Department of Electronic and Telecommunication Engineering
University of Moratuwa

Design Documentation

Multi-turn Absolute Magnetic Encoder

Epa Y.L.A.	210156U
Epa Y.R.A.	210157A

This report is submitted as a partial fulfilment of module EN2160

June 2024

Contents

Comprehensive Design Details

1 Introduction

Encoders are devices that convert one form of information into another. They work by detecting the rotation or linear motion of a shaft or axle and converting it into a signal that can be interpreted by a control system. This signal can be used to monitor the position, speed, and direction of the shaft or axle, and to control its motion with high precision.

There are several types of encoders, including incremental encoders, absolute encoders, rotary encoders, and linear encoders.

Incremental encoders provide information about the motion of the shaft, including its direction and the distance it has moved, while absolute encoders provide the exact position of the shaft at any given time. Rotary encoders are used for rotary motion and can be either incremental or absolute, while linear encoders are used for linear motion.

Encoders have a wide range of applications in various industries including manufacturing, automation, aerospace, healthcare and entertainment. They are used in robotics, CNC machines, motors, tachometers, user interfaces and many other systems that require precise control and monitoring of motion and position. Encoders help improve the efficiency, accuracy and safety of machines and systems and they play a crucial role in enabling automation and advanced manufacturing technologies.

1.1 Types of Encoders

- Rotary Encoders
  - Incremental Rotary Encoders: Measure the change in position from a reference point. They produce two square waves that help determine the direction and distance of rotation. These have a disc with alternating opaque and transparent sectors. As the disc rotates, an LED shines through the sectors, and a photo detector on the other side generates pulses. The number of pulses counted per unit time determines the speed, and the direction can be determined by the phase difference between two output signals.
  - Absolute Rotary Encoders: Provides a unique code for each position. These can determine the exact position without requiring a reference point. Also, these have a disc with multiple tracks each with a unique pattern. Each track corresponds to a binary bit of the position code. The combination of all tracks gives a unique binary code for each position.
- Linear Encoders
  - Incremental Linear Encoders: Similar to incremental rotary encoders but measures linear displacement. These encoders use a scale with equally spaced marks and a read head that moves along the scale. As the read head moves, it detects the marks and generates pulses.

  - Absolute Linear Encoders: Provide an absolute position along a linear path. These encoders use a scale with a unique pattern along its length and a read head that reads the pattern. The pattern provides a unique position code.

- Optical Encoders: These use light (typically LEDs and photodetectors) to detect position changes. Commonly used in high-precision applications.
- Magnetic Encoders: Use magnetic fields to detect position changes. They are robust in harsh environments and less sensitive to dust and other contaminants.

From these types, we are designing a multi turn absolute magnetic encoder.

1.2 Uses of Encoders

Encoders are used in various fields for different applications. Below are some common ones.

- Industrial Automation
  - Position and speed control in machinery.
  - Feedback for robotic arms and CNC machines.
- Consumer Electronics
  - Volume controls in audio equipment.
  - User interface controls in various devices.
- Automotive Applications
  - Steering angle sensors.
  - Engine position sensors.
- Aerospace
  - Position feedback in control systems.
  - Monitoring the position of control surfaces.
- Medical Equipment
  - Position sensing in imaging equipment.
  - Feedback in robotic surgery tools.

This section includes a detailed description of the electronic components and circuitry of multi-turn absolute magnetic encoder.

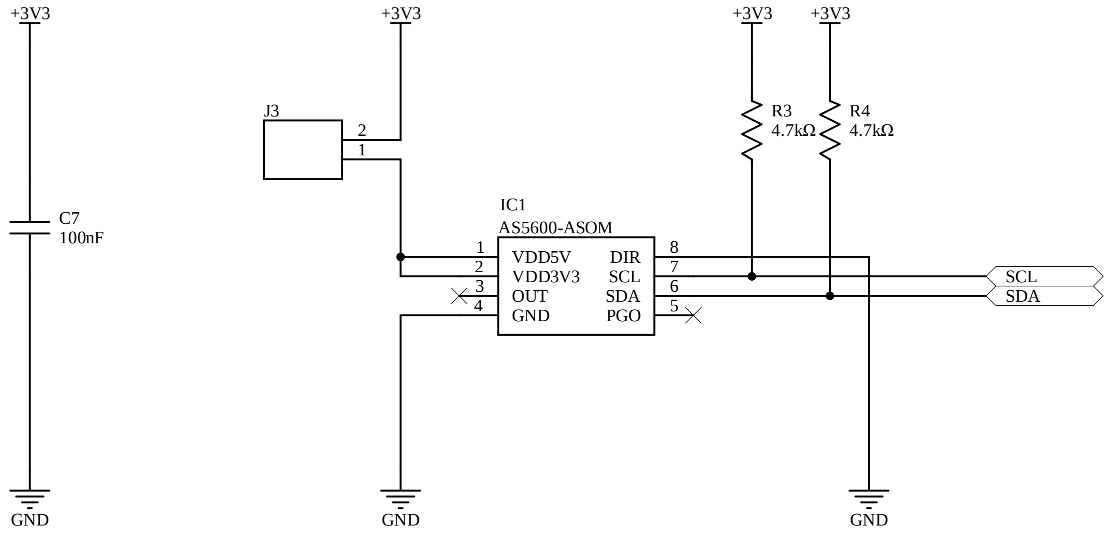
2 Electronics Documentation

2.1 Components Selection

Our main components include selecting a hall effect sensor IC, a magnet for the placement on the shaft, a microcontroller and power management circuit components which includes a battery and power regulation and management ICs.

2.1.1 Sensing Circuit

Sensing Circuit



Title	Hall Effect IC		
Size A4	Number 1		Revision 1
Date:	6/23/2024	Sheet 3 of 6	
File:	F:\Altium Projects\..\Hall Effect IC.SchDoc	Drawn By:	Yasoda Lasiru

Through the sensing circuit, we detect the magnetic field change of a diametrically magnetized magnet using hall effect sensor IC which is placed on axis to the magnet. Important information such as the specifications of the magnet to select, according to the hall effect sensor IC, the air gap between the IC and the magnet etc. are presented below. They were obtained after a thorough analysis of the datasheet of the select hall effect sensor IC - AS5600 - which is manufactured by a leading company in the field - AMS OSRAM.

AS5600 hall effect sensor IC works in 3.3V. For that we have to connect VDD5V and VDD3V3 together and connect to 3.3V. A separate power isolation jumper J3 is used here so that we can disconnect the hall effect IC from the ATMega328p while we are programming the microcontroller. To pull up the I2C lines SCL and SDA, we have used R3 and R4 which are 4.7kΩ . This value is selected as per the datasheet of AS5600. Since we are reading the angle value through I2C we do not need to use OUT pin - which outputs the angle value in a PWM signal. PGO pin is used to enter the programming mode of the AS5600. Since we are using the default configurations, we won't be needing that pin as well. The C7 100nF capcitor is use between the power lines according to the recommendation of the datasheet.

AS5600 Hall Effect Sensor

AS5600 is a Hall-effect based rotary magnetic position sensor. It uses planar sensors that convert the magnetic field component perpendicular to the surface of the chip into a voltage. The signals coming from the Hall sensors are first amplified and filtered before being converted by the analog-to-digital converter (ADC) inside the IC.

This sensor IC measures the absolute angle of a diametric magnetized on-axis magnet. The chip contains 4 hall effect sensors arranged in a circle. Together, they can detect changes in the direction of the magnetic field.

- Resolution of 12 bits
- temperature range of -40°C to 125°C
- supply voltage range -0.3V to 6.1V
- power continuous consumption of 50mW at 70°C
- Analog and PWM outputs
- PWM output frequency upto 920Hz
- I2C interfacing option
- programmable start and zero angle
- automatic entry into low power mode operation

There are various hall effect sensor ICs in the global market to choose from. Below are the criteria which led to the selection of the AS5600 magnetic hall effect sensor IC.

- Output Resolution: The AS5600 has a 12-bit digital to analog conversion (DAC) output resolution
- Maximum Angle Programmable: You can program the maximum angle from 18 to 360 degrees

- Selectable Output: The AS5600 provides an analog output ratiometric to VDD or PWM-encoded digital output
- Low-Power Consumption: The AS5600 has a smart low power mode feature to automatically reduce the power consumption
- Easy Setup: The AS5600 has automatic magnet detection
- Robust Environmental Tolerance: The AS5600 is qualified for a temperature range from -40°C to 125°C
- Magnet Airgap: The typical airgap is between 0.5 mm and 3 mm, and it depends on the selected magnet. A larger and stronger magnet allows a larger airgap

Magnet

According to the AS5600 data-sheet, the recommended specifications of the magnet are as follows.

- Diametrically magnetized: cylindrical
- Any one of diameter 6mm and thickness of 2.5mm, diameter 8mm and thickness of 3mm or diameter 8mm and thickness of 4mm
- magnetic input range should be between 30-90mT

  We have selected a cylindrical neodymium magnet of diameter 6mm and thickness of 2.5mm.

2.2.1 Microcontroller Circuit

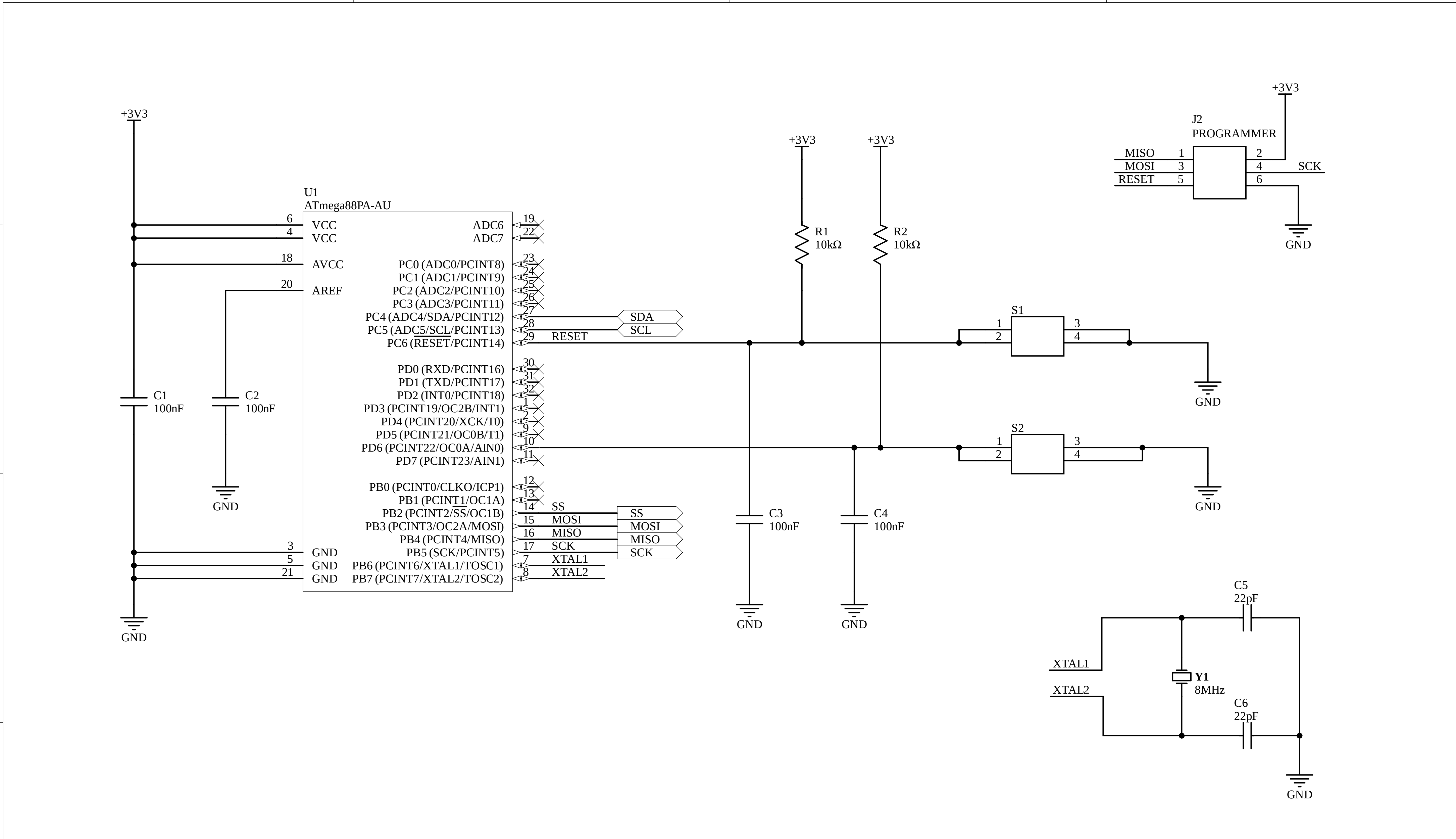
We have used a switch S1 to reset the microcontroller. It's pulled up through R1 10kΩ resistor and connected to ground through C3 100nF capacitor.

Also, we have used another switch in the circuit, which is used to manually tare the turned angle, used for debugging purposes. Same as S1, it's pulled up through R2 10kΩ and connected to ground through C4 100nF capacitor.

J2 is the programming port which is used to program the microcontroller. These SPI pins are also connected separately to the communication port, which is available to the user. Since we aim to minimize power consumption, we are using Y1 8MHz crystal oscillator as per the recommendation of the datasheet. This is connected to ground through two 22pF capacitors - which are also chosen according to the datsheet.

We have also used 100nF capacitors here to suppress high frequency noise and stabilize the voltage.

Microcontroller Circuit



Title	Microcontroller	
Size: A4	Number: 1	Revision: 1
Date: 6/23/2024		Sheet 2 of 6
File: F:\Altium Projects\..\Microcontroller.SchDoc		Drawn By: Yaseema Rusiru

ATMega328P-AU

We are using ATMega328P-AU as the micro-controller for our multi turn absolute magnetic encoder. It is a low power consuming microcontroller which can implement both SPI and I2C communication protocols at the same time, while saving power. It interfaces with the AS5600 sensor module to process position data and generate output signals. This includes the absolute angle and the algorithm to count the number of turns.

- 8 bit AVR RISC based
- 8KB flash memory 512B EEPROM and 1KB SRAM
- 10 bit ADC
- 8 bit timer/counter with prescalar and two PWM channels
- Internal calibrated oscillator

We have considered the following criteria when selecting thee microcontroller.

- CPU Architecture: The ATMega328P-AU is based on the AVR RISC architecture1. This architecture is known for its high performance and power efficiency
- Memory: The ATMega328P-AU has 32KB of in-system programmable flash memory, 1KB of EEPROM, and 2KB of internal SRAM. These are required for our application
- Operating Voltage: The ATMega328P-AU operates between 1.8-5.5V. We opted for 3.3V in order to run at a speed of 8MHz.
- Power Efficiency
- Seperate I2C and SPI bus availability to read AS5600 IC data while also giving the angle output to the user through SPI interface
- Availability

2.3.1 Power Management

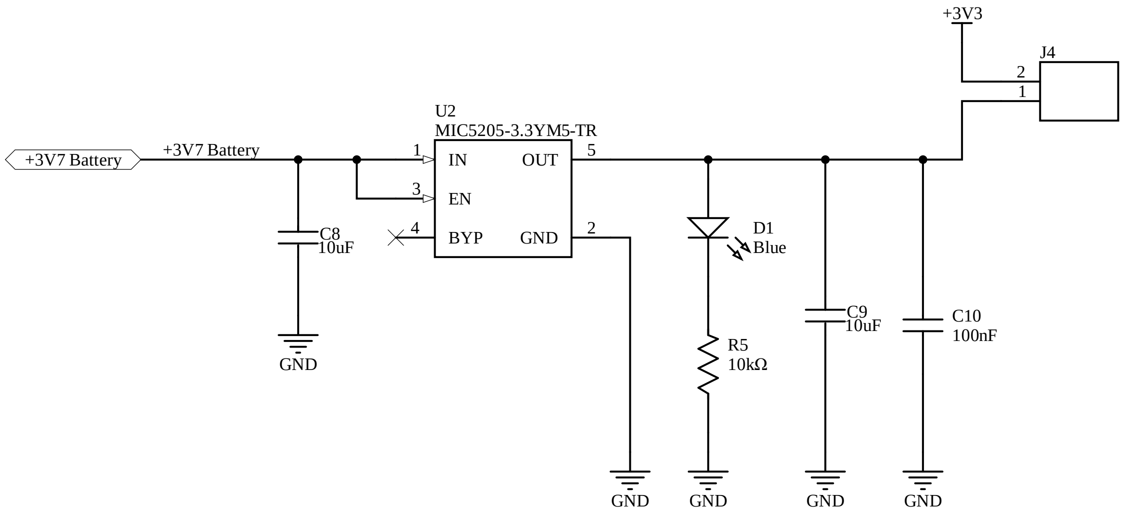
As the main power supply for the hall effect sensor IC and the microcotroller, the user should input an external power source within the given voltage range (3.4V to 8V). Through this supply, the rechargeable battery charges, allowing the system to utilize its' power even during power outages. For this we have implemented a separate charging circuit with over current and voltage protection.

