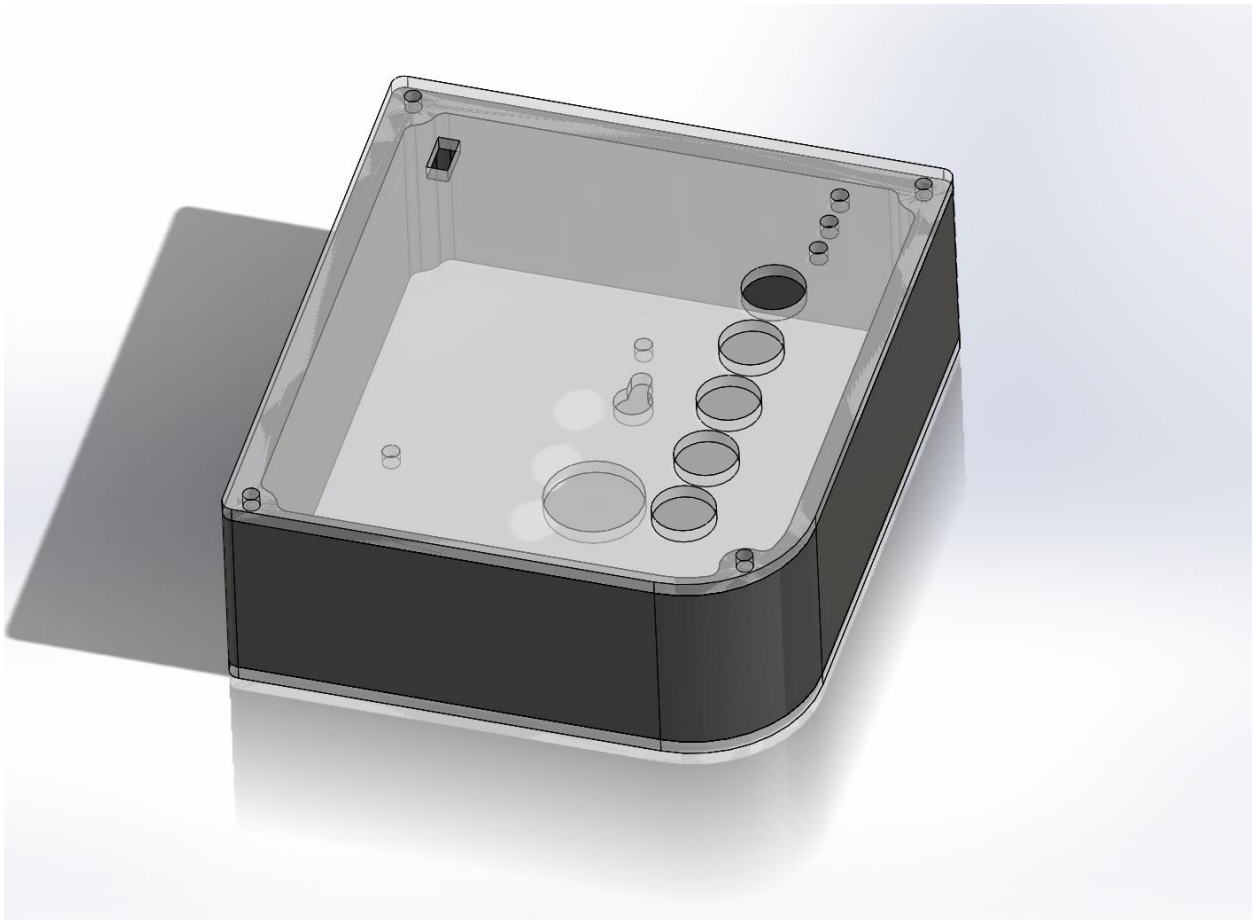


# GATEBOX: SIMPLIFIED HARDWARE INSTRUCTIONS

By Shafeeq Rabbani



## Contents

Introduction .....	2
Choosing a Bluetooth Low Energy Chip .....	2
Wire the Arduino UNO to the Bluetooth Low Energy Chip .....	3
The Relays Circuit.....	4
Soldering onto the Perf Board: .....	4
Printing your Own PCB.....	5
Pinout for Relays, Push Buttons, LEDs and Slide Switch .....	6
Final Steps: .....	7

