SASTA OLIPHANT SCIENCE AWARDS

SOUTH AUSTRALIAN SCIENCE TEACHERS ASSOCIATION

Prize Winner

Programming, Apps & Robotics

Year 3-4

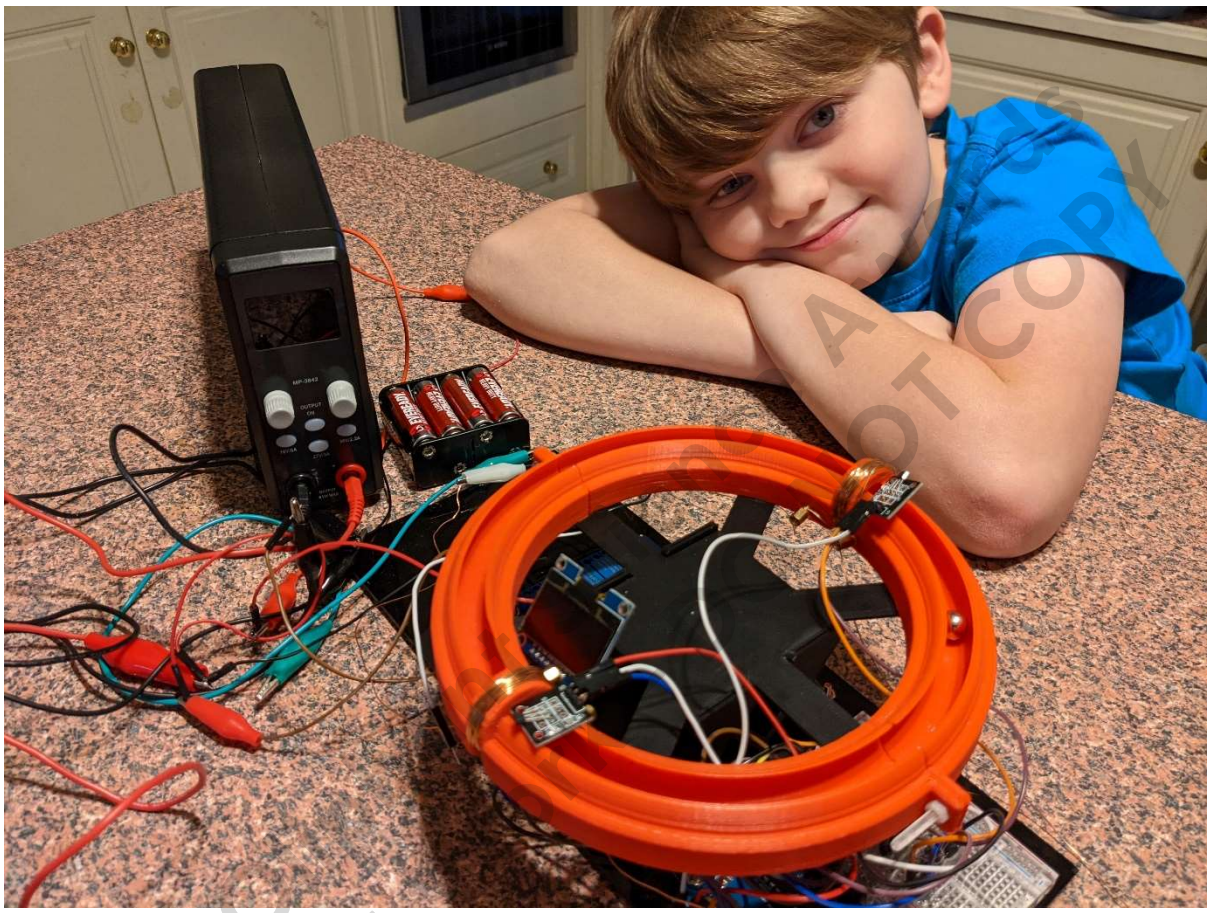
Jackson Burford

St John's Grammar School - Junior School

Model Particle Accelerator

By Jackson Burford

Oliphant Science Awards 2021



Introduction

The Particle Accelerator Model will accelerate a metal ball and measure the speed of the ball. The way it will do that is we will put the ball in the track next to an electromagnet. Once you do that you can turn on the Arduino and the Lab Power Supply. The electromagnet will turn on because the Hall Effect sensor will sense the ball, but the electromagnet will turn off once the ball has gone past. Then it will accelerate to the next proximity sensor then the next electromagnet will turn on then as the ball passes it will turn off and then it will all go back in a loop going over and over until we turn it off.

