Lesson 16 Controlling Stepper Motor With Remote

Introduction

In this lesson, you will learn a fun and easy way to control a stepper motor from a distance using an IR remote control.

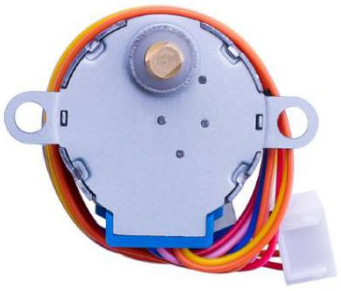
Hardware Required

- ✓ 1 * RexQualis UNO R3
- ✓ 1 * Breadboard
- ✓ 1 * IR receiver module
- ✓ 1 * IR remote
- ✓ 1 * ULN2003 stepper motor driver module
- ✓ 1 * Stepper motor
- ✓ 1 * Breadboard Power supply module
- ✓ 1 * 9V 1A Adapter
- ✓ 9 * F-M Jumper Wires
- ✓ 1 * M-M Jumper Wires

Principle

28BYJ-48 Stepper Motor

The 28BYJ-48 is a small, 5 volt geared stepping motors. These stepping motors are apparently widely used to control things like automated blinds, A/C units and are mass produced. stepper01 Due to the gear reduction



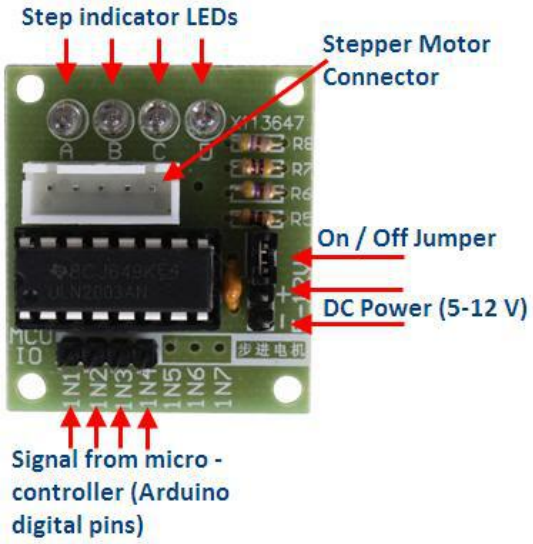
ratio of *approximately* 64:1 it offers decent torque for its size at speeds of about 15 rotations per minute (RPM). With some software ‘‘trickery’’ to accelerate gradually and a higher voltage power source (I tested them with 12 volts DC) I was able to get about 25+ RPM. These little steppers can be purchased together with a small breakout board for the Arduino compatible ULN2003 stepper motor driver. Quite a bargain, compared to the price of a geared DC motor, a DC motor controller and a wheel encoder! The low cost and small size makes the 28BYJ-48 an ideal option for small robotic applications, and an excellent introduction to stepper motor control with Arduino. Here are the detailed specs of the 28BYJ-48 stepper motor.

Stepper motor 28BYJ-48 Parameters

- ✓ Model:28BYJ-48
- ✓ Rated voltage:5VDC
- ✓ Number of Phase:4
- ✓ Speed Variation Ratio:1/64
- ✓ Stride Angle:5.625°/64
- ✓ Frequency:100Hz
- ✓ DC resistance:50Ω±7%(25°C)
- ✓ Idle In-traction Frequency:>600Hz
- ✓ Idle Out-traction Frequency:>1000Hz
- ✓ In-traction Torque >34.3mN.m(120Hz)
- ✓ Self-positioning Torque >34.3mN.m
- ✓ Friction torque:600-1200 gf.cm
- ✓ Pull in torque:300 gf.cm

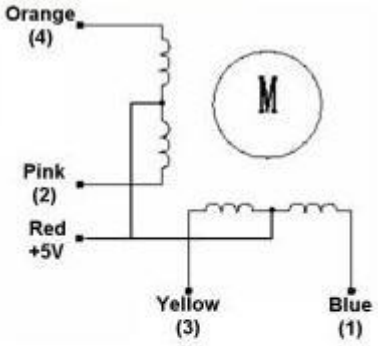
- ✓ Insulated resistance >10MΩ(500V)
- ✓ Insulated electricity power:600VAC/1mA/1s
- ✓ Insulation grade:A
- ✓ Rise in Temperature <40K(120Hz)
- ✓ Noise<35dB(120Hz,No load,10cm)