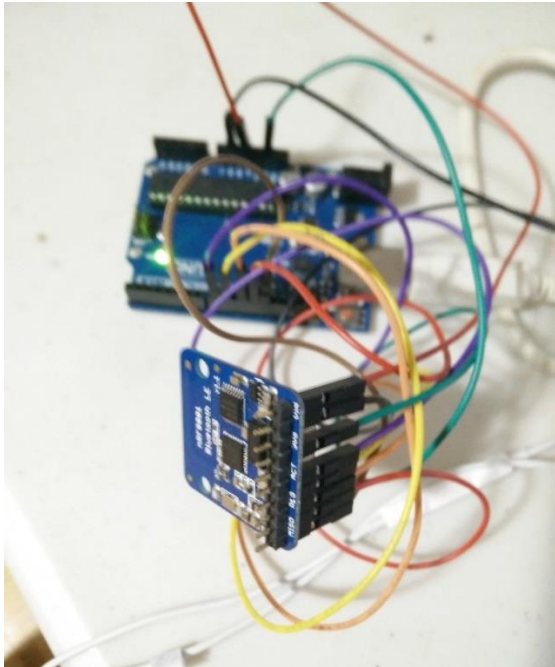

## Introduction

As mentioned in the Simplified Electronics Parts List document, You can approach this in the following ways:

- A. You can use an Arduino UNO board, a Bluetooth Low Energy Chip and a perf board for the relay circuitry. **This is the simplest possible approach to this project.**
- B. You can use an Arduino UNO board, a Bluetooth Low Energy Chip and a PCB (from PCB V7.brd). This is a lot like the previous method but can speed up development since you will not have to solder as many parts. As you are already using an Arduino UNO, you will not have to solder all the parts on the resulting PCB.
- C. You can solder all the components on the PCB V7.brd board (the PCB has place for Arduino UNO's components) and use a Bluetooth Low Energy chip.

## Choosing a Bluetooth Low Energy Chip

There are two options you can chose from:

Option 1	Option 2
The Adafruit Bluetooth Low Energy nRF8001 breakout board ( <a href="https://www.adafruit.com/product/1697">https://www.adafruit.com/product/1697</a> )	The RedBearLab BLE shield ( <a href="http://redbearlab.com/bleshield/">http://redbearlab.com/bleshield/</a> )
	
If you go with this option, you will need 8 male-female header cables, a row of 10 male headerpins , a soldering iron and some solder. Solder the header pins onto the nRF8001 breakout board.	The pins of the RedBearLab shield attach onto the Arduino UNO so you won't need any header wires. However from experience, the Case Enclosure is not big enough to fit the Arduino UNO with a shield along with the relays board and other electronics. You

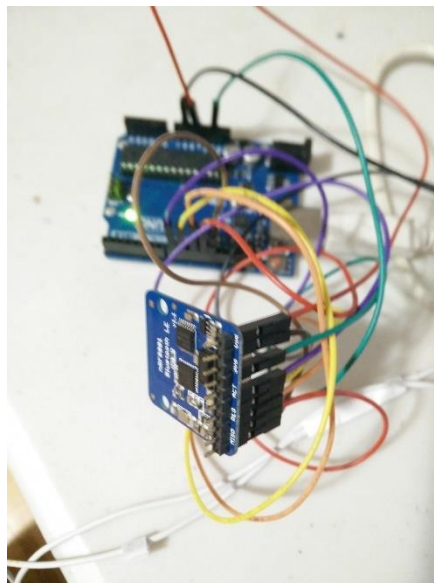
	may need a thicker case Elosure (by modifying the thickness in the side_covering.SLDPRT flie) or make a new one entirely from scratch).
--	---

### Wire the Arduino UNO to the Bluetooth Low Energy Chip

You should follow **the same pinout** to interface the Arduino UNO board to the Bluetooth Low Energy chip from the previous step **regardless** of which option you chose. If you are using the RedBearLab shield, this pinout should not be a problem as there is only one way to fix the shield onto the Arduino UNO board. If you are using the Bluetooth Low Energy module breakout board from Adafruit, make sure to follow this pinout:

Arduino UNO pin	Bluetooth Low Energy Chip pin
13	SCK (connects to SPI clock)
12	MISO (connects to SPI MISO)
11	MOSI (connects to SPI MOSI)
9	REQ (connects to SPI chip select pin)
8	RDY is the interrupt pin from the nRF8001
None	ACT
RESET	RST
None	3V OUT
GND	GND (connects to Ground pin on Arduino UNO)
5V	VIN (connects to 5V pin on Arduino UNO)