Since we intend to use the microcontroller and the hall effect sensor IC with 3.3V supply, a voltage regulator is also to be used as part of the power management circuit. Moreover, we have utilized low power modes in both the microcontroller and the hall effect sensor IC in order to ensure that the power used is minimized. Furthermore this sub assembly consists of TP4056 and FS8205A ICs.

Voltage Regulator Circuit

J4 power isolation connector is used to isolate the microcontroller from the LDO while programming. So, we should power the microcontroller through the programming port while programming.

Voltage Regulator Circuit



Title: Voltage Regulator		
Size: A4	Number: 1	Revision: 1
Date: 6/23/2024		Sheet 4 of 6
File: F:\Altium Projects\..\Voltage Regulator.SchDoc		Drawn By: Yasoda Lasiru

We have used D1 blue diode to indicate whether power is available - specifically 3.3V which is available at the OUT pin. Here we have used 10kΩ resistor to minimize the current used by the diode. The datasheet of MIC5205-3.3YM5-TR LDO specifically mentions to use a capacitor value greater than 2.2 uF between ground and the OUT pin.

Since enable/shutdown is not required in our application, we have connected IN and EN pins (according to the datasheet).

Since noise is not a major concern BYP pin is not connected according to the datasheet. To handle the input voltage fluctuations, C8 10uF capacitor is used.

MIC5205-3.3YM5-TR Low-dropout Voltage Regulator

The MIC5205-3.3YM5-TR is a low-dropout (LDO) voltage regulator. It is designed to deliver a fixed output of 3.3V. This LDO regulator is commonly used in battery-powered devices due to its low quiescent current of 10nA and it provides good line and load regulation characteristics.

- Operating temperature range -40 - 125°C
- 150mA typical output current
- Output Voltage Accuracy: The MIC5205-3.3YM5-TR offers better than 1% initial accuracy
- Low Dropout Voltage: The MIC5205-3.3YM5-TR as a very low dropout voltage (typically 17 mV at light loads and 165 mV at 150 mA)12. This low dropout voltage allows the regulator to operate efficiently even when the input voltage is only slightly above the output voltag
- Temperature Coefficient: The MIC5205-3.3YM5-TR has a very low temperature coefficient
- Load and Line Regulation: The MIC5205-3.3YM5-TR provides extremely tight load and line regulation

Battery Charger Circuit

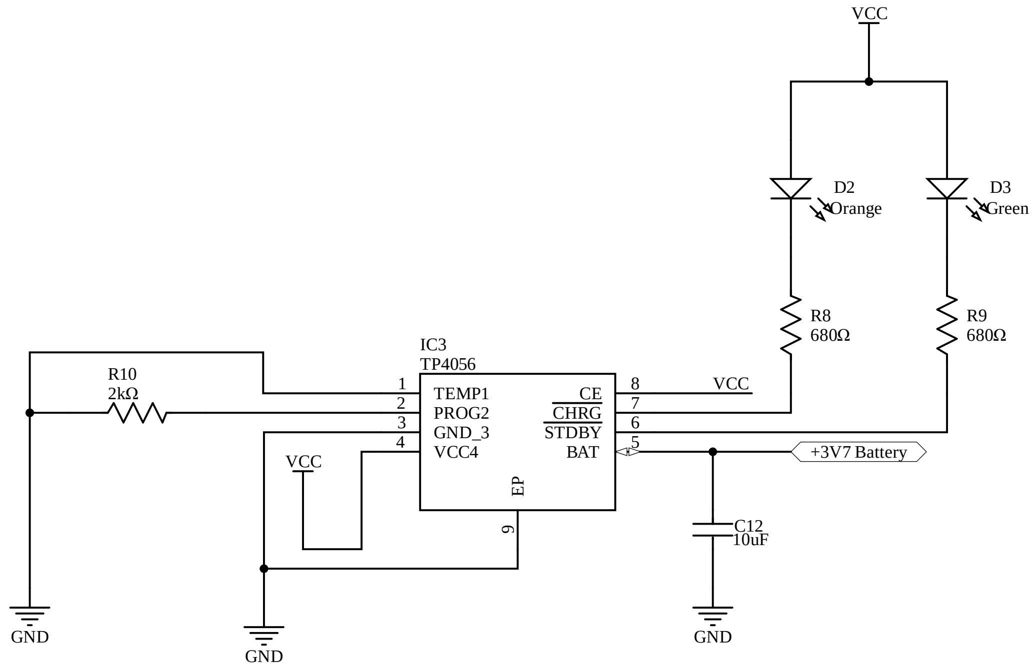
A high input in the CE pin will put the TP4056 in the normal operating mode. Hence, we have attached VCC to CE pin. The positive terminal of the battery should be connected to the BAT pin.

PROG pin is used to set the constant charge current and it can be used to monitor the charge current. This pin is grounded through a 2kΩ resistor. It sets the charging current which is 580mA according to the datasheet. BAT pin provides charge current to the battery and provides regulation voltage of 4.2V. When the voltage of the VIN pin drops to within 30mV of the BAT pin voltage, TP4056 enters low power sleep mode. This drops the BAT pin's current to less than 2uA.

We can get an idea about the charging state of the battery through the intensity of the D2 and D3 LEDs.

Charge state	Orange LED	Green LED
charging	ON	OFF
Charge Termination	OFF	ON
Vin too low/ no battery	OFF	OFF

Battery Charger Circuit



Title: Battery Charger		
Size: A4	Number: 1	Revision: 1
Date: 6/23/2024		Sheet 6 of 6
File: F:\Altium Projects\..\Battery Charger.SchDoc		Drawn By: Yaseema Rusiru

TP4056 Constant Current Constant Voltage Linear Charger IC

The TP4056 is a standalone linear charger IC designed for single cell lithium-ion batteries. The charge current can be programmed externally with a single resistor. We have used 2kΩ for this. As per the datasheet, the charging current then becomes 580mA.

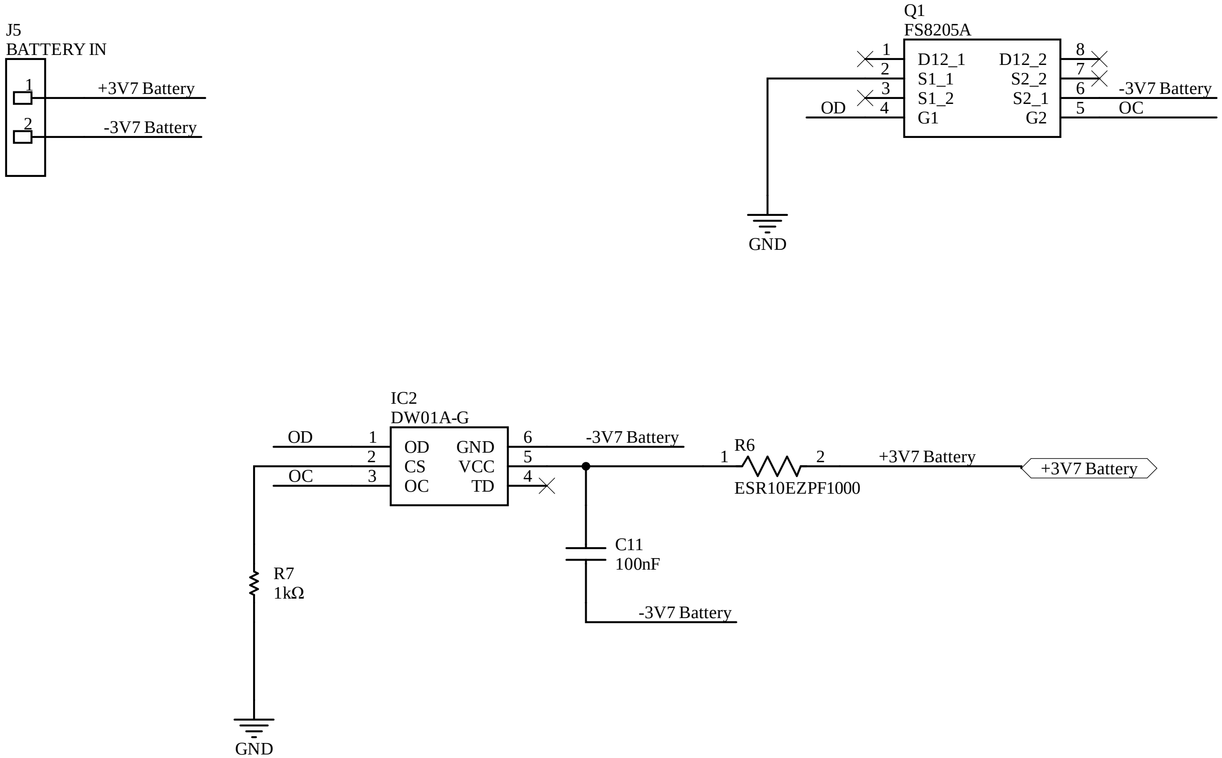
- Programmable charge current upto 1A
- Linear charger for single cell Lithiom Ion batteries
- Constant current
- Constant voltage
- Input supplt voltage -0.3V - 8V
- Maximum junction temperature 145
- Operating temperature range -40 - 85°C

Battery Protection Circuit

External battery can be connected to the circuit through J5. Positive and negative terminals of the battery are connected to VCC and GND pins through R6. C1 and R6 are selected according to the specifications given in the datasheet. R7 resistor value is also given in the datasheet. Since TD pin is used for testing purposes, it is not used. CS input pin is used for current sensing and detecting the charging.

Both S1_1 pin and S1_2 pin are the same. Both S2_1 pin and S2_2 pin are the same. G1 is the gate of one N channel MOSFET. G2 is the gate of the other N channel MOSFET. S1_1 is the source of the first MOSFET, S2_1 is the source of the second MOSFET. Hence these pins are grounded here. OD pin of DW01A-G is used to connect MOSFET gate for discharge control. OC pin is also connected to a MOSFET gate for charge control.

Battery Protection Circuit



Title: Battery Protection Circuit		
Size: A4	Number: 1	Revision: 1
Date: 6/23/2024		Sheet 5 of 6
File: F:\Altium Projects\..\Battery Protection.SchDoc		Drawn By: Yaseema Rusiru

FS8205A Dual N Channel Power MOSFET

This IC is used to ensure safe and efficient charging and discharging of the battery, preventing overcharging or excessive discharge which could damage the battery.

- Drain source voltage 20V
- Continuous drain current at 25°C - 6A
- Total power dissipation at 25°C - 1W
- Operating junction temperature range -55 to 150°C

DW01A-G Battery Protection IC

The DW01A-G is a one cell Lithium-ion/Polymer battery protection IC. It is designed to protect lithium-ion/polymer batteries from damage or degrading the lifetime due to overcharge, overdischarge, and/or overcurrent for one-cell lithium-ion/polymer battery powered systems. The DW01A-G ensures that the battery operates within its safe limits, thereby extending its lifespan and maintaining its performance

- Overcharge detection voltage 4.30±0.05V
- Overcharge release voltage 4.10±0.05V
- Overdischarge detection voltage 2.40±0.10V
- Overdischarge release voltage 3.00±0.10V
- Supply current at 3.9V VCC typical 3µA max 6µA

2.7 Functional Block Diagram

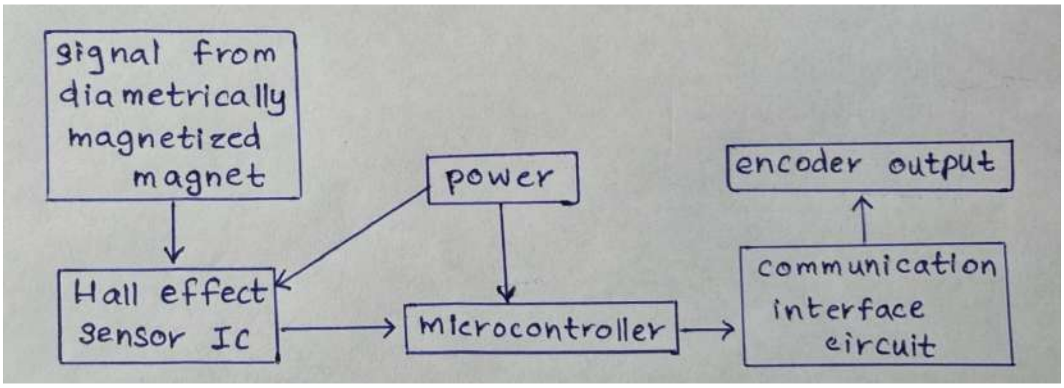
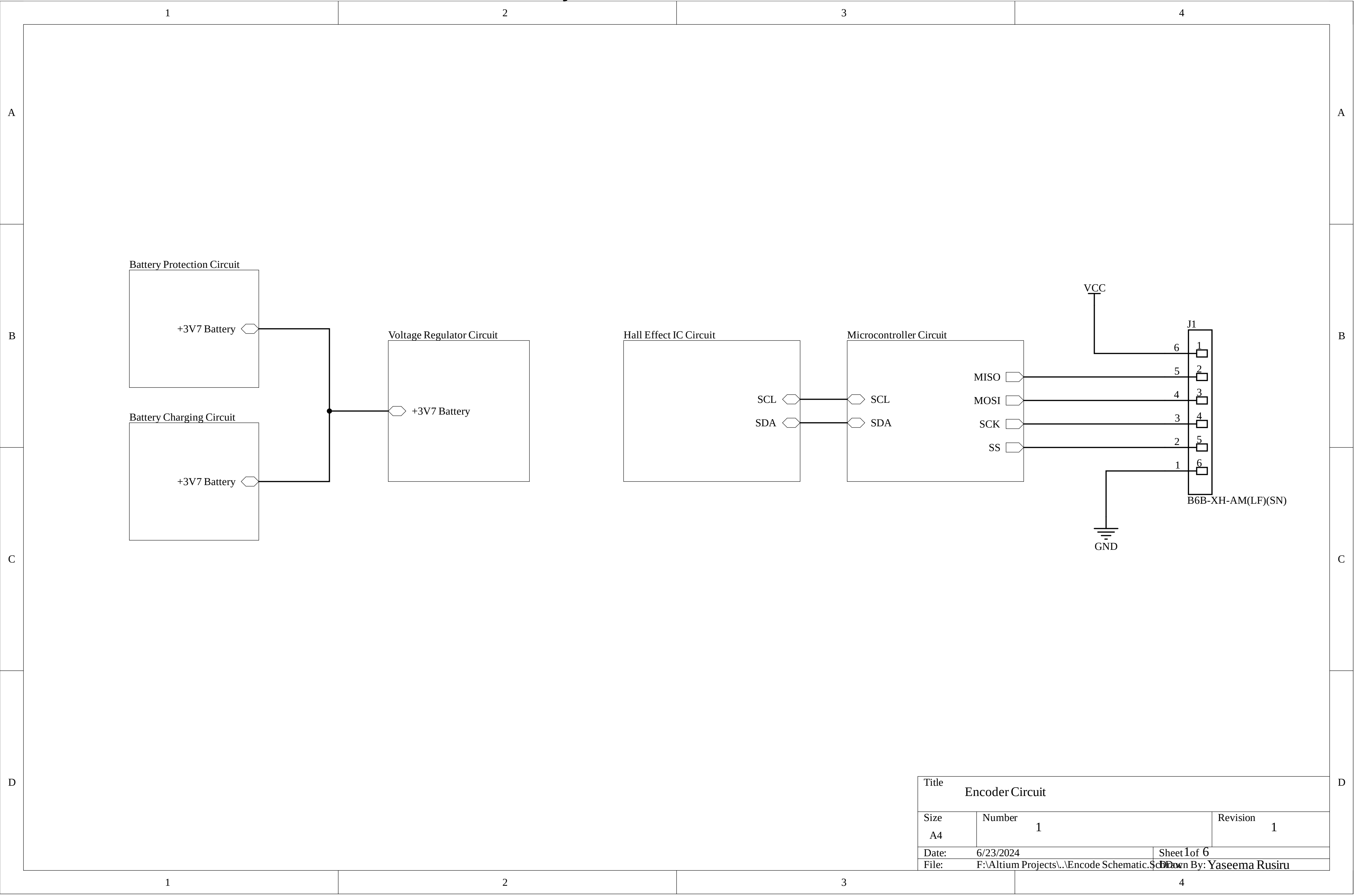