My Aim

My Aim is to make a Particle accelerator model that will show you how a Particle accelerator works and measure speed by using maths. The speed of the ball will show up on the OLED screen for us to see how fast the ball is going meters per second

How It Is Scientific

To do this I also needed to measure the circumference of the tube to get the correct distance. The speed only picks up when the Hall effect sensor senses the ball. The reason for that is because it's a lot easier to see and a lot neater with it only being picked up by only one Hall Effect Sensor.

The reason I did this project was because I thought making a particle collider simulator would be cool and I was inspired by the large hadron collider but after a while I realised, I didn't have enough time to make a particle collider simulator so I adjusted it to an accelerator simulator, but I might expand it to a collider next year.

First, I needed to learn about electromagnets. What I needed was copper, a battery and a core. I wrapped copper around the core leaving it with two ends. One end connected to the positive side of the battery and the other connected to the negative side of the battery to power the electromagnet.

Finally got up to the hard part and needed to make math equations to figure out the speed. I was using one Hall Effect Sensor to measure the speed. The equation was hard to work out but it made it easy to get the speed onto the arduino. The equation is speed = Distance / time. It was hard getting the code onto the OLED screen. I needed to use separate code then the original OLED screen code.

Making The Accelerator

First Electromagnet

The first thing I made was an electromagnet. First we went to the shops to get some copper. Then we needed to at home put the copper onto the nail and connect two ends of the copper to the battery and get it working. The first few times I made a mistake. Me and dad were the only people in the house who knew what we needed to do and how. However at home by myself when dad was out I managed to get it working.

The Electromagnets

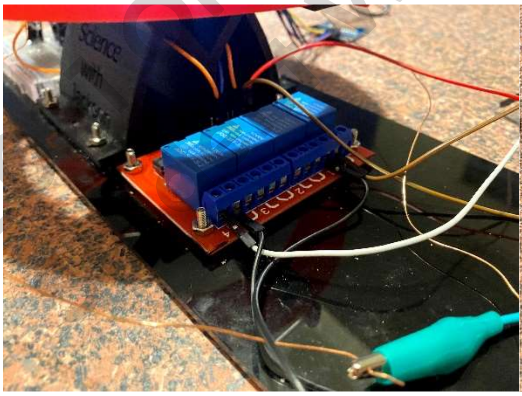
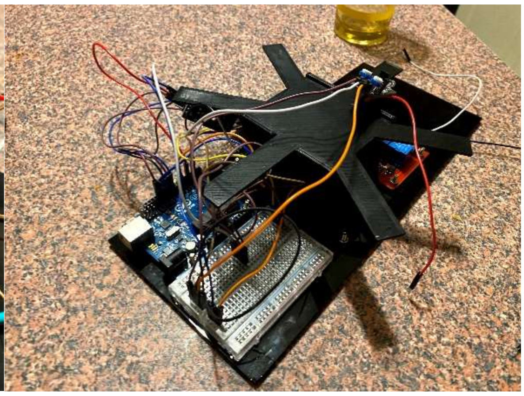
First, I needed to learn how electromagnets work so then I could go to the shops and get copper so when I went back home I got one of the nails then started wrapping copper around it then connected it to a battery, it took me a few goes and then I got it working. The things I needed for my first electromagnet were a nail, some copper, pliers and a battery.


The Code

I knew the next thing to do was to start the code for the electromagnet but at first, I had to replace the electromagnet with an LED. After about a night or 2 I got the LED working so what I did next was get another LED and get that working at the same time as the other one, so they were both flashing. The next step was to get 2 IR sensors working so if one IR sensor picks up something one LED would turn on and if the other IR sensor picks up something the other LED would pick up something. Then I needed to replace the sensors and small bits of the code.

Voltage – 5v or 12v?

At first everything but my electromagnets were being powered by 5 volts. But when it came to testing it wasn't working so I was trying to check why the electromagnets weren't working and it turned out it was because the relay needed to be powered by 12 volts. I fixed this by adding a battery pack giving 12 volts.