**Note:** It may be easier and lesser prone to error if you connect the header cables using the color code shown.

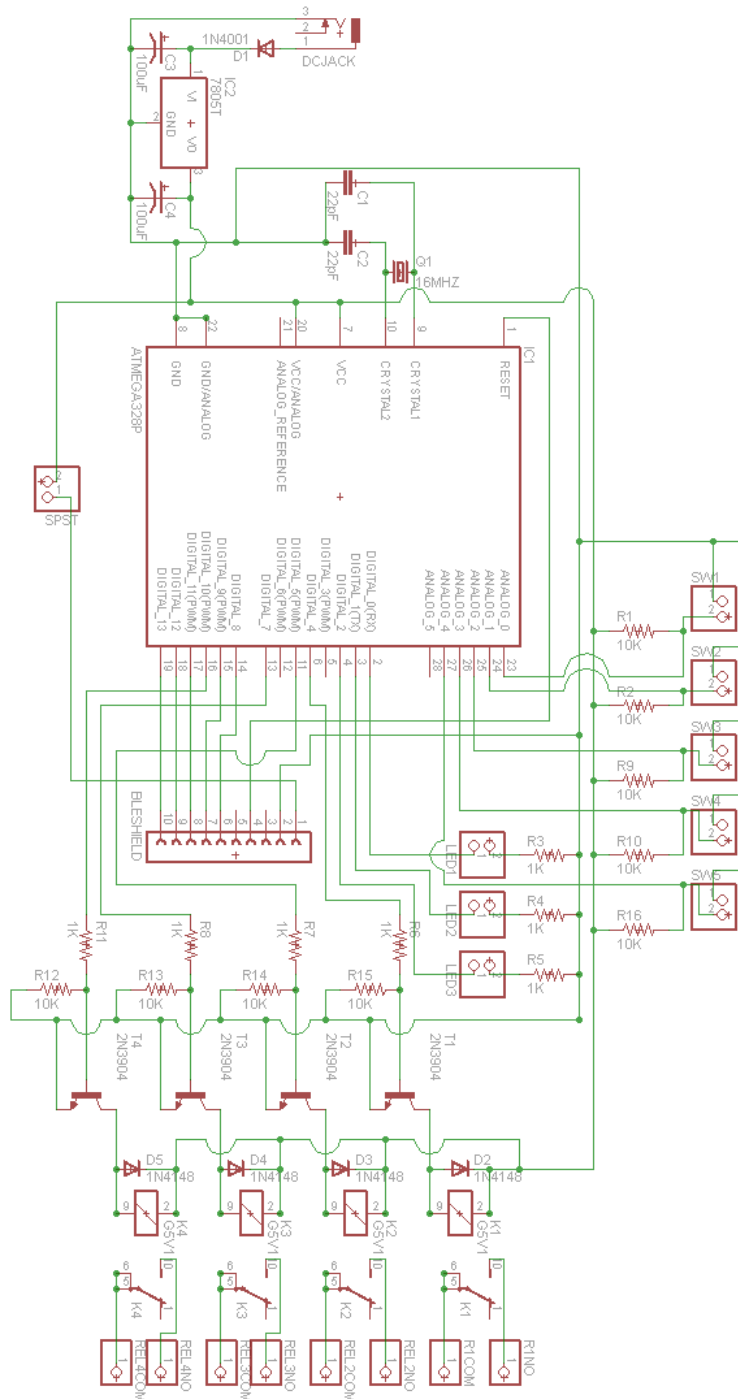


## The Relays Circuit

You may decide to have a portion of your circuitry printed (in which case you would want to print the file **PCB V7.brd** or you may choose to solder everything in which case you will need a perf board.