Figure 1: *Functional Block Diagram of the Selected Design*

2.8 System Overview

System Overview



Title: Encoder Circuit		
Size: A4	Number: 1	Revision: 1
Date: 6/23/2024	Sheet 1 of 6	
File: F:\Altium Projects\..\Encode Schematic.SchDoc	Drawn By: Yaseema Rusiru	

2.10 PCB Design

2.10.1 PCB Specifications

- Material: FR4
- Thickness: 1.6mm
- Copper Weight: 1 oz
- Radius : 18mm

2.10.2 2D Layout

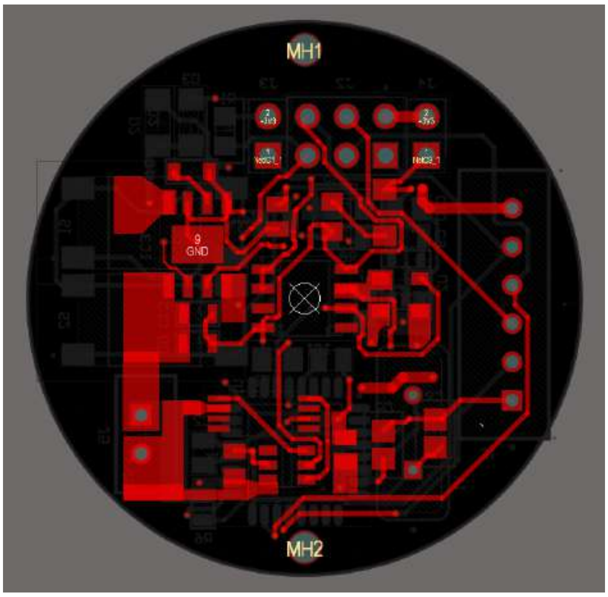


Figure 2: *Top Layer*

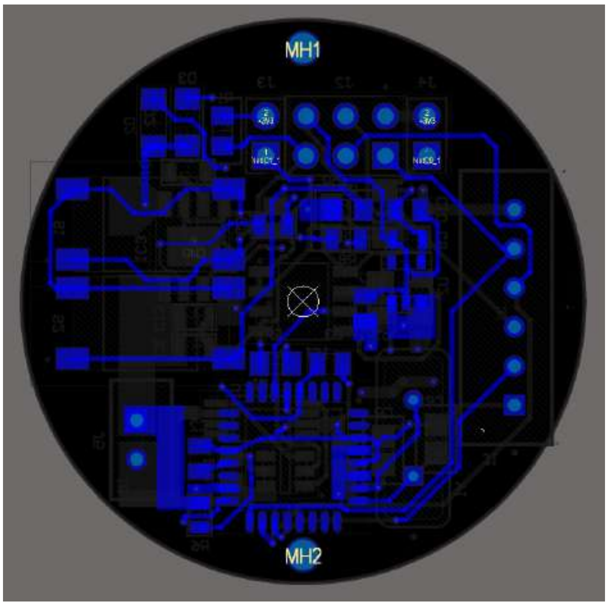


Figure 3: *Bottom Layer*

2.10.3 Before Soldering PCB

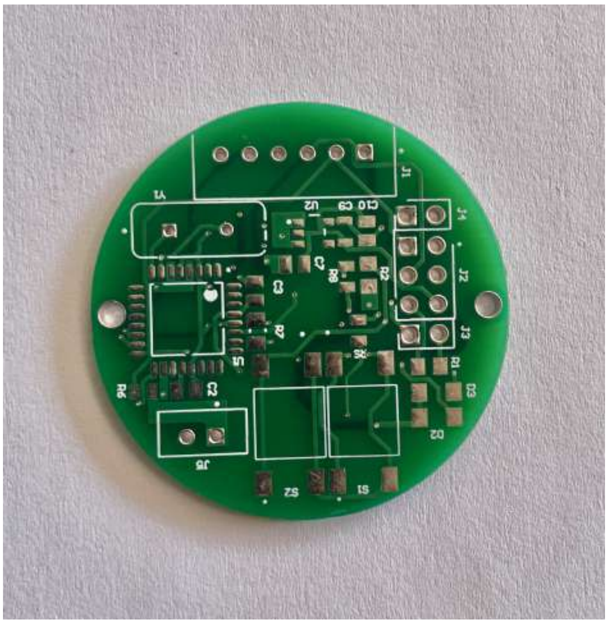


Figure 4: *Bare PCB - Top*

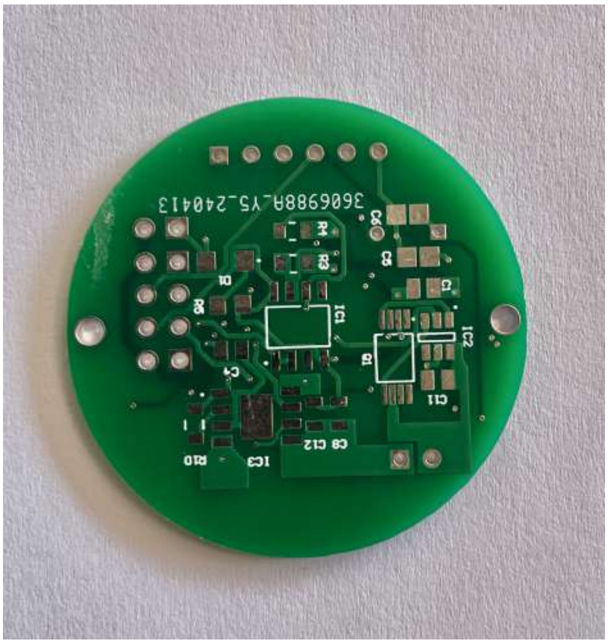


Figure 5: *Bare PCB - Bottom*

2.10.4 After Soldering PCB

Figure 6: *Soldered PCB*

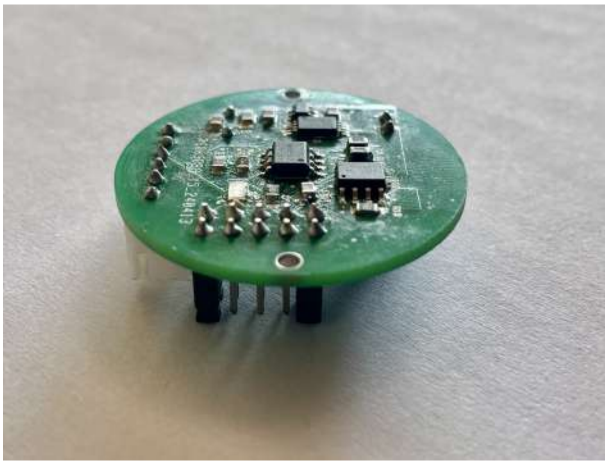


Figure 7: *Soldered PCB*

2.10.5 Photographs as evidence for the PCB testing

Here motor is controlled through raspberry pi and the encoder data is collected from the raspberry pi to be shown on the dashboard.

Figure 8: *Testing*

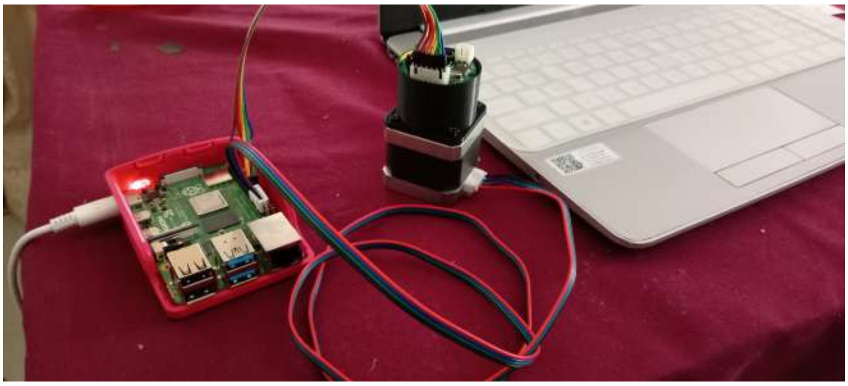


Figure 9: *Testing*

Figure 10: *Testing*

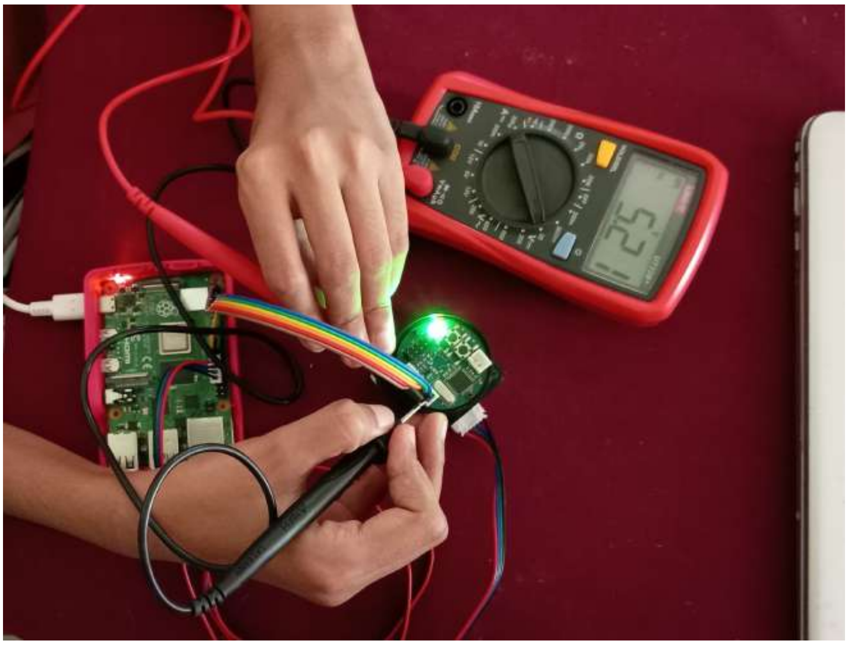


Figure 11: *Testing*

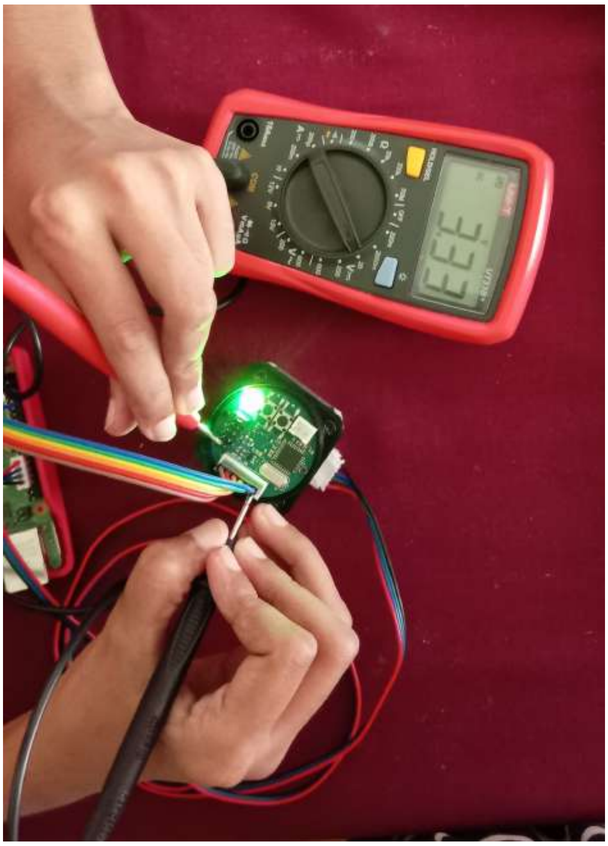


Figure 12: *Testing*

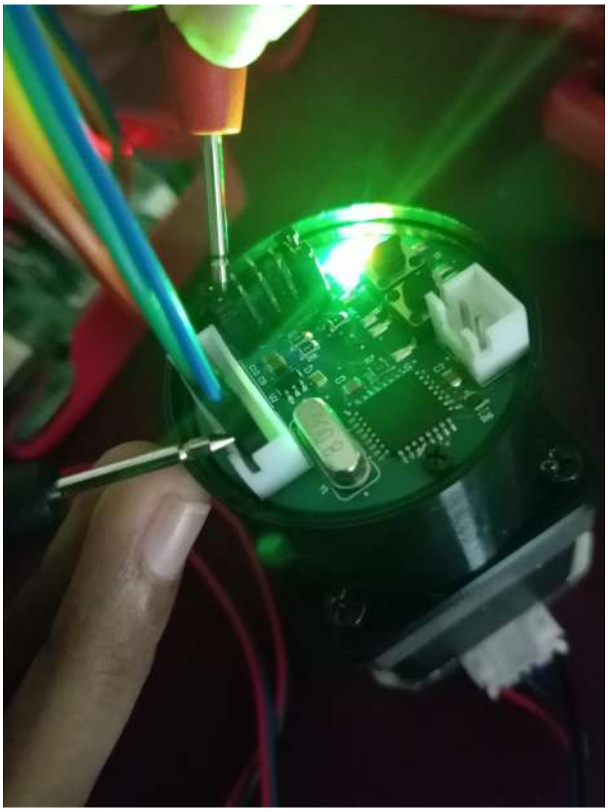


Figure 13: *Testing*

2.11 Bill of Materials

Designator	Description	Quantity
C1,C2,C3,C4,C7,C10,C11	16V 100nF X7R ±10% 0805 Multilayer Ceramic Capacitors MLCC - SMD/SMT ROHS	7
C5,C6	16V 22pF ±5% 0805 Multilayer Ceramic Capacitors MLCC - SMD/SMT ROHS	2
C8,C9,C12	16V 10uF X7S ±10% 0805 Multilayer Ceramic Capacitors MLCC - SMD/SMT ROHS	3
D1	25mA -30℃ +85℃ Positive post 460nm 469nm Blue 120° 80mW 1206 Light Emitting Diodes (LED) ROHS	1
D2	25mA -30℃ +85℃ 601nm 603nm Positive post Orange 120° 55mW 1206 Light Emitting Diodes (LED) ROHS	1
D3	25mA 516nm 525nm -30℃ +85℃ Positive post Emerald 120° 80mW 1206 Light Emitting Diodes (LED) ROHS	1
IC1	AS5600-ASOM	1
IC2	DW01A-G	1
IC3	TP4056	1
J1	CONN HEADER VERT 6POS 2.5MM	1
J2	Push-Pull,P=2.5mm Wire To Board Connector ROHS	1
J3,J4,J5	CONN HEADER VERT 6POS 2.5MM	3
Q1	FS8205A MOSFET(N CHANNEL)	1
R1,R2,R5	RES SMD 10K OHM 0.5% 1/8W 0805	3
R3,R4	RES SMD 4.7K OHM 0.5% 1/8W 0805	2
R6	RES SMD 100 OHM 0.5% 1/8W 0805	1
R7	RES SMD 1K OHM 0.5% 1/8W 0805	1
R8,R9	RES SMD 680 OHM 0.5% 1/5W 0805	1
R10	RES SMD 2K OHM 0.5% 1/8W	1
S1,S2	6mm 5mm Round Button 50mA Brick nogging 6mm SPST 12V SMD,6x6mm Tactile Switches ROHS	2
U1	8-bit Microcontrollers - MCU ATMEGA328PB 20MHZ IND TEMP	1
U2	IC REG LINEAR 3.3V 150MA SOT23-5	1
Y1	Crystals 8MHz 20pF	1

2.12 Design Specifications

The circuit operates at 3.3V. It can tolerate maximum 8V from the external source and this voltage also charges the battery. When the main power is not available, it switches the power source to battery. The MIC5205-3.3YM5-TR is a low-dropout (LDO) voltage regulator that outputs 3.3V. It requires an input voltage that's at least 3.4V to account for the dropout voltage. Therefore, a 3.7V battery will be sufficient. Below are the required calculations when selecting it.

Battery Capacity: To calculate the minimum capacity of the battery, we need the total current consumption of the circuit.