ULN2003 Driver Board



The ULN2003 stepper motor driver board allows you to easily control the 28BYJ-48 stepper motor from a microcontroller, like the Arduino Uno. One side of the board side has a 5 wire socket where the cable from the stepper motor hooks up and 4 LEDs to indicate which coil is currently powered. The motor cable only goes in one way, which always helps. UNL2003 board On the side you have a motor on / off jumper (keep it on to enable power to the stepper). The two pins below the 4 resistors, is where you provide power to the stepper. Note that powering the stepper from the 5 V rail of the Arduino is not recommended. A separate 5-12 V 1 Amp power supply or battery pack should be used, as the motor may drain more current than the microcontroller can handle and could potentially damage it. In the middle of the board we have the

ULN2003 chip. At the bottom are the 4 control inputs that should be connected to four Arduino digital pins.



Half-Step Switching Sequence

Lead Wire Color	---> CW Direction (1-2 Phase)							
	1	2	3	4	5	6	7	8
4 Orange	-	-						-
3 Yellow		-	-	-				
2 Pink				-	-	-		
1 Blue						-	-	-

Code interpretation

```
#include "Stepper.h"

#include "IRremote.h"

/*----- Variables, Pins -----*/

#define STEPS  32   // Number of steps per revolution of Internal shaft

int  Steps2Take;  // 2048 = 1 Revolution

int receiver = 12;  // Signal Pin of IR receiver to Arduino Digital Pin 6

/*-----( Declare objects )-----*/

// Setup of proper sequencing for Motor Driver Pins

// In1, In2, In3, In4 in the sequence 1-3-2-4
```

```
Stepper small_stepper(STEPS, 8, 10, 9, 11);
IRrecv irrecv(receiver);       // create instance of 'irrecv'
decode_results results;      // create instance of 'decode_results'
void setup()
{
   irrecv.enableIRIn();   // Start the receiver
}
void loop()
{
if (irrecv.decode(&results)) // have we received an IR signal?
   {
      switch(results.value)
      {
         case 0xFF02FD: // VOL+ button pressed
         small_stepper.setSpeed(500); //Max seems to be 500
         Steps2Take   =   2048;   // Rotate CW
         small_stepper.step(Steps2Take);
         delay(2000);
         break;
         case 0xFFC23D: // VOL- button pressed
         small_stepper.setSpeed(500);
         Steps2Take   =   -2048;   // Rotate CCW
```

```
 small_stepper.step(Steps2Take);
 delay(2000);
 break;
     }
 irrecv.resume(); // receive the next value
  }
}/* --end main loop -- */
```