### Soldering onto the Perf Board:

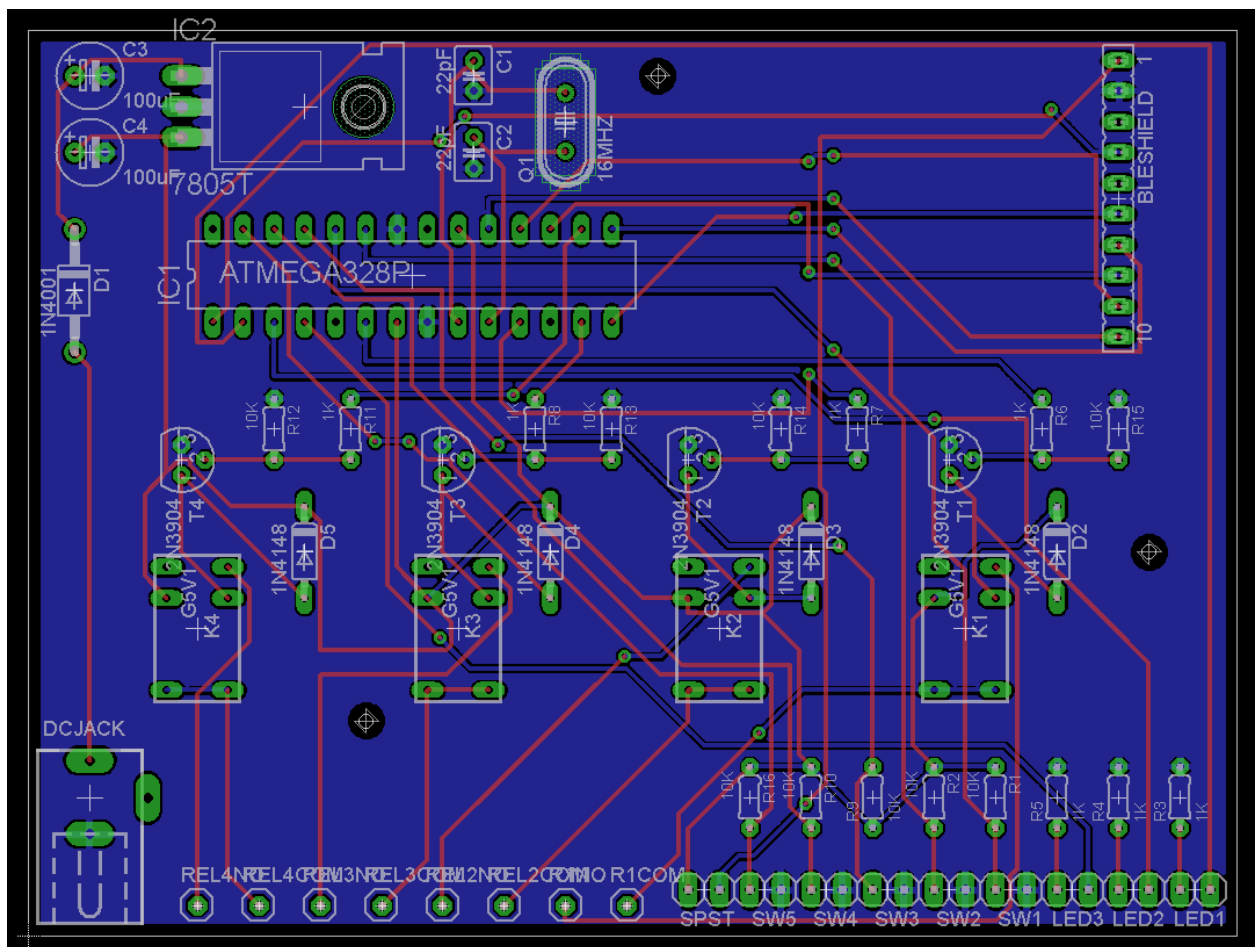
You will need to solder the items mentioned on the Simplified Electronics Parts List so that you produce this circuit:



## Printing your Own PCB

To print your own PCB, use the **PCB V7.brd** file in Simplified Schematic. If you are using an Arduino UNO board (Option A or B), you should **not** solder the following components onto the PCB.

- DC JACK
- Diode D1
- Capacitors C1, C2, C3, C4
- ATMEGA328P IC1
- Voltage Regulator 7805T IC2
- Quartz Crystal 16MHZ Q1



If you are using the entire board with the built in Arduino UNO chip (Option C), then you must solder everything. The Atmega328p chip will need to be programmed with the Arduino Bootloader before you can upload the Arduino Code onto it. There are a number of ways on the internet which show how to program Atmel328p chips with the Arduino Bootloader. For example, you can use the AVR ISP MKII from Atmel. Choose whatever method you are comfortable with. You can also purchase an Atmega328p chip that comes with the Arduino bootloader already on it.

## Pinout for Relays, Push Buttons, LEDs and Slide Switch



The Relays and LEDs are operated by output pins and the Push Buttons are operated by input pins.

The following table shows which pin corresponds to what function in the Arduino code “ConnectedCar” and “ConnectedCarTest”. Solder these connections or use header pins from the Arduino to the relay circuit board.