The Track

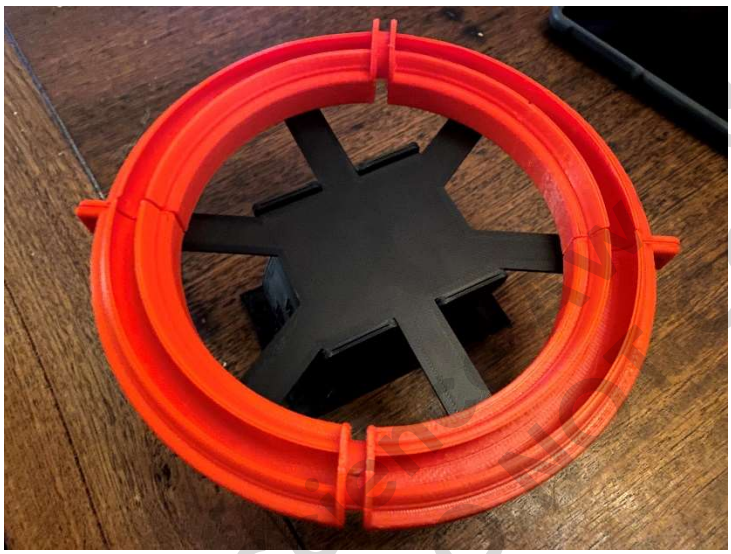
At first, I was going to use rubber tubing but then I realised it would be better to use a 3D printed track. The design was from Instructables, and I had it changed a little bit and then had someone print it. I didn't use their electronics or code at all. Once it was printed I needed to put it together and screw it on some plastic.

Circumference

I also needed to learn how to measure the circumference which did take me a while but then I figured it out and the equation is

$C = 2 \pi r$

Then I needed to start doing math and learn how to find the circumference of a circle. Dad gave me a few equations and I had to work out the circumference by only knowing the Diameter. Most of the time I figured it out but some of the time I still got it wrong.



What Sensors?

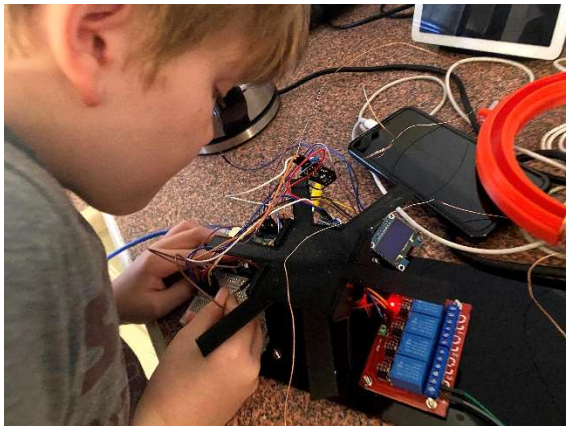
I realised I needed proximity sensors instead of IR sensors because -the IR sensors picked up the tube however the proximity sensor would only pick up the metal ball. Later I swapped to Hall Effect sensors because they pick up magnetic balls.

The Speed

Working out the speed was one of the most difficult parts because there was a lot of maths in it and I had to use the Hall Effect sensor to help measure the speed. The equation that made it easy was:

Speed = distance / time

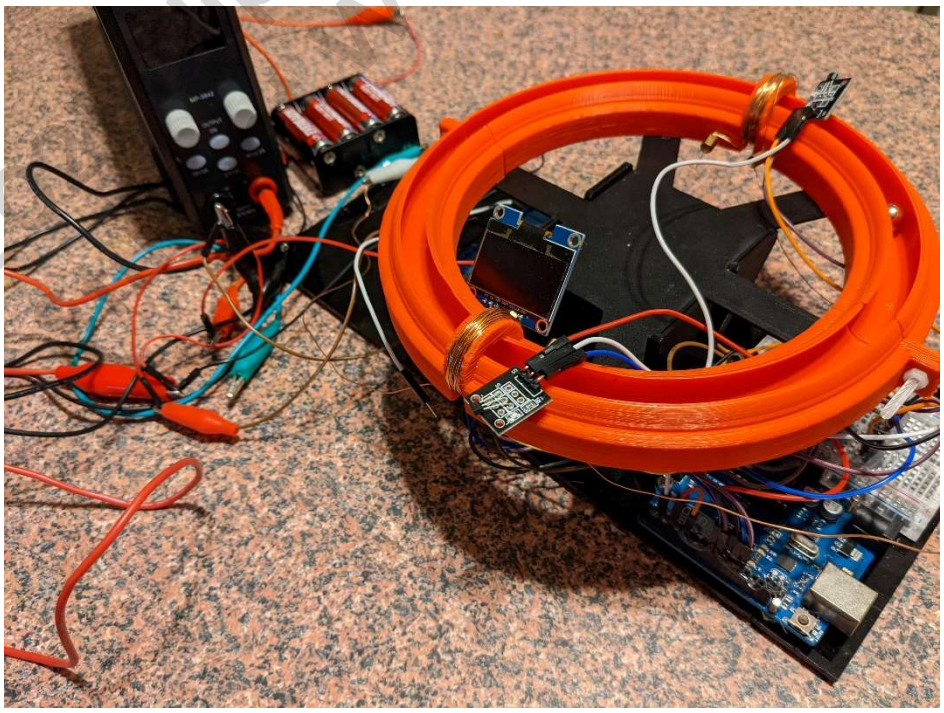
However I did need to make a lot of variables to make it work. I will highlight this part of the code in green. I also needed to get the speed onto an OLED screen which was challenging because instead of putting in normal OLED code I needed to put in a variable on the OLED screen printing the speed variable. I will highlight this part of the code in red.



