CS3514 Laboratory Session:

Name: Nathan Crowley

Student ID: 118429092

Question: Remote Controll

Repeat the following sections for each question

Answer:

I start by including the Irremote library. Then I initialise the pin I have the IR sensor connected to, Pin 7. The IR sensor is set as an IR reciever and the decoded results set to results.

In my setup function, I initialise my serial monitor to 9600 bits per second data rate. Aswell as initalise the IR sensor to recieve IR signals.

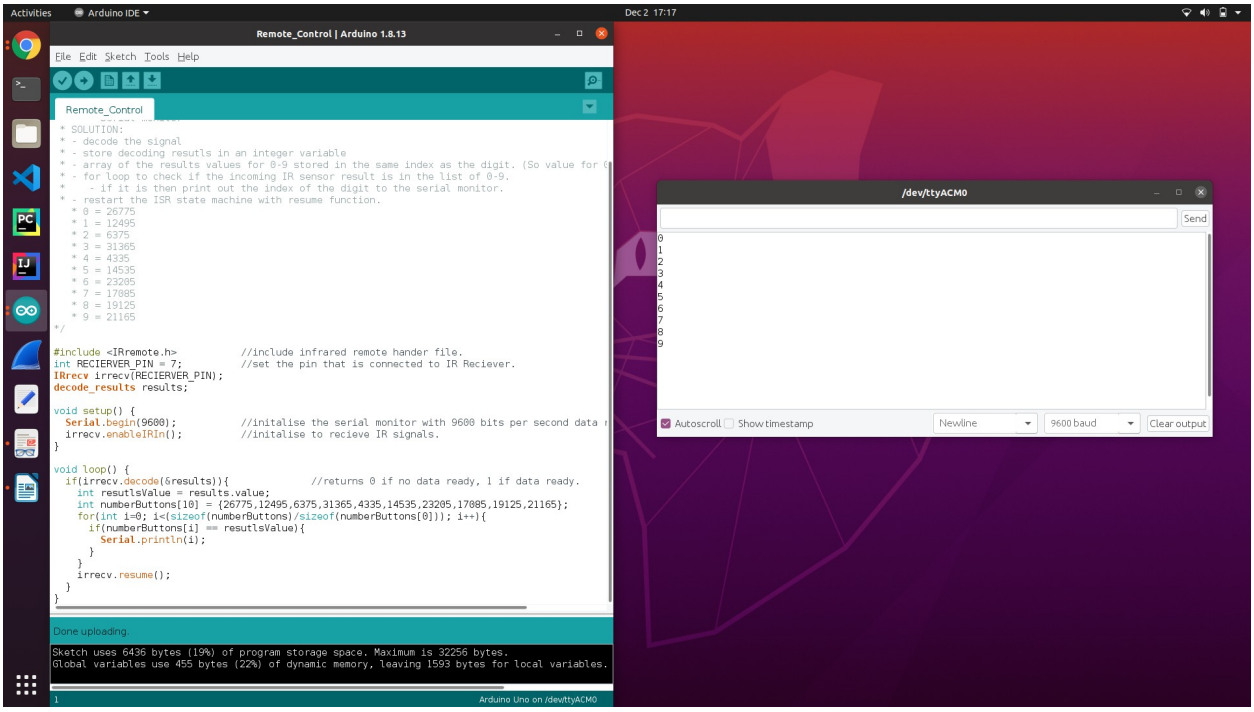
In my loop function, if there is data input detected, save the input signal values to an integer. I then set an array of integers that hold the values that are given when a digit (0-9) on the remote. A for loop iterates through this array and checks if the input signal matches any integer in the array, if it matches I print the index, and therefore the correct button pressed, to the Serial monitor. The IR sensor is reset at the end as to allow more communication.

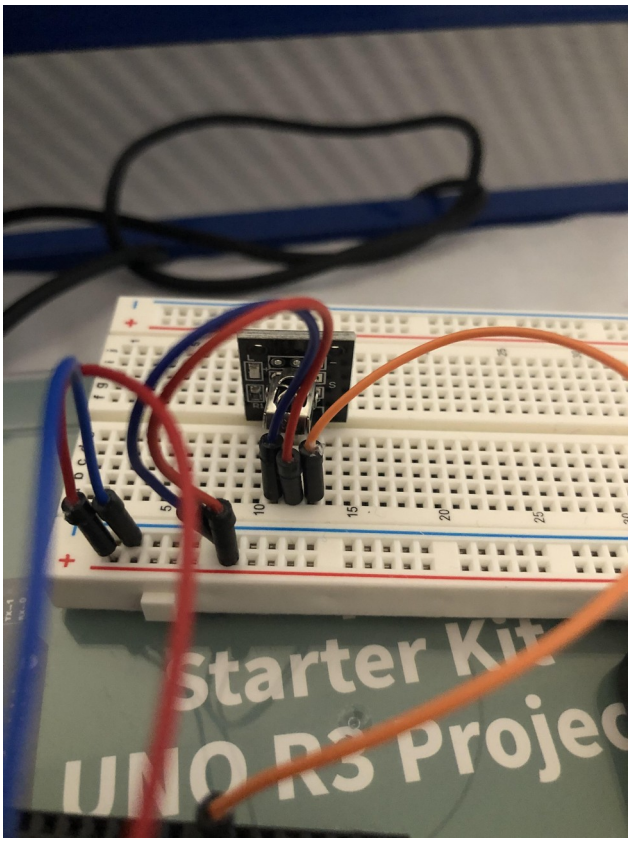
Code:

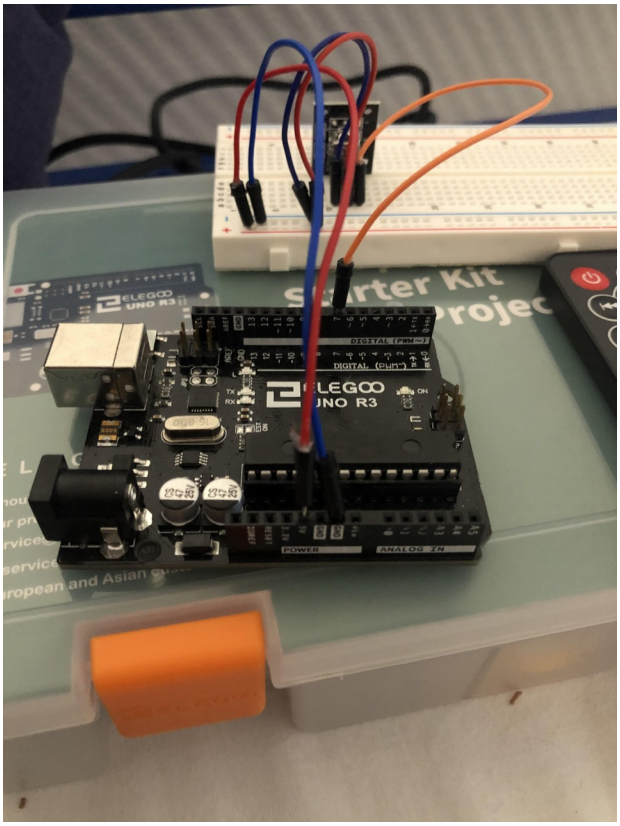
```
/*
 * TASK: write a program and build circuit to capture button presses fro  the IR Remote.
 *      if one of the remote buttons is pressed, the corresponding number should be wirtten to the
 *      Serial monitor
 * SOLUTION:
 * - decode the signal
 * - store decoding resutls in an integer variable
 * - array of the results values for 0-9 stored in the same index as the digit. (So value for 0 is at index 0)
 * - for loop to check if the incoming IR sensor result is in the list of 0-9.
 *    - if it is then print out the index of the digit to the serial monitor.
 * - restart the ISR state machine with resume function.
```

```
   * 0 = 26775
   * 1 = 12495
   * 2 = 6375
   * 3 = 31365
   * 4 = 4335
   * 5 = 14535
   * 6 = 23205
   * 7 = 17085
   * 8 = 19125
   * 9 = 21165
*/

#include <IRremote.h>           //include infrared remote hander file.
int RECIERVER_PIN = 7;          //set the pin that is connected to IR
Reciever.
IRrecv irrecv(RECIERVER_PIN);
decode_results results;

void setup() {
  Serial.begin(9600);           //initalise the serial monitor with 9600 bits
per second data rate.
  irrecv.enableIRIn();          //initalise to recieve IR signals.
}

void loop() {
  if(irrecv.decode(&results)){              //returns 0 if no data ready, 1 if
data ready.
    int resutlsValue = results.value;
    int numberButtons[10] =
{26775,12495,6375,31365,4335,14535,23205,17085,19125,21165};
    for(int i=0; i<(sizeof(numberButtons)/sizeof(numberButtons[0])); i+
+){
      if(numberButtons[i] == resutlsValue){
        Serial.println(i);
      }
    }
    irrecv.resume();
  }
}
```

Photos:


Activities
Arduino IDE
Dec 2 17:17
Remote_Control | Arduino 1.8.13
File Edit Sketch Tools Help
Remote_Control
* SOLUTION:
* - decode the signal
* - store decoding resutls in an integer variable
* - array of the results values for 0-9 stored in the same index as the digit. (So value for 0
* - for loop to check if the incoming IR sensor result is in the list of 0-9.
* - if it is then print out the index of the digit to the serial monitor.
* - restart the ISR state machine with resume function.
* 0 = 26775
* 1 = 12495
* 2 = 6375
* 3 = 31365
* 4 = 4335
* 5 = 14535
* 6 = 23205
* 7 = 17085
* 8 = 19125
* 9 = 21165
*/
#include <IRremote.h> //include infrared remote hander file.
int RECIERVER_PIN = 7; //set the pin that is connected to IR Reciever.
IRrecv irrecv(RECIERVER_PIN);
decode_results results;
void setup() {
Serial.begin(9600); //initalise the serial monitor with 9600 bits per second data r
irrecv.enableIRIn(); //initalise to recieve IR signals.
}
void loop() {
if(irrecv.decode(&results)){ //returns 0 if no data ready, 1 if data ready.
int resutlsValue = results.value;
int numberButtons[10] = {26775,12495,6375,31365,4335,14535,23205,17085,19125,21165};
for(int i=0; i<(sizeof(numberButtons)/sizeof(numberButtons[0])); i++){
if(numberButtons[i] == resutlsValue){
Serial.println(i);
}
}
irrecv.resume();
}
}
Done uploading.
Sketch uses 6436 bytes (19%) of program storage space. Maximum is 32256 bytes.
Global variables use 455 bytes (22%) of dynamic memory, leaving 1593 bytes for local variables.
1
Arduino Uno on /dev/ttyACM0
/dev/ttyACM0
Send
0
1
2
3
4
5
6
7
8
9
Autoscroll
Show timestamp
Newline
9600 baud
Clear output


Starter Kit
UNO R3 Projec


LEGOO
UNO R3