins DIO0 and DIO1. On the contrary of AutomationBoard, SideKick's XBee modules are directly responsible for turning the relays on/off. You can also use the pins CH1 and CH2 to connect devices like switches or sensors. When the pins CH1 are closed (short-circuit) the pin DIO2 read value will be “HIGH”, if open, it will be “LOW”. This is interesting if you want to use a kind of manual control in parallel with the automatic control.

A module on the computer can read those values (DIO2 and DIO3) and take an action, following your code.

Step 10 – Have fun and share!

We would love to see what you are making with your AutomationBoard, so we invite you to join our social network and post your project.

www.garagelab.com

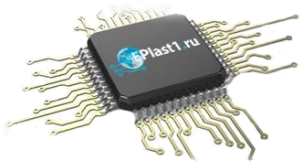
Компания «ЭлектроПласт» предлагает заключение долгосрочных отношений при поставках импортных электронных компонентов на взаимовыгодных условиях!

Наши преимущества:

- Оперативные поставки широкого спектра электронных компонентов отечественного и импортного производства напрямую от производителей и с крупнейших мировых складов;
- Поставка более 17-ти миллионов наименований электронных компонентов;
- Поставка сложных, дефицитных, либо снятых с производства позиций;
- Оперативные сроки поставки под заказ (от 5 рабочих дней);
- Экспресс доставка в любую точку России;
- Техническая поддержка проекта, помощь в подборе аналогов, поставка прототипов;
- Система менеджмента качества сертифицирована по Международному стандарту ISO 9001;
- Лицензия ФСБ на осуществление работ с использованием сведений, составляющих государственную тайну;
- Поставка специализированных компонентов (Xilinx, Altera, Analog Devices, Intersil, Interpoint, Microsemi, Aeroflex, Peregrine, Syfer, Eurofarad, Texas Instrument, Miteq, Cobham, E2V, MA-COM, Hittite, Mini-Circuits,General Dynamics и др.);

Помимо этого, одним из направлений компании «ЭлектроПласт» является направление «Источники питания». Мы предлагаем Вам помощь Конструкторского отдела:

- Подбор оптимального решения, техническое обоснование при выборе компонента;
- Подбор аналогов;
- Консультации по применению компонента;
- Поставка образцов и прототипов;
- Техническая поддержка проекта;
- Защита от снятия компонента с производства.



Как с нами связаться

Телефон: 8 (812) 309 58 32 (многоканальный)
Факс: 8 (812) 320-02-42
Электронная почта: org@eplast1.ru
Адрес: 198099, г. Санкт-Петербург, ул. Калинина, дом 2, корпус 4, литера А.