Current consumption of main components:

- AS5600 Hall Effect Sensor: 1.5mA (typical)
- ATMEL ATMega328P-AU: 3mA (maximum)
- MIC5205-3.3YM5-TR: negligible
- TP4056: negligible
- FS8205A: negligible
- DW01A-G: negligible

Total current consumption = 1.5mA + 3mA = 4.5mA

If the device need to operate for 24 hours without external power(battery life), it would need a battery with a capacity of at least 4.5mA x 24 hours = 96mAh. The battery can be connected by the user, so the required capacity can be decided as per the following table.

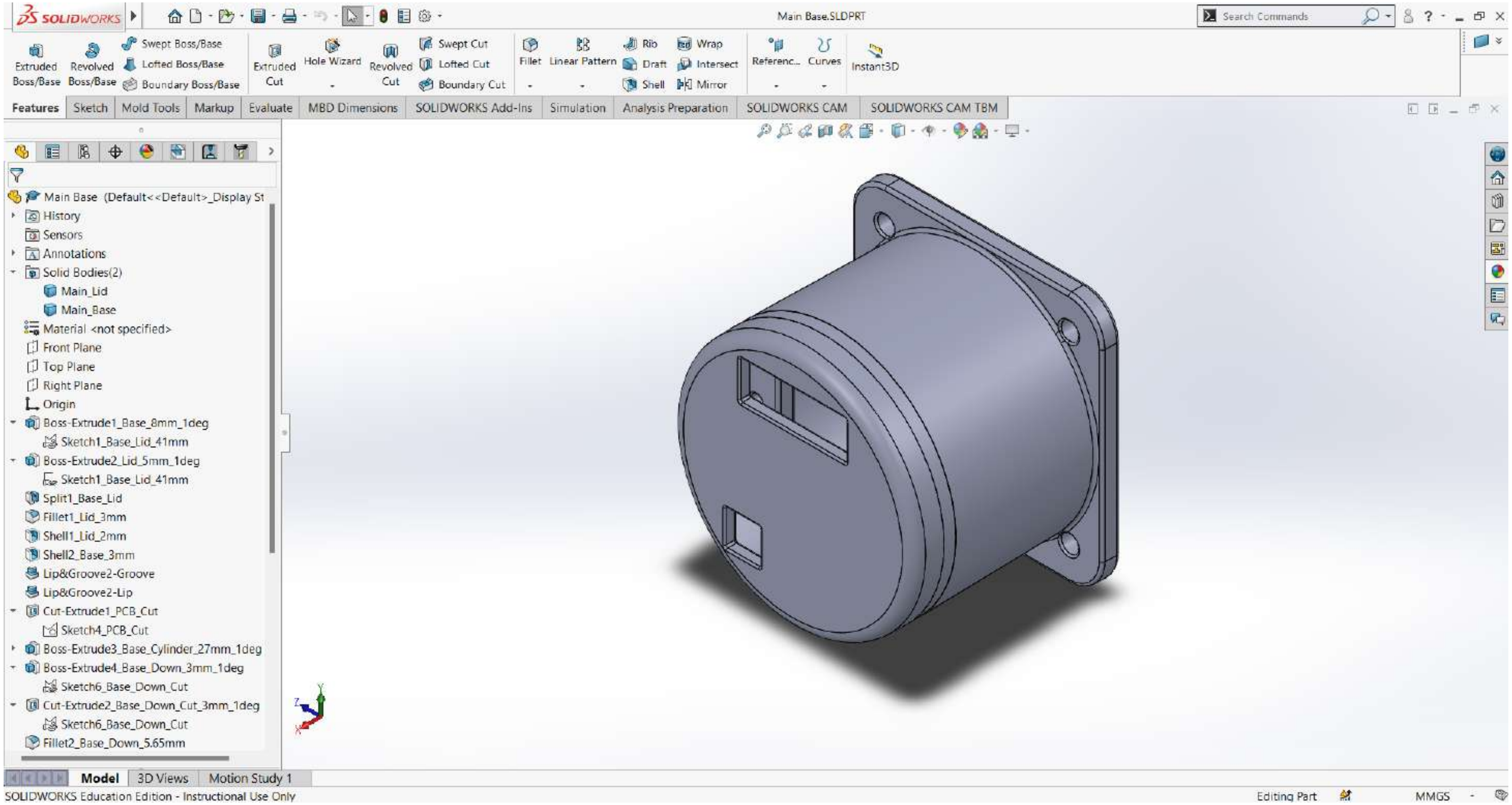
Duration time (approximate)	Battery Capacity
24 hours	96mAh
7 days	672mAh
1 month	2880mAh

The TP4056 charger IC decides the time taken for the battery to fully charge, by the external resistor connected to it. As per the datasheet of TP4056 charger IC we have used 2kΩ for that. So the charging current is documented as 580mA. As per the battery used by the user, the charging time varies. If the user takes 1800mAh 3.7V battery it will take 1800/580 = 3 hours approximately to fully charge the battery.