### Output pins

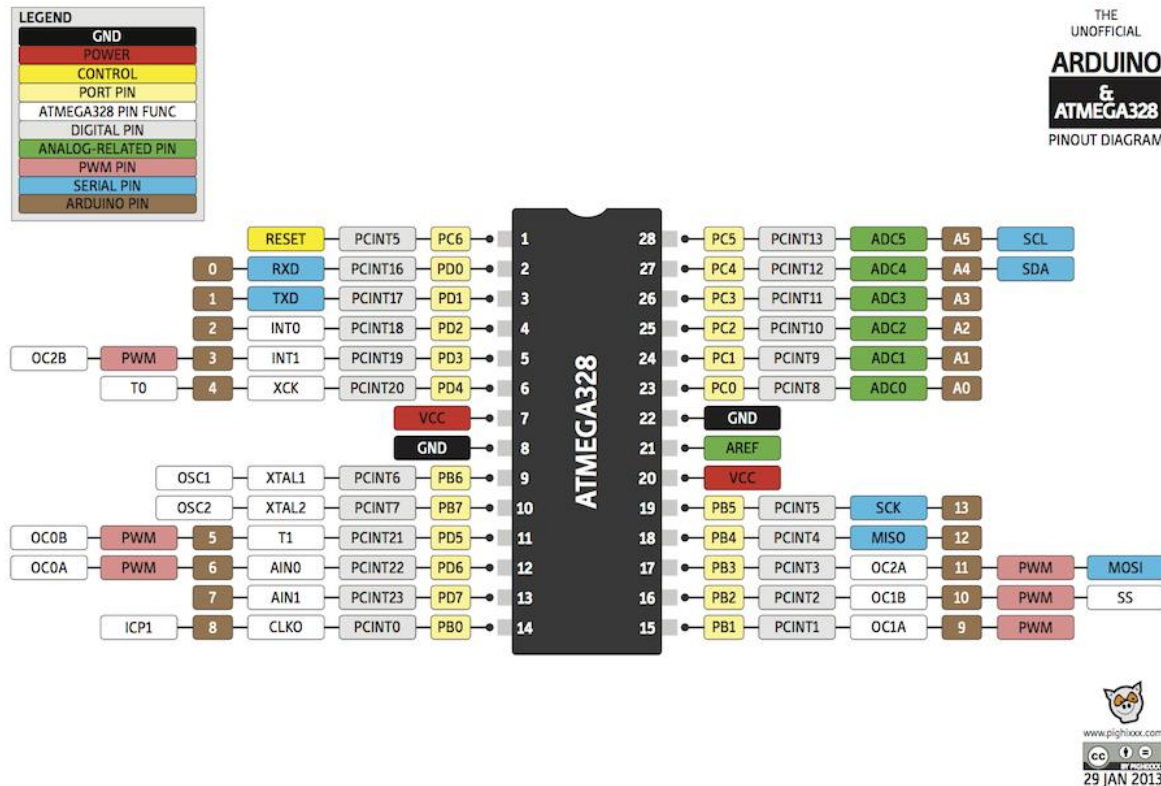
Component	Function	Pin Number
Relays	Garage Stop	7
	Garage Open	4
	Garage Close	5
	Lights On/Off	10
LEDs	LED: Red	3
	LED: Green	A5
	LED: Yellow	2

### Input pins

Component	Function	Pin Number
Push Button	Garage Stop	A0
	Garage Open	A1
	Garage Close	A2
	Lights On/Off	A3
	Bluetooth Unlock	A4
Slide Switch	Disable Bluetooth	This switch must be on the 5V wire connection between the

		Arduino and the Bluetooth Low Energy Chip. Hence, “Disable Bluetooth” turns of the power itself to the Bluetooth Chip.
--	--	--

If you are going ahead with Option C and are having trouble identifying which pin on the Atmega328p corresponds which pin number, use this diagram below:



### Final Steps:

Connect wires from pre-existing garage door openers into the terminal blocks. Refer to the Garage Door Openers on Market (Manuals) folder to get an idea of how majority of the garage door openers on the market work. Do the same for lights. As this involves working with the mains, this task must be carried out with the aid of a professional certified electrician.

Connect the 5V dc jack onto the Arduino to power it. You can now insert the electronics into the GateBox panel and close the seal to power it up.

Your GateBox panel is now ready.