Finishing Off

Now all I needed to do was to get the electromagnets working. I bought a Relay for my electromagnets and started connecting the electromagnet up. After a while I realised it would be better to get a 3D printed tunnel for the accelerator, so I found some designs on the internet and then edited it to be my own. When we got it printed, I went back home and started putting it together. For my electromagnets I decided I needed a Lab Power Supply that can go up to 36voltage to power the electromagnets.

When I connected the first electromagnet up it was ready to test but when I tested it it wasn't working because I was controlling the relay with the 5v and for a while we didn't know it needed 12v then we realised we needed to use a battery pack with 8 1.5v batteries. However, the next time we tested the IR sensor wasn't working, it took me a while, but I found out the IR sensor wasn't working probably because by the time we got the IR sensor lots of the sensors there were getting played with and must have broken. When we replaced the IR sensor it was working so I tested it and it worked. We decided that a hall effect sensor would be better to use then an IR sensor because the hall effect sensor would only pick up the ball. Next, I connected the next electromagnet and it worked so that's how I did my Project.



Mistakes

Through this project I have made a lot of mistakes and I had to do a lot of trouble shooting. Some of the time I forgot to manage my code and it would have been easier to just restart that part of the code instead of trying to fix it. Part of the time I was also using the wrong port. But most of the time it was a pinning error. I needed to really put the pins in exactly the right spot. Some of the time I just missed the spot by one pin or sometimes I forgot to connect the things up. But in the end, I fixed everything up.

Code

The code below is the code I used for this year's project and the green part of the code is the part with the speed and the red part is the on with the OLED parts of the code and the parts that are not highlighted is getting it so that when the sensor picks up something the relay will turn on.

```
#include "U8glib.h"
U8GLIB_SH1106_128X64 u8g(13, 11, 10, 9); // SCK = 13, MOSI = 11, CS = 10, A0 = 9

#define RELAY4 12
#define RELAY1 6
int pinIR = 2;
int pinIR2 = 4;
int led = 7;
float EventTime = 0;
float ElapsedTime = 0;
float TotalTime = 0;
float Distance = 50.26; //millimeters
float Speed = 0;
// ----------------------------------------
void draw(void) {
 // graphic commands to redraw the complete screen should be placed here
 u8g.setFont(u8g_font_unifont);
 //u8g.setRot180();
 //u8g.setFont(u8g_font_osb21);
 u8g.drawStr( 0, 22, "Speed:");
 u8g.drawStr( 80, 22, " m/s");
 u8g.setPrintPos(50, 22);
 u8g.print(Speed);
}
// ----------------------------------------
void setup(){
  Serial.begin(9600);
  pinMode(pinIR, INPUT);
  pinMode(pinIR2, INPUT);
  pinMode(RELAY4, OUTPUT);
  pinMode(RELAY1, OUTPUT);
  Serial.println("Detect Ball");

  // flip screen, if required
 u8g.setRot180();
 // set SPI backup if required
 //u8g.setHardwareBackup(u8g_backup_avr_spi);
 // assign default color value
 if ( u8g.getMode() == U8G_MODE_R3G3B2 ) {
 u8g.setColorIndex(255); // white
```

```
  }
  else if ( u8g.getMode() == U8G_MODE_GRAY2BIT ) {
  u8g.setColorIndex(3); // max intensity
  }
  else if ( u8g.getMode() == U8G_MODE_BW ) {
  u8g.setColorIndex(1); // pixel on
  }
  else if ( u8g.getMode() == U8G_MODE_HICOLOR ) {
  u8g.setHiColorByRGB(255,255,255);
  }
  delay(1000);
}
  //pinMode(led, OUTPUT);

void loop(){
  int IRstate = digitalRead(pinIR);
  if(IRstate == LOW){
    digitalWrite(RELAY4,HIGH); // Turns ON Relays 4
    EventTime = millis();
//    Serial.println(" ");
    Serial.print(" speed = ");
    Serial.print(Speed);
    Serial.print(" m/s ");
    Serial.println(" ");
      }
  else if(IRstate == HIGH){
    digitalWrite(RELAY4,LOW); // Turns Relay Off
    ElapsedTime = millis () - EventTime;
    Speed = Distance/ElapsedTime;
    u8g.setPrintPos(50, 22);
    u8g.print(Speed);
//    Serial.println(" ");
//    Serial.print(" speed = ");
//    Serial.print(" 0.00 ");
//    Serial.print(" m/s ");
  }
  int IRstate2 = digitalRead(pinIR2);
  if(IRstate2 == LOW){
    digitalWrite(RELAY1,HIGH); // Turns ON Relays 4
      }
  else if(IRstate2 == HIGH){
    digitalWrite(RELAY1,LOW); // Turns Relay Off
  }
  // LOOP CODE FOR OLED SCREEN
  // picture loop
  u8g.firstPage();
  do {
  draw();
  } while( u8g.nextPage() );
  // rebuild the picture after some delay
  // END OLED
  delay(1);
```