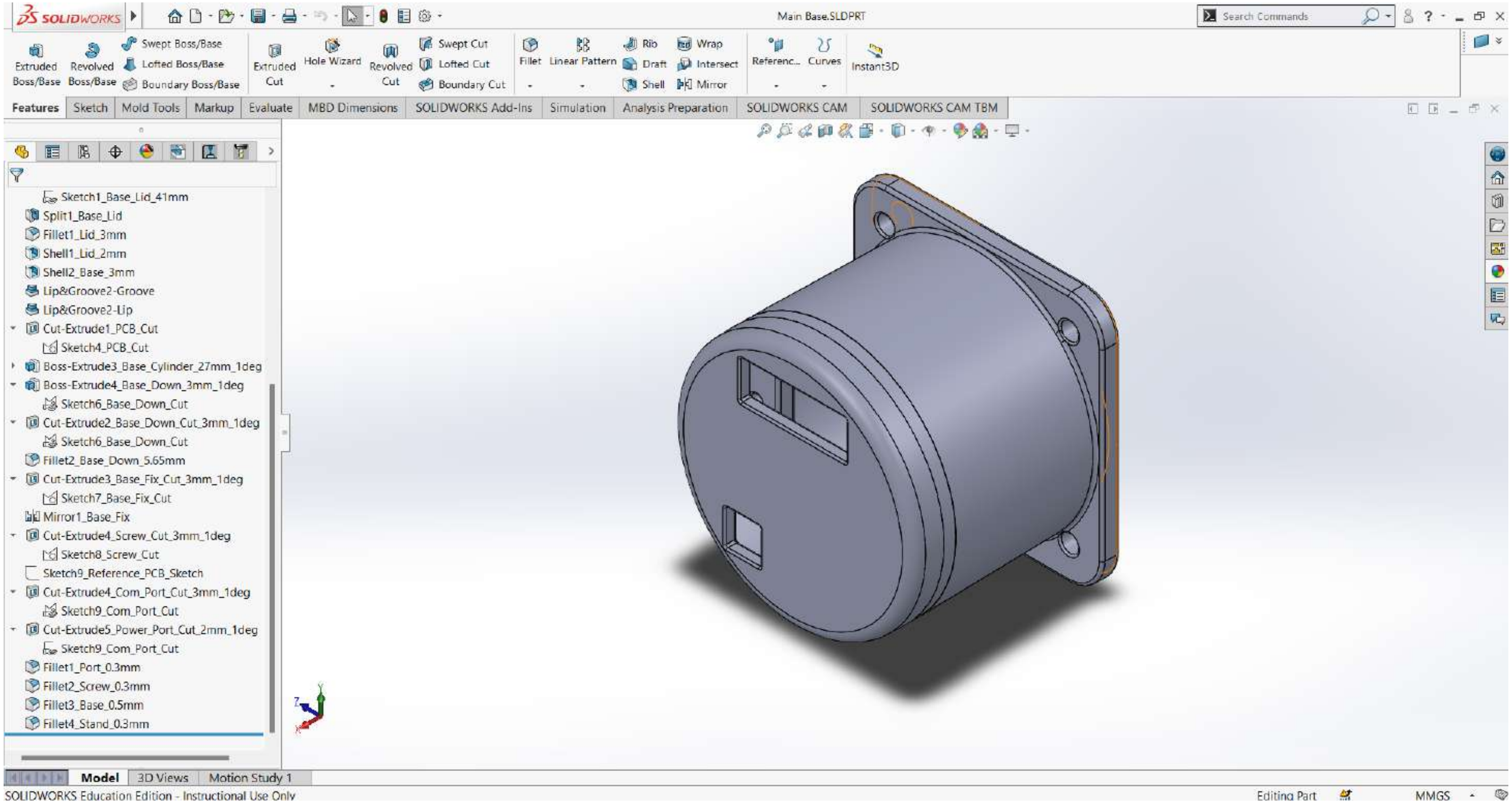
3 Enclosure Design

3.1 SolidWorks Design

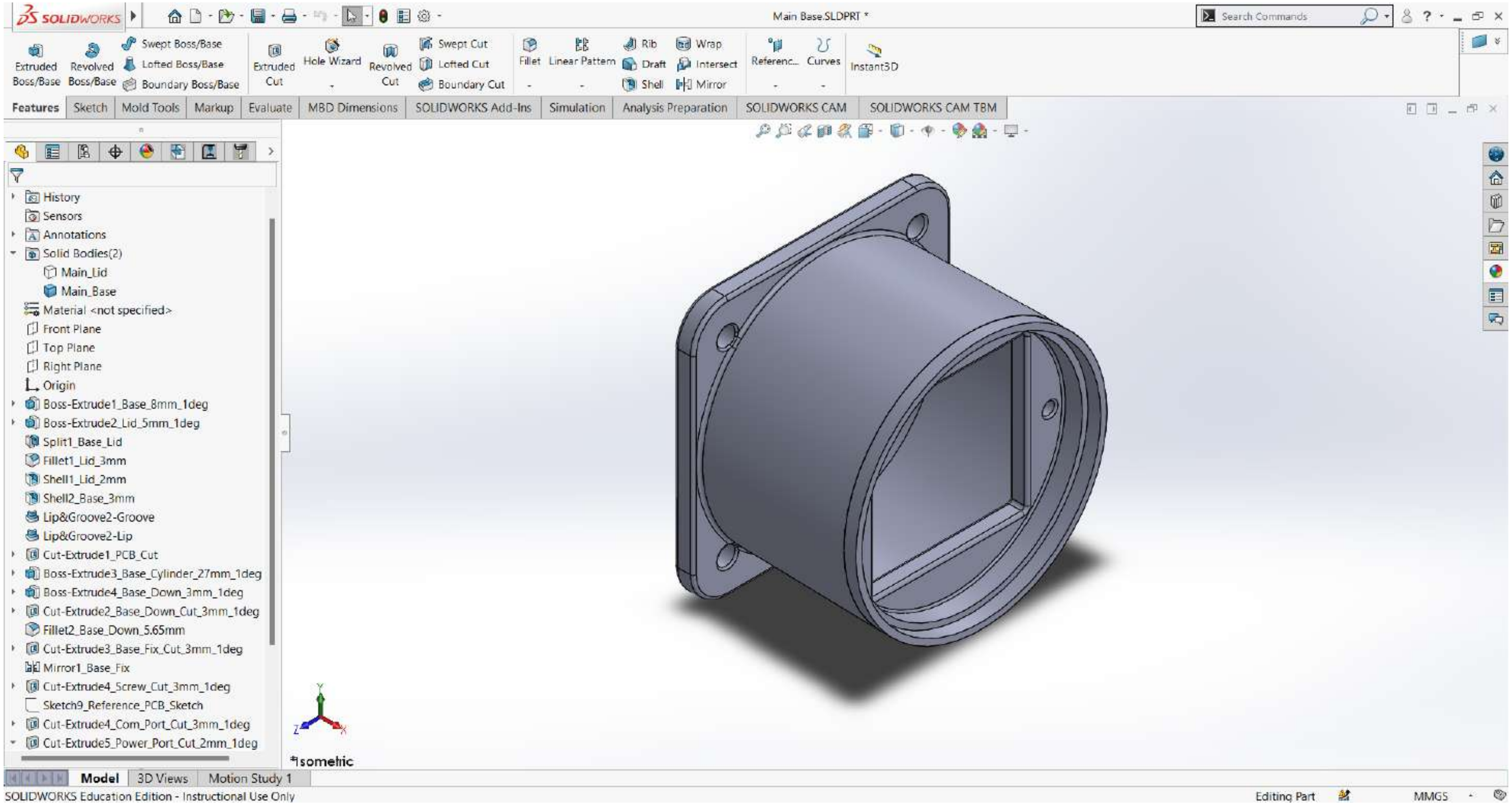
Model Tree - Part 1



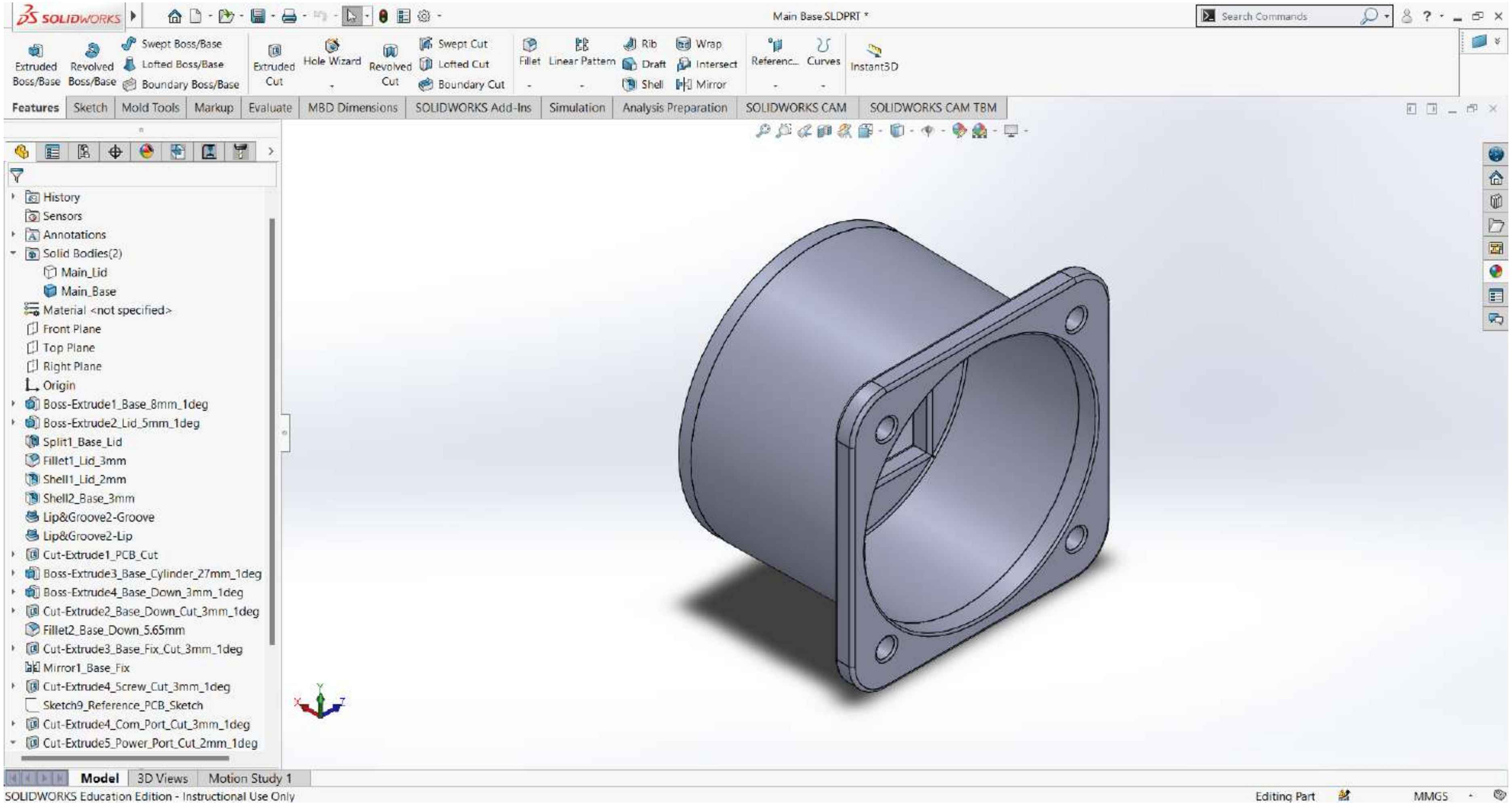
Model Tree - Part 2



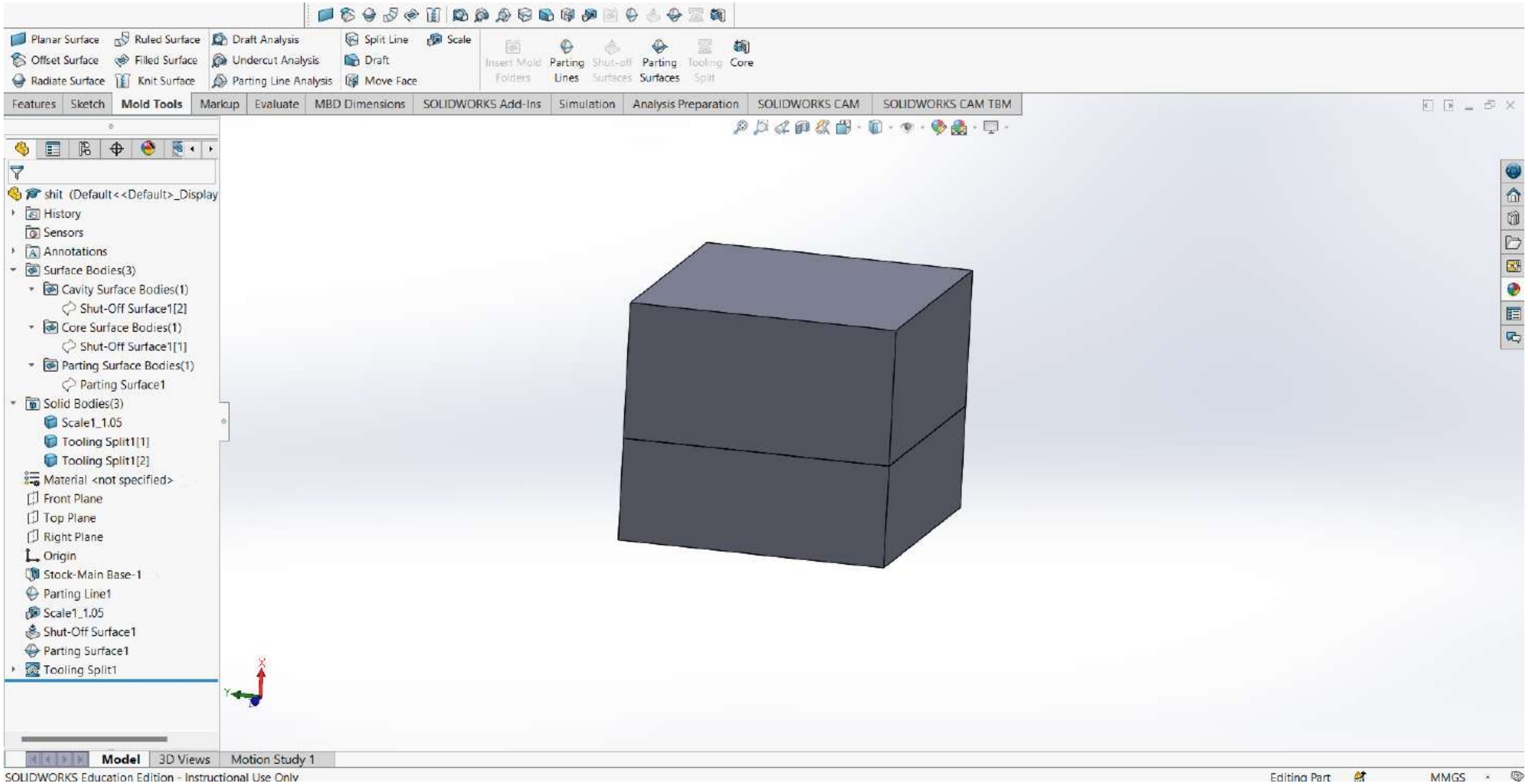
Enclosure Base



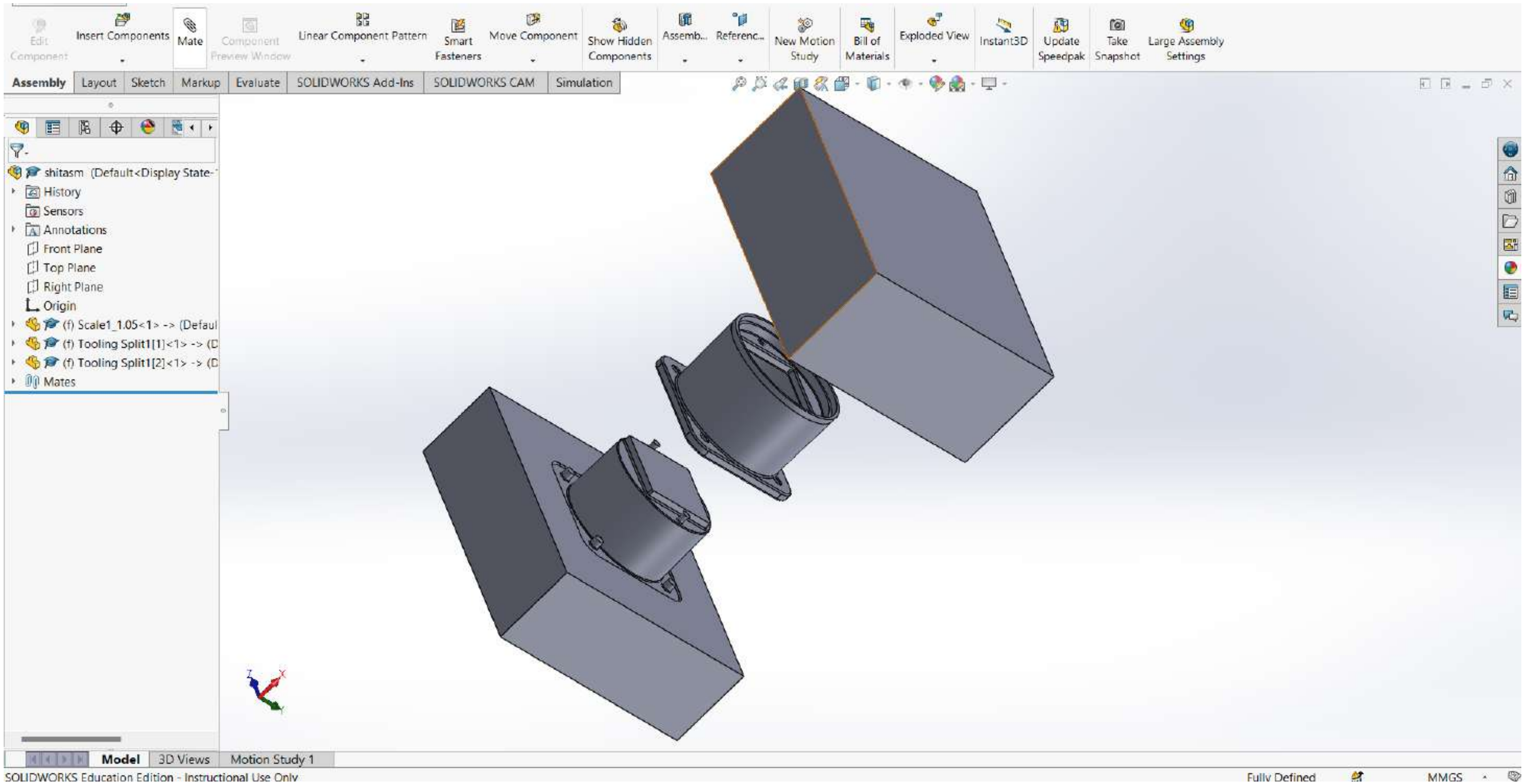
Enclosure Base

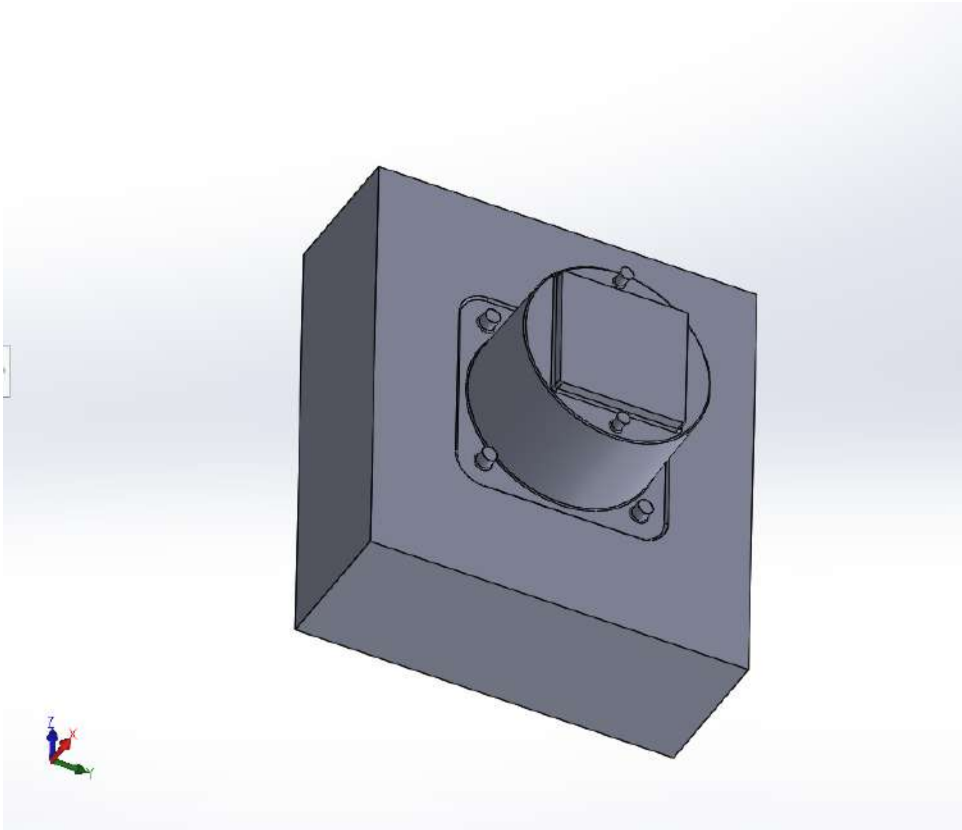


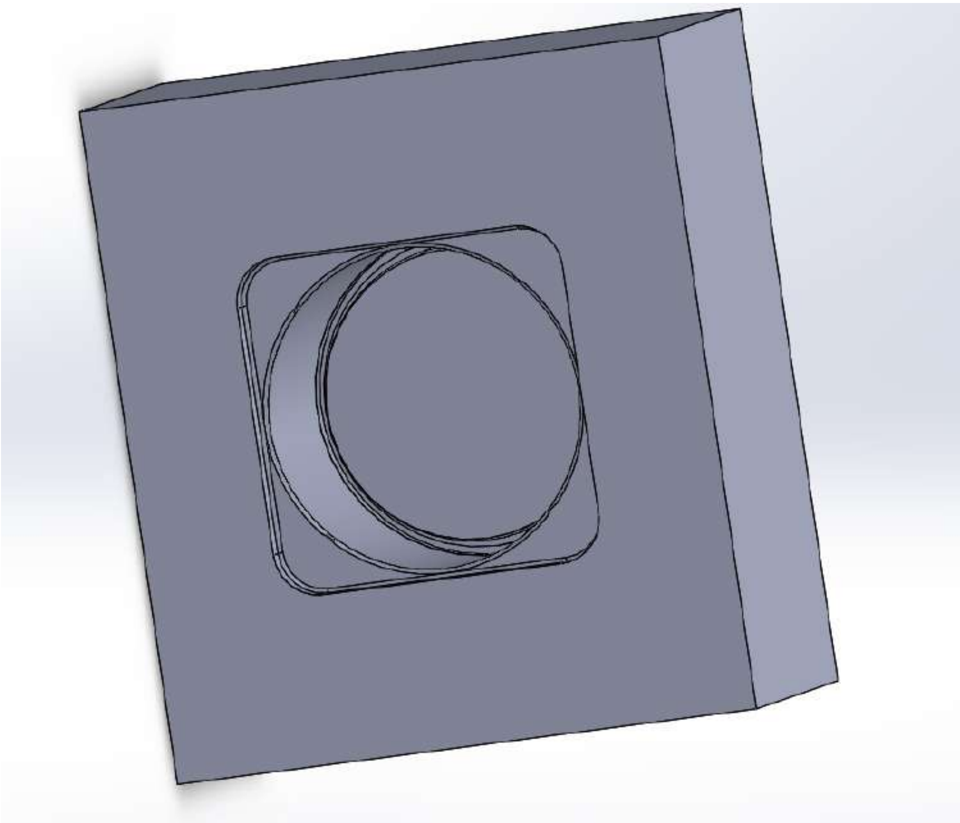
Base Part Mold Design



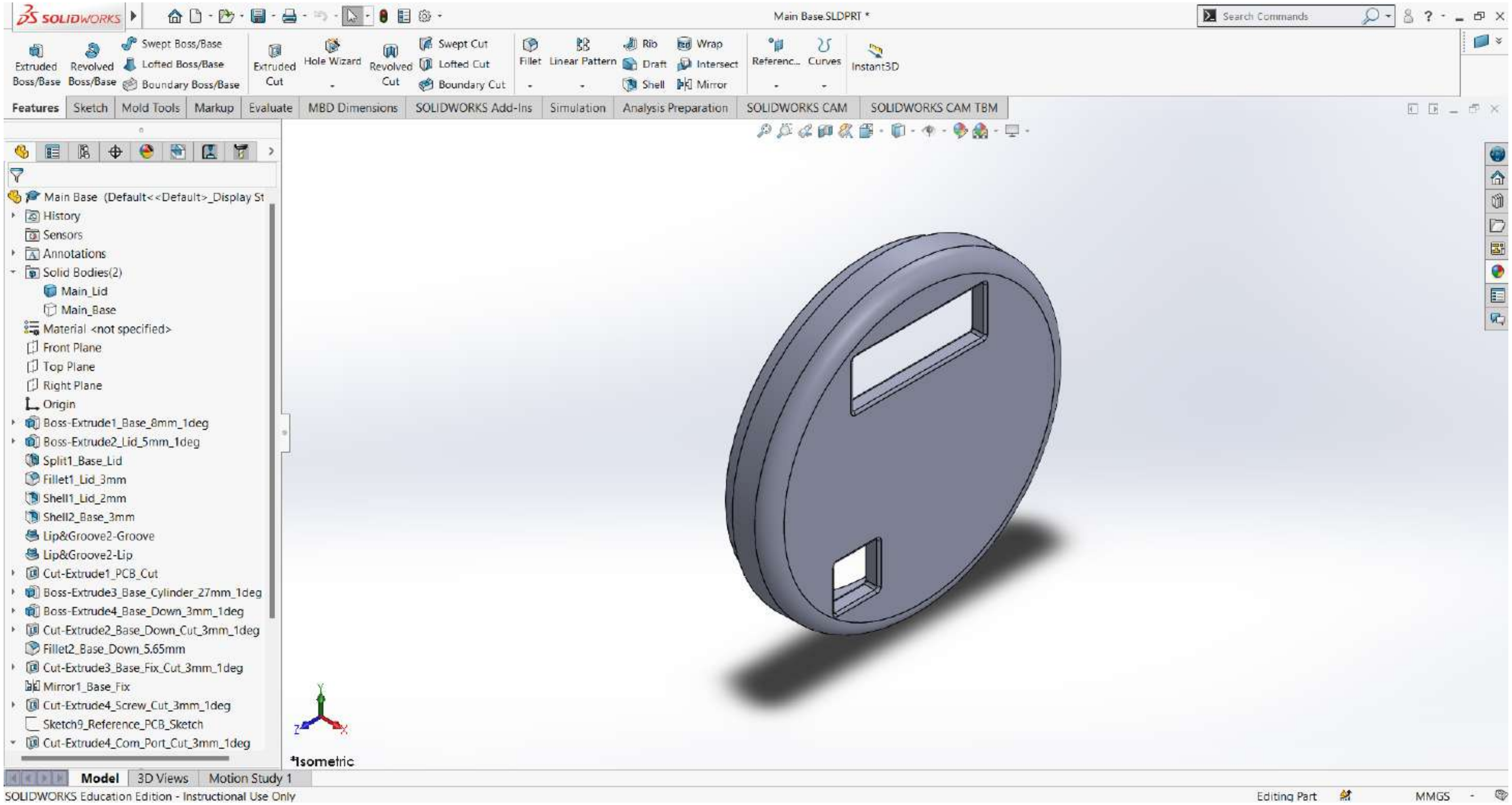
Base Part Mold Design



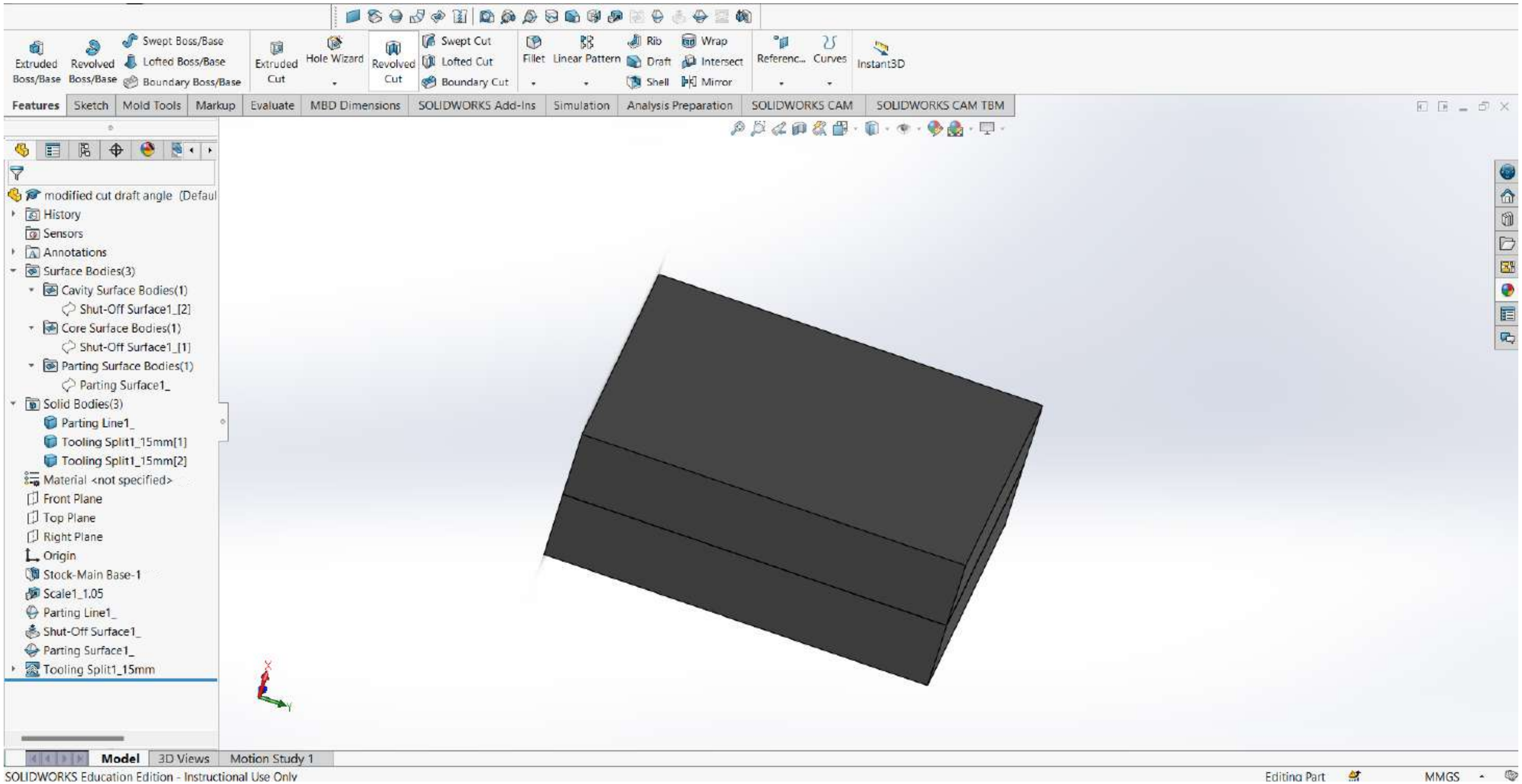




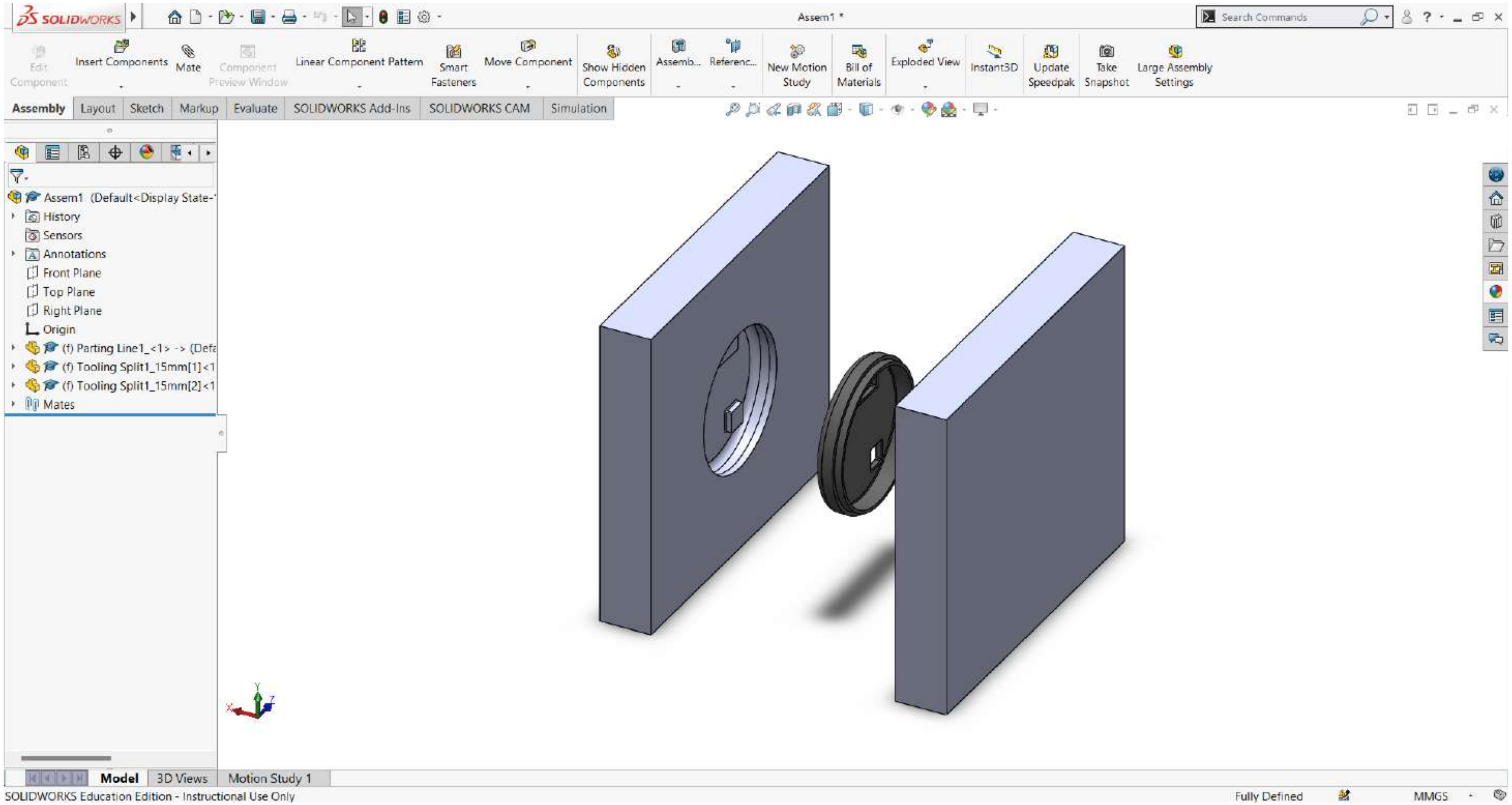
Enclosure Lid

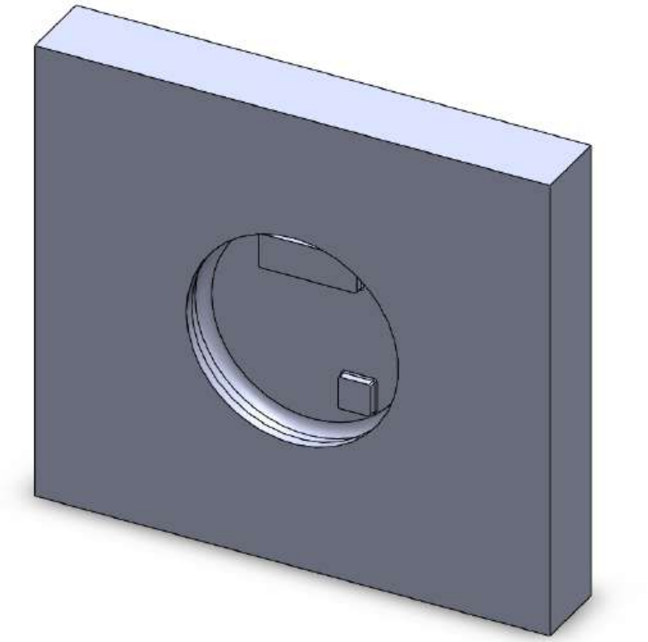


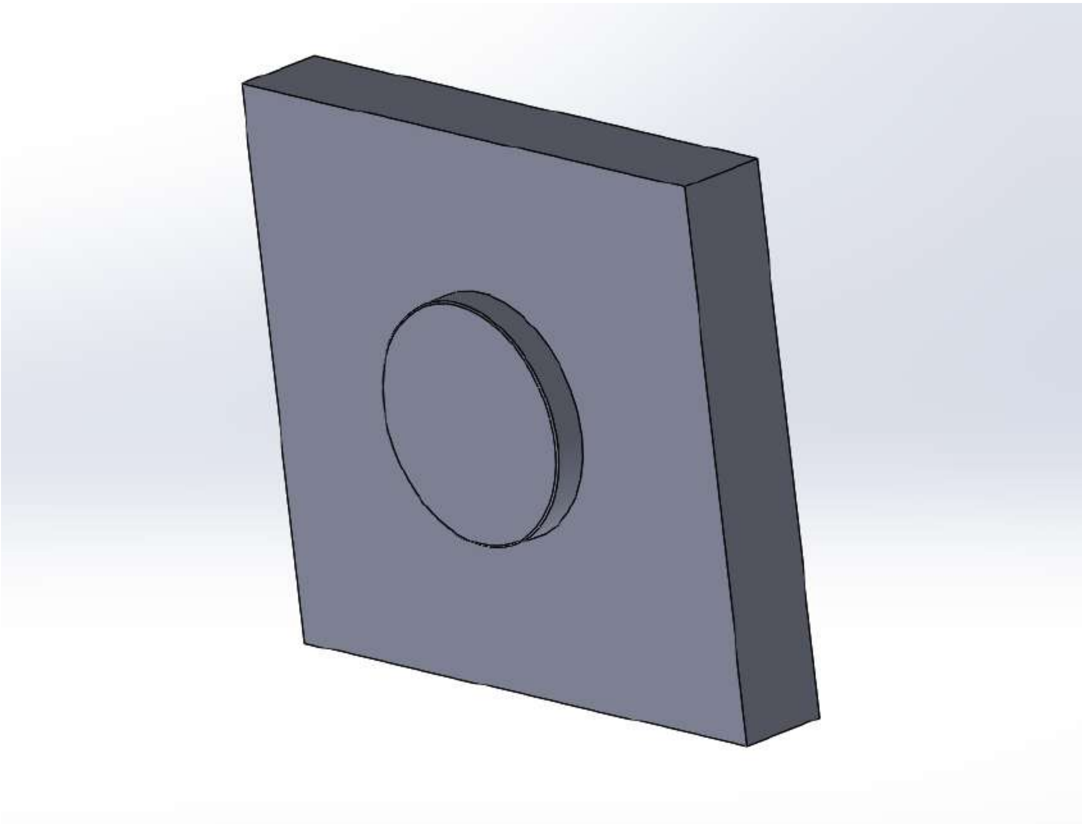
Lid Part Mold Design



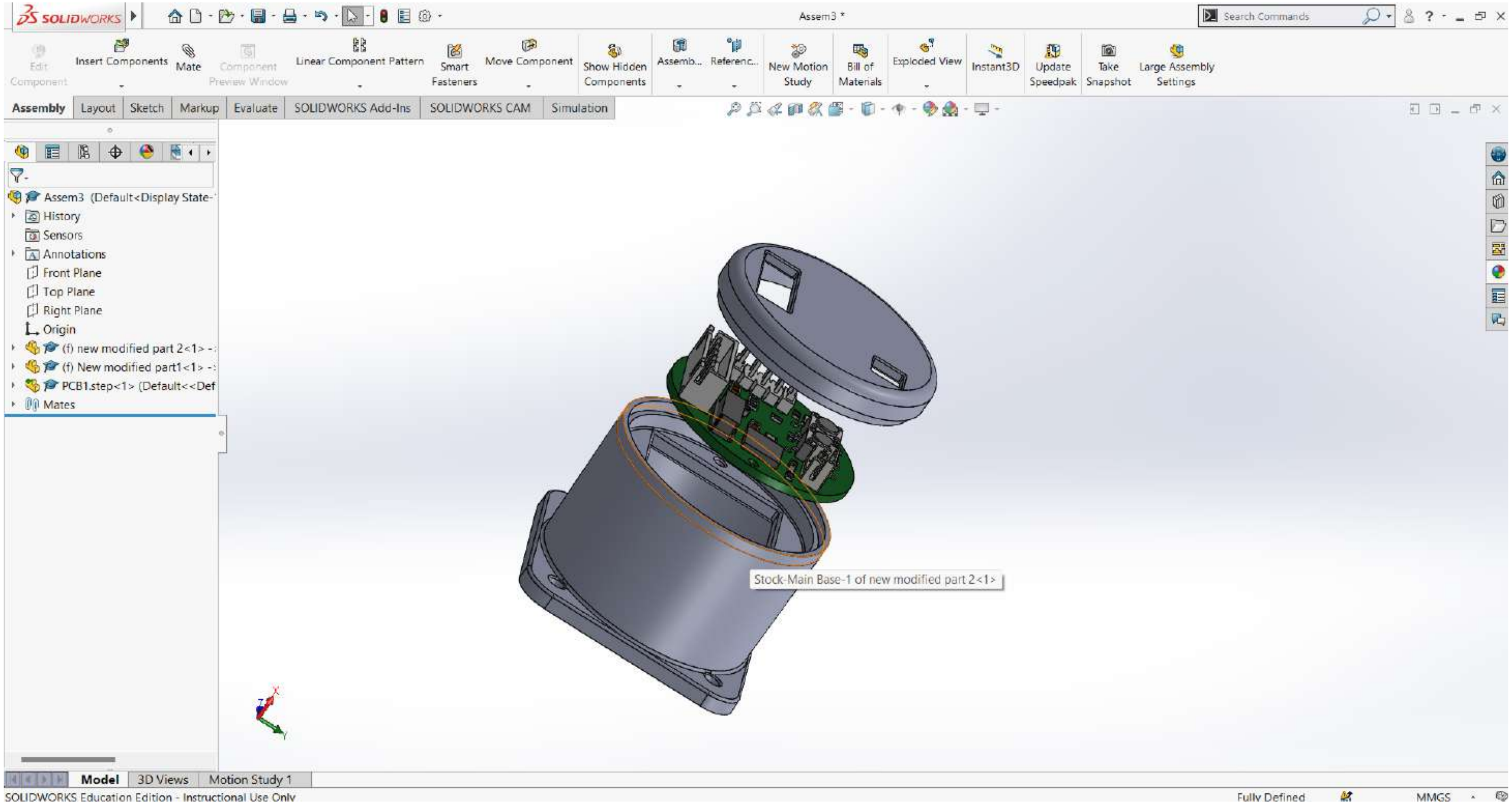
Lid Part Mold Design



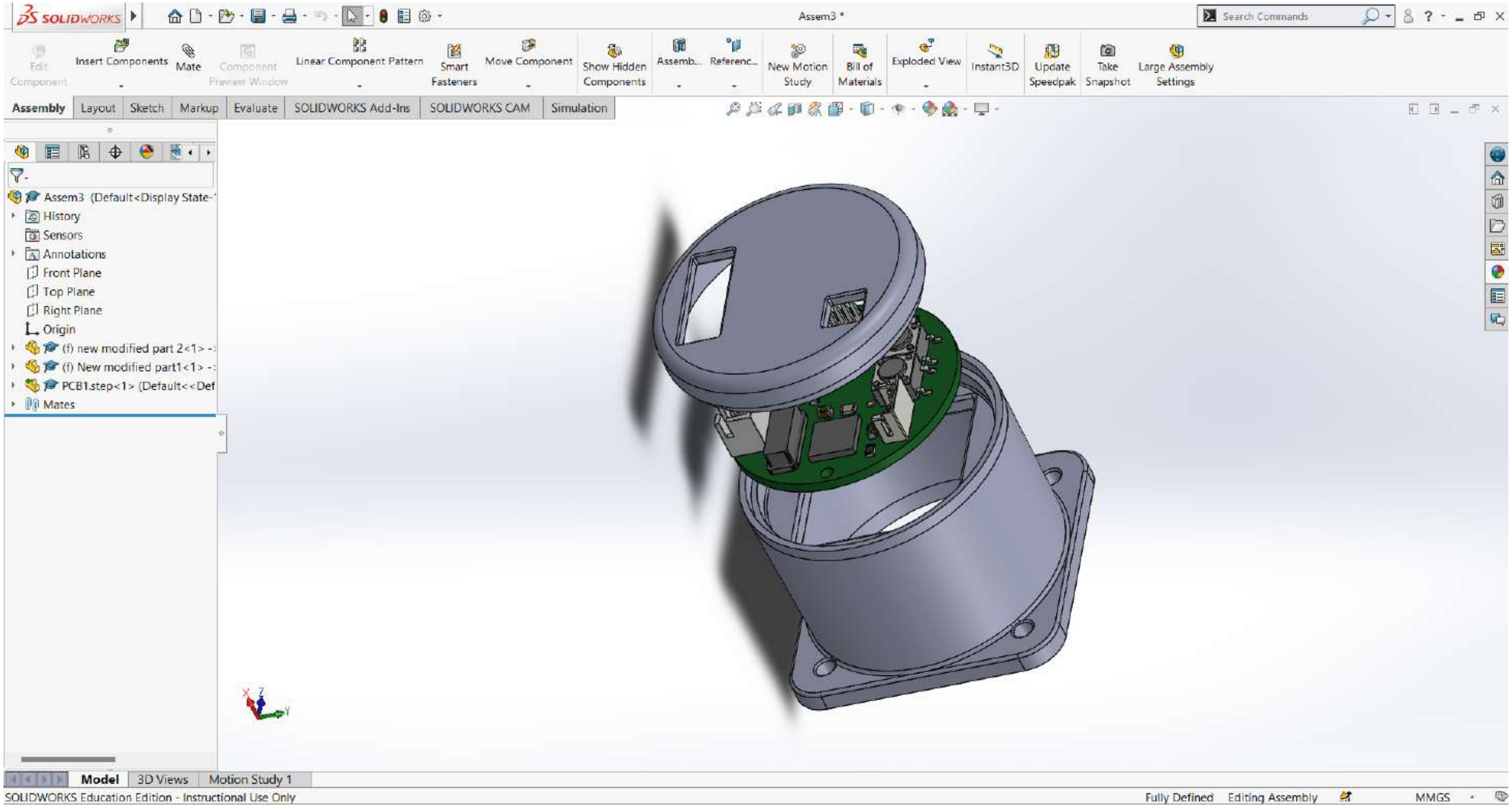


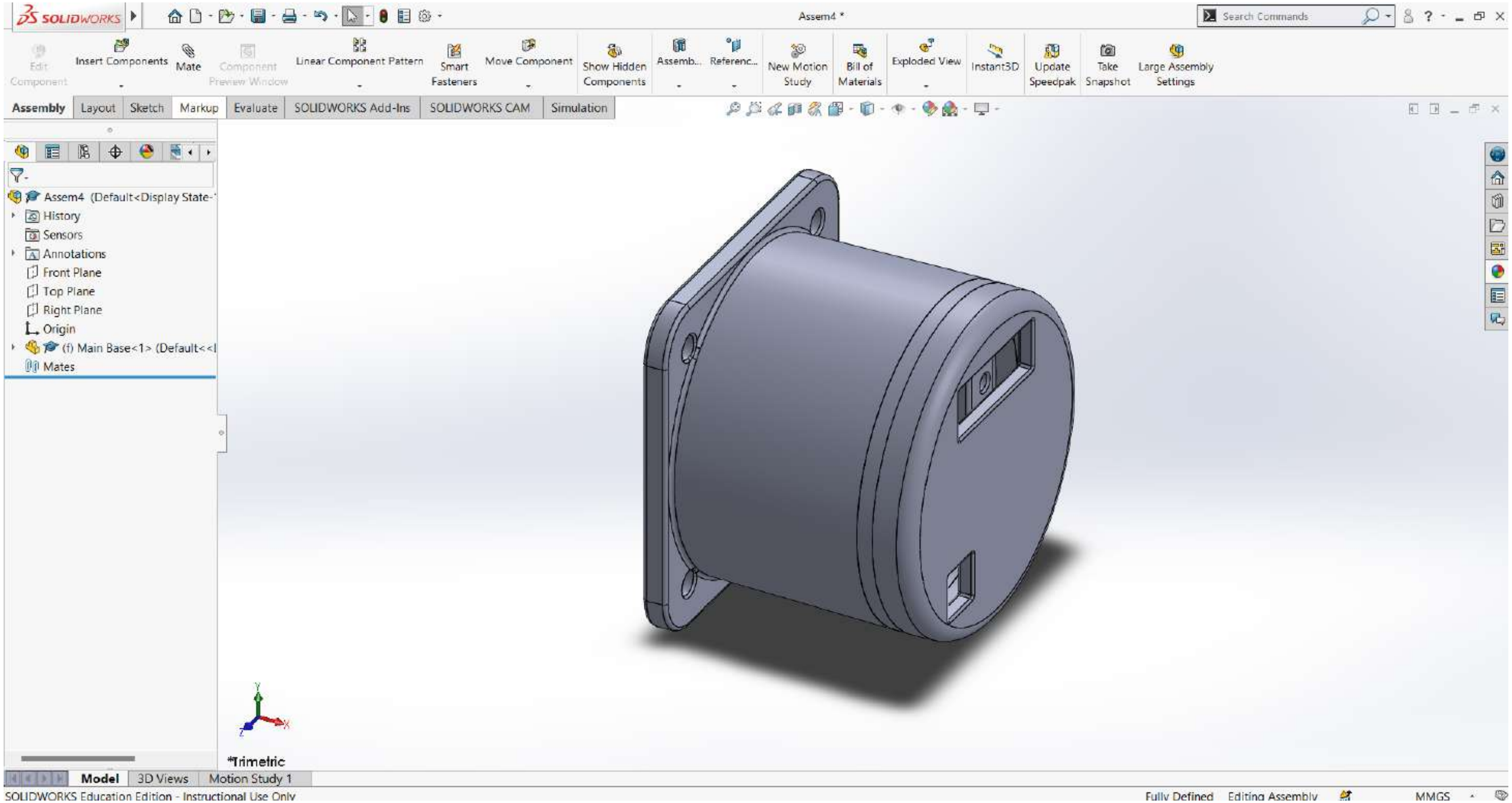


Assembly : Exploded View

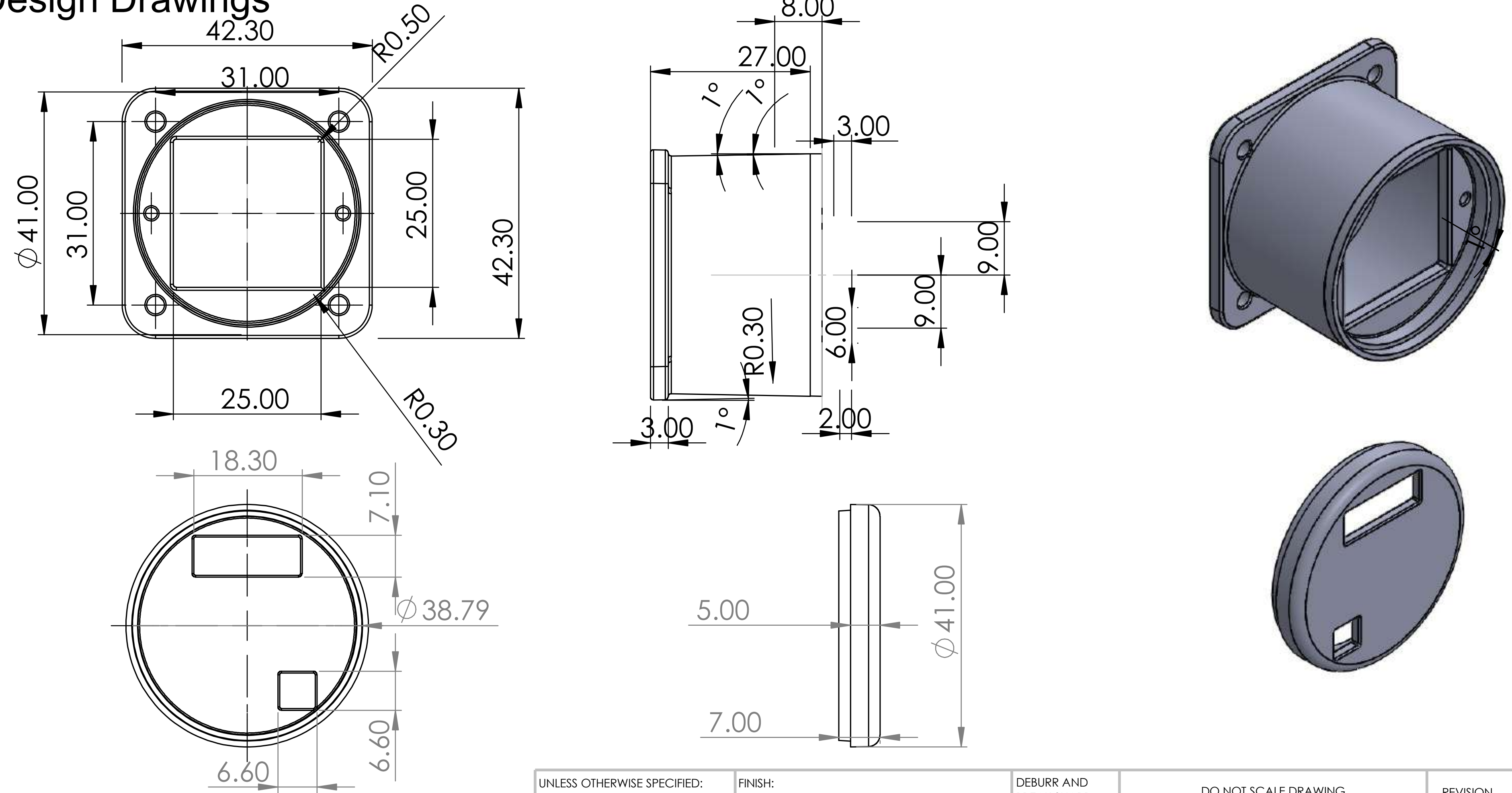


Assembly


SOLIDWORKS
Assem3 *
Search Commands
Edit Component
Insert Components
Mate
Component Preview Window
Linear Component Pattern
Smart Fasteners
Move Component
Show Hidden Components
Assemb...
Referenc...
New Motion Study
Bill of Materials
Exploded View
Instant3D
Update Speedpak
Take Snapshot
Large Assembly Settings
Assembly