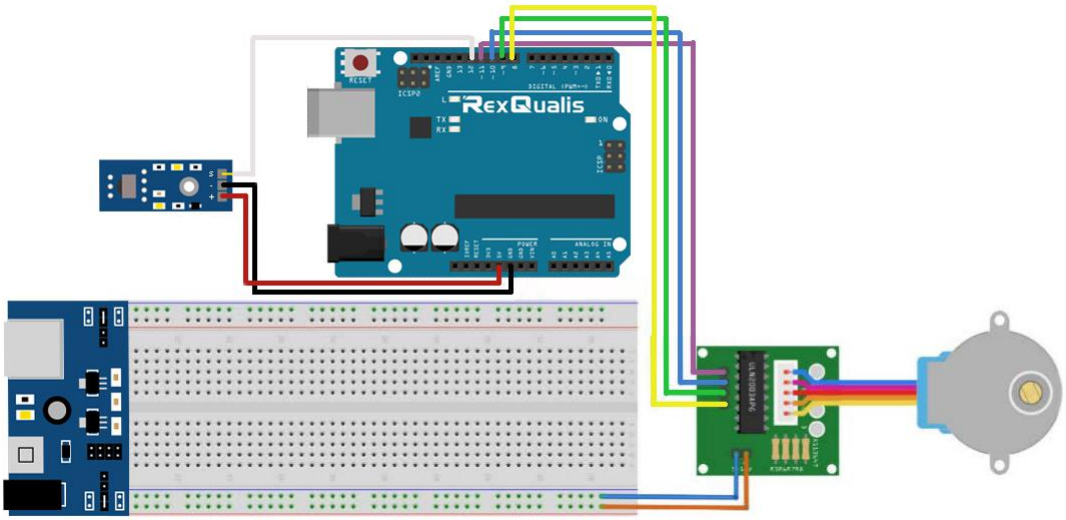
Experimental Procedures

Step 1:Build the circui

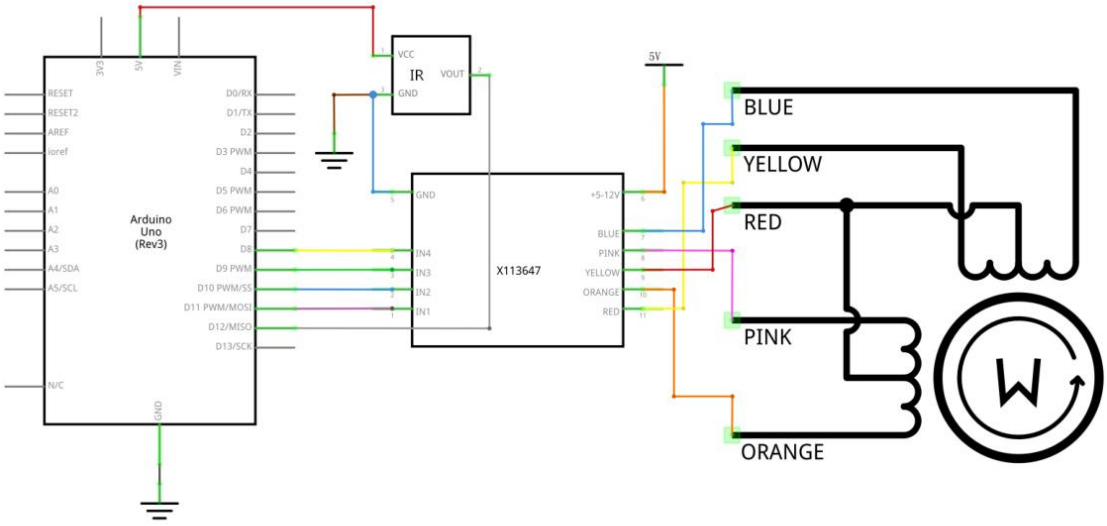
We are using 4 pins to control the Stepper and 1 pin for the IR sensor.

Pins 8-11 are controlling the Stepper motor and pin 12 is receiving the IR information.

We connect the 5V and Ground from the UNO R3 to the sensor. As a precaution, use a breadboard power supply to power the stepper motor since it can use more power and we don’t want to damage the power supply of the UNO R3.