Pin Mapping

ARDUINO			BREADBOARD		
D2	PS1	Black	5F	A	V5
D4	PS2	Black	5G	Relay	VCC
D13	RELAY	IN4	5I	PS2	Brown
D6	RELAY	IN1	5H	PS1	Brown
5V	BB	5F	2F	A	GND
GND	BB	2F	2G	PS1	Blue
GND	RELAY	GND	2H	PS2	Blue
CLK	A	13			
MOS	A	11			
RES	A	reset			
DC	A	9			
CS	A	10			

RELAY			OLED SCREEN		
IN1	A	D6	Gnd	BB	2J
IN4	A	D13	VVC	BB	7H
VCC	BB	5G	CLK	A	13
GND	A	GND	MOS	A	11
Gnd	OLED	2J	RES	A	reset
VVC	OLED	7H	DC	A	9
			CS	A	10

REFERENCES

https://www.instructables.com/Particle-Accelerator-Demo/
**/I didn't use the code from this or their materials, I only used the design. I didn't 3D print it either.
https://www.instructables.com/Connecting-a-12V-Relay-to-Arduino/
https://www.instructables.com/Arduino-Stopwatch-1/
https://www.instructables.com/Hall-Sensor-Tutorial/
https://www.instructables.com/How-to-Use-the-IR-Obstacle-Avoidance-Sensor-on-Ard/
https://www.instructables.com/How-to-Control-an-Electromagnet-with-an-Arduino/
https://www.instructables.com/Coding-Timers-and-Delays-in-Arduino/
https://www.instructables.com/Arduino-Timer-Interrupts/
https://www.jaycar.com.au/medias/sys_master/images/images/9524871757854/XC4434-dataSheetMain.pdf
https://www.jaycar.com.au/medias/sys_master/images/images/9536761200670/XC4440-dataSheetMain.pdf
https://www.jaycar.com.au/medias/sys_master/images/images/9537052934174/XC4609-dataSheetMain.pdf
https://www.auselectronicsdirect.com.au/assets/files/TA0045%204%20channel%20Relay%20Module.pdf
https://mail.google.com/mail/u/0/#search/from%3Abrettjburford%40gmail.com/FMfcgxwKkRBWKKhxWTrXXkPGrcSrzDfp

PARENT'S REPORT

I am Jackson's father, and have helped guide him through this project. As always, I want to help make clear the parts Jackson did, and what help he was given.

This year, my only real contribution was as follows:

- Helping find learning resources
- Helping identify what components will be required
- Help in developing a structure/guide for code development
- Organising 3D printing of the track
- Financial (of course)
- Help identify occasional errors in code

I have not had direct input into the code except for help in finding errors. The most critical thing I did was to help Jackson understand how a simple implementation of a single sensor and an LED can be grown to multi-sensor, and then replace LEDs with electromagnets. Though he did require a little verbal direction with relays.

Jackson's approach to coding was initially haphazard which meant numerous 'start from scratch' scenarios. He would often find example code implementations for components online, and then would integrate this into his program and modify as required. The modification would often be extensive in this project, but in having a solid structure to begin with ensured he was able to properly understand the logic of his code.

It was especially challenging for him to implement a speed display, and it does require further improvements for accuracy.