Layout
Sketch
Markup
Evaluate
SOLIDWORKS Add-Ins
SOLIDWORKS CAM
Simulation
Assem3 (Default<Display State-
History
Sensors
Annotations
Front Plane
Top Plane
Right Plane
Origin
(f) new modified part 2<1> -
(f) New modified part1<1> -
PCB1.step<1> (Default<<Def
Mates
Model
3D Views
Motion Study 1
SOLIDWORKS Education Edition - Instructional Use Only
Fully Defined
Editing Assembly
MMGS


SOLIDWORKS
Assem4 *
Search Commands
Edit Component
Insert Components
Mate
Component Preview Window
Linear Component Pattern
Smart Fasteners
Move Component
Show Hidden Components
Assemb...
Referenc...
New Motion Study
Bill of Materials
Exploded View
Instant3D
Update Speedpak
Take Snapshot
Large Assembly Settings
Assembly
Layout
Sketch
Markup
Evaluate
SOLIDWORKS Add-Ins
SOLIDWORKS CAM
Simulation
Assem4 (Default<Display State-
History
Sensors
Annotations
Front Plane
Top Plane
Right Plane
Origin
(f) Main Base<1> (Default<<
Mates
*Trimetric
Model
3D Views
Motion Study 1
SOLIDWORKS Education Edition - Instructional Use Only
Fully Defined
Editing Assembly
MMGS

Design Drawings



UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:	FINISH:		DEBURR AND BREAK SHARP EDGES	DO NOT SCALE DRAWING	REVISION 01
	NAME	SIGNATURE	DATE	TITLE:	
DRAWN	Yasoda Lasiru		05/07/2024	Enclosure Drawing	