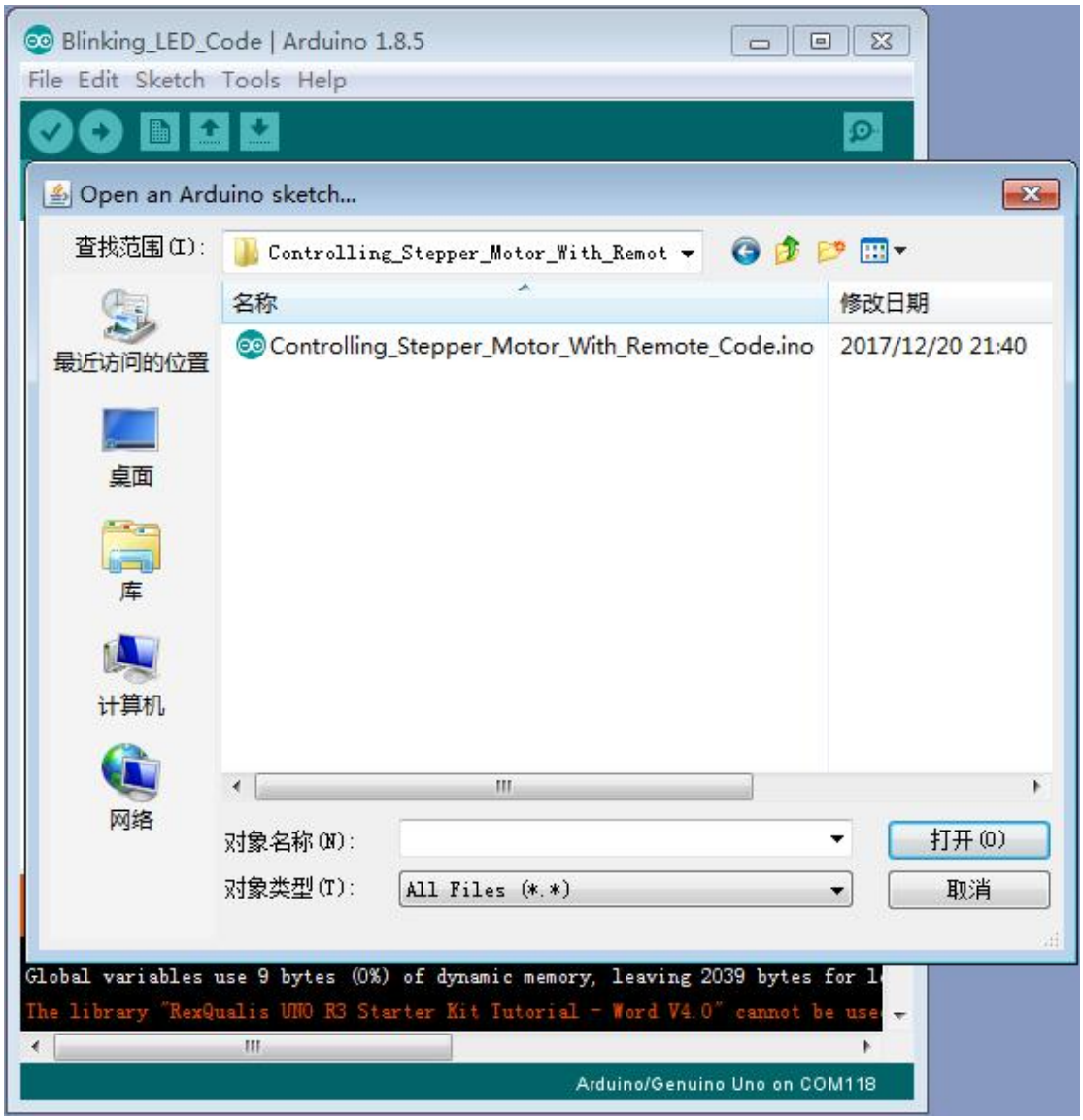


Schematic Diagram



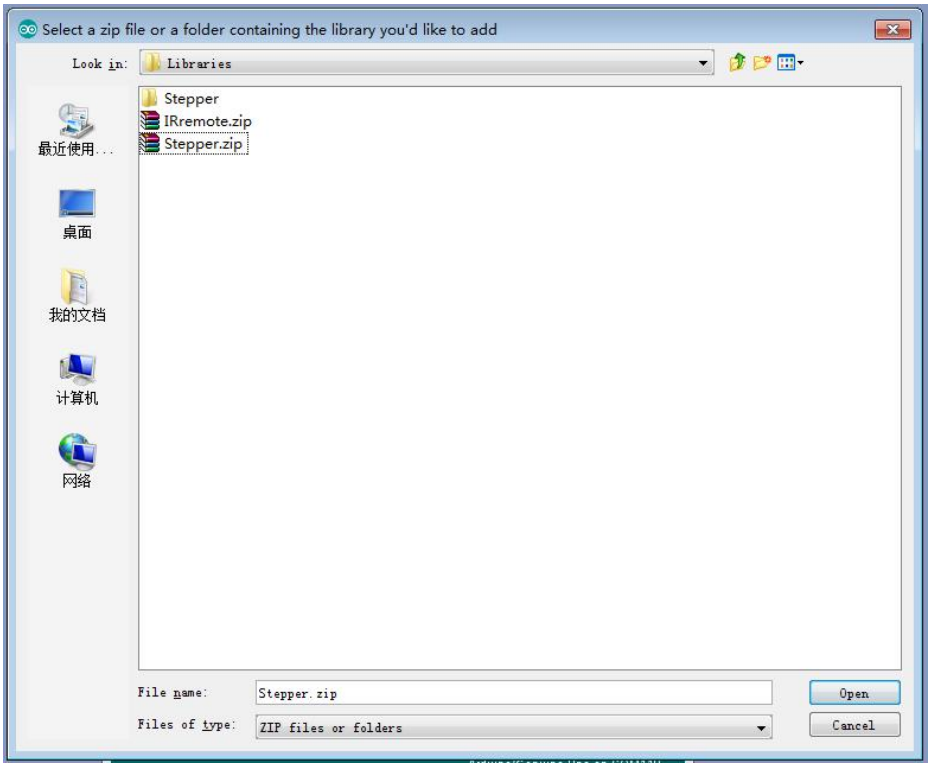
Step 2:Open the code:

Controlling_Stepper_Motor_With_Remote_Code



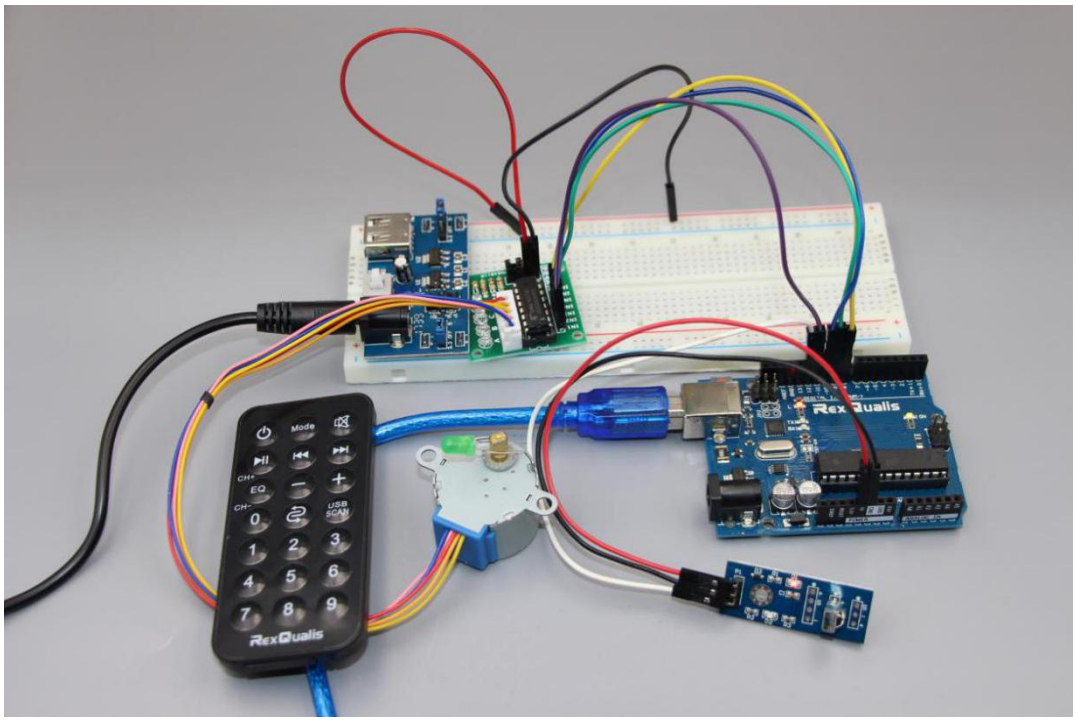
Step 3:Attach Arduino UNO R3 board to your computer via USB cable and check that the 'Board Type' and 'Serial Port' are set correctly.

Step 4:Load the Library:Stepper(IRremote Already loaded in last lesson)



Step 5:Upload the code to the RexQualis UNO R3 board.

Then you can control the motor left and right by remote control and infrared control.



You can see the video of the experiment results on YouTube: https://youtu.be/yTYlL710x3U

If it isn’t working, make sure you have assembled the circuit correctly, verified and uploaded the code to your board.For how to upload the code and install the library, check Lesson 0 Preface.