CHK'D					
APPV'D					
MFG					
Q.A				MATERIAL: -	DWG NO. 01
					A4
				WEIGHT:	SCALE:1:1 SHEET 1 OF 1

3.2 Photographs of Physically Built Enclosure

Figure 14: *Enclosure*

Figure 15: *Enclosure*

Figure 16: *Enclosure*

Figure 17: *Enclosure*

Figure 18: *Enclosure*

3.3 Photographs of System Integration

Below are steps to fix the encoder to the motor.

Figure 19: *Fix coupler with magnet to the shaft*

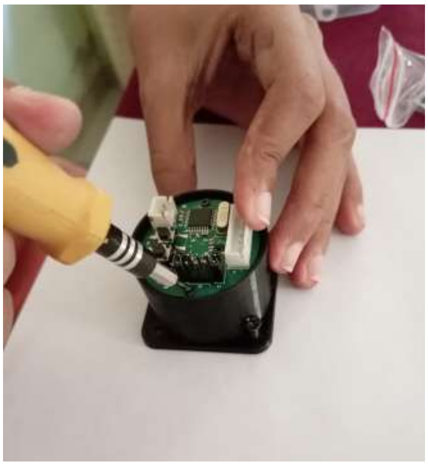


Figure 20: *Fix the PCB to the main enclosure part*

Figure 21: *Now we have to fix this to the motor*

Figure 22: *Fixing the screws*

Figure 23: *Fixing the screws*

Figure 24: *Fixing the screws*

Figure 25: *Fixing the screws*

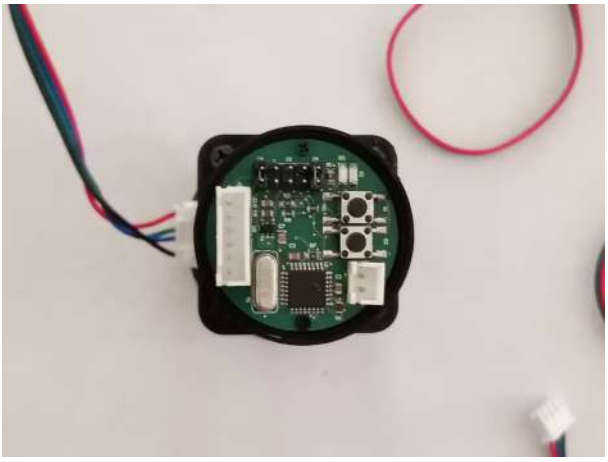


Figure 26: *After fixing all the screws*

Figure 27: *Fix the lid*

Figure 28: *Side view after fixing*

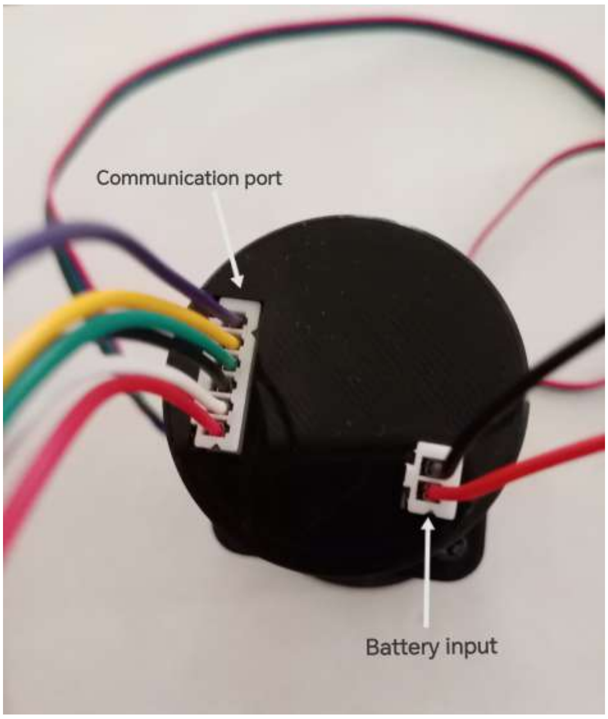


Figure 29: *Now we can fix the communication port and battery input like this*

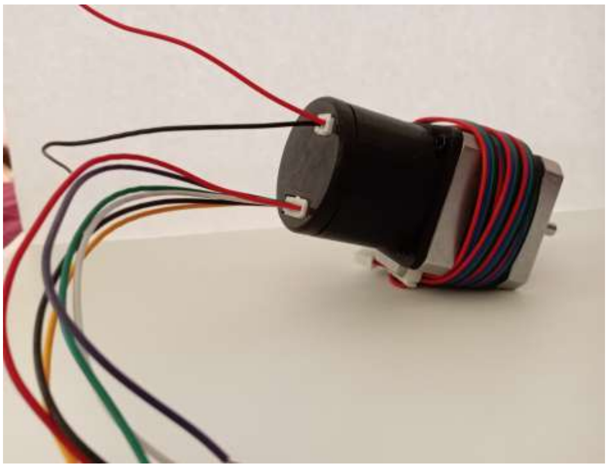


Figure 30: *Side view*

4 Software Documentation

Our encoder software should be accessible to the user outside reading the rotated angle. The main outputs should be the turned angle and the turns count. These values are output through the SPI protocol.

We have selected this protocol because of its' fast response time, ability to be implemented in the ATMega microcontroller and the ease of implementation. Since we are communicating with the hall effect IC through I2C interface, SPI is the best option to avoid any potential conflicts. This has been implemented using C in Microchip Studio. The microcontroller reads the raw angle from the hall effect sensor IC and calculate the turns count, finally it outputs the angle and the turns count with three bytes, first two bytes dedicated for the angle.

Users can access these bytes via the SPI ports in the encoder and request bytes by the index. The pin mapping to access the SPI is as below.

- User MISO → MISO
- User MOSI → MOSI
- User CS → CS
- User CLK → CLK

The following code has been used inside the ATMega328p microcontroller to facilitate the communication between the user and the encoder. Interrupts have been used to provide the output the user whenever it is requested. Here the ATMega328p acts as the SPI slave. Readings from hall effect IC is taken in the loop function. For that, the ATMega328p is acting as the I2C master, since it is recommended to read through I2C protocol from the AS5600 hall effect sensor IC.

The angle value is a 16 bit value and can be accessed by sending 0x01 and 0x02 to the encoder. 0x01 will give the high 8 bits and 0x02 will give the low 8 bits. 0x03 returns the turns count which is also a 8 bit value.

```
#define F_CPU 8000000UL

#include <avr/io.h>      //Include AVR io
#include <avr/interrupt.h> //Include std interrupt

#define I2C_FREQ 100000L
#define AS5600_ADDRESS 0x36 // Default device I2C address
#define SS     PINB2      //Define SPI pins
#define MOSI   PINB3
#define MISO   PINB4
#define SCK    PINB5
#define SCL    PINC5      //Define I2C pins
#define SDA    PINC4

/* byte 1 : angle[15:8]
byte 2 : angle[7:0]
byte 3: turns[7:0] */

uint8_t byte1 = 0;
uint8_t byte2 = 0;
uint8_t byte3 = 0;

uint8_t q=0;
uint8_t preQ = 0;

uint16_t int1;
uint16_t int2;
uint16_t angle;

void delay_func(uint8_t count) { //delay function
    volatile uint8_t delay = 0;
```

```
    while (delay < count) {
        delay++;
    }
}

void startI2C(uint32_t address) {  //start I2C
    TWCR = (1<<TWSTA)|(1<<TWEN)|(1<<TWINT); //Enable I2C and generate START
        condition
    while(!(TWCR&(1<<TWINT))); //Wait until I2C finish its current work
    TWDR = address;   //Write SLA+W in I2C data register
    TWCR = (1<<TWEN)|(1<<TWINT); // Enable I2C and clear interrupt flag
    while (!(TWCR&(1<<TWINT))); // Wait until I2C finish its current work
}

uint8_t getAngle(uint8_t reg){
    uint8_t byte;

    // Start I2C transmission to write the register address
    startI2C((AS5600_ADDRESS << 1) | 0); // AS5600 address with write bit

    TWDR = reg;  //write register address to TWDR register
    TWCR = (1<<TWEN)|(1<<TWINT); //clear interrupt flag
    while (!(TWCR&(1<<TWINT))); //wait until interrupt flag is cleared

    // Start I2C transmission to read the data
    startI2C((AS5600_ADDRESS << 1) | 1); // Address with read bit

    // Read byte
    TWCR = (1<<TWEN)|(1<<TWINT); //clear interrupt flag again to start reading
    while (!(TWCR&(1<<TWINT))); //wait until read operation is complete
    byte = TWDR;   //read the received byte in TWDR register

    // Stop I2C transmission
    TWCR = (1<<TWSTO)|(1<<TWINT)|(1<<TWEN); //generate stop condition
    while (TWCR&(1<<TWSTO)); //wait until stop condition is executed

    return byte;
}

int main(void){
    DDRB &= ~((1<<MOSI)|(1<<SCK)|(1<<SS)); // Make MOSI, SCK, SS as input
    DDRB |= (1<<MISO);          // Make MISO pin as output
    SPCR |= (1<<SPE) ;          // Enable SPI in slave mode
    SPCR |= (1<<SPIE);          // Enable SPI interrupts
    SPDR = 0;                   //clear SPI data first

    sei(); //global interrupt enable

    DDRC |= (1<<SDA)|(1<<SCL); //set SCL and SDA as output
    PORTC |= (1<<SDA)|(1<<SCL); //set SCL and SDA high
```

```
    TWSR = 0x00; // Set pre scaler to 1
    TWBR = ((F_CPU / I2C_FREQ) - 16) / 2; //initialize i2c frequency

    ADCSRA = 0;    //disable ADC
    PRR = (1<<PRTIM1)|(1<<PRTIM0)|(1<<PRTIM2);  //disable timers

    while (1) {
        byte1 = getAngle(0x0D); //get high byte
        byte2 = getAngle(0x0C); //get low byte

        int1 = (int) byte1;
        int2 = (int) byte2;

        uint16_t raw_angle = (int1<<8)|int2; //raw angle calculation - between
            0 - 4096 (12 bit value)
        angle = raw_angle*360/4096; //to get angle in 0-360 range

        //update byte3
        if (0<=angle && angle<=90) q = 1;
        else if (90<angle && angle<=180) q = 2;
        else if (180<angle && angle<=270) q = 3;
        else if (270<angle && angle<360) q = 4;
        if (q!=preQ) {
            if (q==1 && preQ==4) byte3++; //increase turns count
            if (q==4 && preQ==1) byte3--; //decrease turns count
            preQ = q;
        }

        delay_func(100);
    }
    return 0;
}

ISR (SPI_STC_vect){
    uint8_t c = SPDR; //get the requested register name
    switch (c) {
        case 0x01: //asking for byte1 (high byte)
        SPDR = byte1;
        break;

        case 0x02: //asking for byte2 (low byte)
        SPDR = byte2;
        break;

        case 0x03: //asking for byte3 (turns count)
        SPDR = byte3;
        break;
    }
}
```

References

[1] AMS AS5600 position sensor position sensors Datasheet - AMS-Osram ams. Available at: https://ams-osram.com/products/sensors/position-sensors/ams-as5600-position-sensor (Accessed: 15 May 2024).

[2] ATMEGA328P Datasheet. Available at: https://www.microchip.com/en-us/product/atmega328p

[3] DW01A Datasheet. Available at: https://cxem.net/master/files/97_DW01A-DS-11_EN.pdf

[4] FS8205A Datasheet. Available at: https://www.ic-fortune.com/upload/Download/FS8205A-DS-12_EN.pdf

[5] TP4056 Datasheet. Available at: https://dlnmh9ip6v2uc.cloudfront.net/datasheets/Prototyping/TP4056.pdf

[6] MIC5205 Datasheet. Available at: https://ww1.microchip.com/downloads/en/DeviceDoc/20005785A.pdf

Appendix

A Daily Log

- **28 Feb 2024**

  Final Project Proposal submitted

- **23 Feb - 26 Mar 2024**

  Review progress
  A comprehensive study was conducted on the existing work in the industry regarding multi-turn absolute magnetic encoders. The study revealed that there are primarily two types of encoders available in the market: optical and magnetic encoders. Among these, magnetic encoders were found to be more prevalent due to their robustness, durability, and immunity to external factors such as dust and moisture.

  Furthurmore, multi-turn magnetic encoders come in various configurations, including different resolutions, shaft sizes, and enclosure types. Some encoders only consisted of a PCB, requiring a specific mechanism to attach them to the motor. On the other hand, most encoders had enclosures that included a user interface (UI) to display additional information such as position, speed, and direction.

  The study also highlighted the importance of enclosure design in ensuring the reliability and durability of the encoder. Some enclosures were designed to provide IP67 or IP68 protection against dust and water ingress, while others were designed to withstand harsh operating conditions such as high temperatures, vibrations and shocks.

  Based on the findings , a detailed analysis was performed to identify the key design considerations for our multi-turn absolute magnetic encoder. The analysis included a comparison of different enclosure designs, shaft sizes, and UI options as well as an evaluation of their impact on the overall performance and reliability of the encoder. The results of this analysis will be presented in the design documentation.

- **27 Feb - 2 Mar 2024**

  Studied about how the EDR project can be extended for the development of the Sri Lankan Industry
  It was observed that solid theoretical foundation is essential before attempting any project and working according to scientific principles is crucial for project success. Also, selecting suitable projects that align with the capabilities and resources of the Sri Lankan industry is vital for contributing to its development. To prevent future mistakes, we should focus more on understanding market and industry needs, as well as the unique challenges of the Sri Lankan industry.

- **4 Mar - 10 Mar 2024**

  Planned the Next Steps

Explored various circuits and methodologies to be used for our own design. These included three main types.

- Using a gearbox

  There are several encoders in the market using gearboxes. The advantages of using a gearbox is that it doesn't need a separate power source to function. The disadvantage is bulkiness it has and the complexness associated with making the gears. Apart from that it's a good option.

Figure 31: *Gear Box in an Encoder*

- Using wiegand sensor This sensor is not currently available to purchase.

Figure 32: *Wiegand Sensor in an Encoder*

- Using battery backup

  This seems to be the most practical option. But, this approach needs to minimize the power consumption a lot to have a significant efficiency compared to others.

- **10 Mar - 22 Mar 2024**

  Create Stakeholder Map and Observe Users

  Identified the potential stakeholders related to our project. Studied about the various situations where multi turn absolute magnetic encoders are currently used. These include robotic arms, rotating shaft of motors etc.

- **23 Mar - 28 Mar 2024**

Figure 33: *Battery Backup Encoder*

Enclosure Design Discussion
It was decided to have three main parts in our enclosure. They are separately, base part to fix to the motor, holder part for the PCB and battery and main housing part. These parts were discussed with details and rough designs were made.

- **29 Mar 2024**

  Battery Selection
  Decided the battery specifications and calculations associated with it, to run the device for a longer time. Calculations are included in documentation.

- **23 Mar - 1 Apr 2024**

  Designing the circuit
  After careful consideration, it was decided to implement a battery backup circuit for our multi-turn absolute magnetic encoder. The design criteria for the circuit were established and documented, taking into account the specific requirements of the AS5600 hall effect sensor IC, ATMEGA328p-AU micro controller and other main power consuming components.

  To ensure accurate and reliable data transmission, it was decided to use the I2C protocol to read data from the AS5600 hall effect sensor IC and output it to the user through SPI protocol. This decision was based on the high precision and accuracy of the I2C protocol, as well as the analog output of the AS5600. By reading the SPDR data register using I2C, we can obtain the exact angle turned with 12-bit precision, making it a more convenient and reliable solution.

  In addition to finalizing the circuit design, we also discussed the importance of conducting a power consumption test to verify that our design meets the low power consumption requirements. This test will involve measuring the power consumption of the circuit in various operating modes and comparing it to the design specifications. Based on the results of this test, we may need to make further optimizations to reduce power consumption and extend battery life.

- **1 Apr - 8 Apr 2024**

  Designed the schematics and finalized the PCB.

- **13 Apr 2024**

  Sent the PCB design files to manufacture

- **22 Apr 2024**

Enclosure Design Discussion
Initially it was decided to go with three parts for the enclosure. It was decided to make a few changes to that and combine two of those for the ease of assembly and manufacturing. Now base part and PCB holding part will be combined to a single one and separate part will be designed as the lid.

- **22 Apr - 29 Apr 2024**

  Designed the enclosure with new changes.

- **30 Apr - 3 May 2024**

  Soldered the PCB

- **18 Mar - 4 May 2024**

  Coding

  It was experimented with several algorithms to implement the communication between the user, ATMega IC and the hall effect sensor IC. First it was tried implementing the protocols without using interrupts, but with furthure experiment, interrupts turned out to be the appealing solution to manage both the protocols at once.

  Here it was made Raspberry pi as the SPI master and ATMEGA328p-AU as the SPI slave. Also, ATMEGA328p would act as the I2C master for AS5600 hall effect sensor IC. It would read data from AS5600 through I2C and count the turns and then send the turned angle along with the number of turns to the raspberry pi through SPI. Raspberry pi was solely used for the purpose of interfacing and testing of our multi turn absolute magnetic encoder.

  It was managed to successfully read the data through user interface even though initially we faced some difficulties in reading them. The SPI protocol seemed to not work properly when both the I2C and SPI protocols were being used inside the ATMEGA328p. With SPI running using vector interrupts, we were able to get both of the protocols running on the micro controller at the same time.

  A sample code to test the encoder is been given below. This has been tested in a Raspberry Pi Model 4B.

  test.c

```c
#include <stdio.h>
#include <stdint.h>
#include <fcntl.h>
#include <sys/ioctl.h>
#include <linux/spi/spidev.h>
#include <unistd.h>
#include <sys/mman.h>

#define SPI_DEVICE "/dev/spidev0.0" // Define SPI device path
#define GPIO_BASE 0x20200000        // Define base address for GPIO pins
#define GPIO_LEN 0x1000             // Define length of GPIO memory
```

```
volatile uint32_t *gpio;  // Declare a pointer to a volatile uint32_t
    for GPIO access

void delay_func(uint32_t count)
{
  volatile uint32_t numDelay = 0;
  while (numDelay < count)
  {
    numDelay++;
  }
}

void setup_gpio() {  //setup gpio
    int fd = open("/dev/mem", O_RDWR|O_SYNC);  //open mem location for
        read write
    if (fd<0) {
        perror("Failed to open /dev/mem");
        return;
    }
    gpio = (volatile uint32_t *)mmap(NULL, GPIO_LEN,
        PROT_READ|PROT_WRITE, MAP_SHARED, fd, GPIO_BASE);
    if (gpio == MAP_FAILED) {
        printf("Failed to map GPIO memory");
        close(fd);
        return;
    }
    close(fd);
}

void gpio_set_output(int pin) {
    volatile uint32_t *reg = &gpio[pin/10];
    int shift = (pin%10)*3;
    *reg = (*reg & ~(7<<shift))|(1<<shift);
}

void gpio_write(int pin, int value) {
    if (value)
            gpio[7] = 1<<pin;// address of the GPIO register for the
                given pin
            // shift value
    else      //write to gpio
        gpio[10] = 1<<pin;
}

unsigned char spi_transfer(int fd, unsigned char tx) {
    unsigned char rx;
    struct spi_ioc_transfer tr = {
        .tx_buf = (unsigned long)&tx,   //set transmit and receive buffers
        .rx_buf = (unsigned long)&tx,
        .len = 1,
    };
```

```
    ioctl(fd, SPI_IOC_MESSAGE(1), &tr); //SPI transfer
    return tx;                          //return returned value
}

int main() {
    setup_gpio();                   //set up
    gpio_set_output(0);
    int fd = open(SPI_DEVICE, O_RDWR);  //open for read write
    if (fd < 0) {
        printf("Failed! No SPI device.");
        return 1;
    }

    uint8_t mode = SPI_MODE_0;      //SPI mode
    uint8_t bits = 8;               //num bits
    uint32_t speed = 2000000;       //2MHz speed
    if(ioctl(fd, SPI_IOC_WR_MODE, &mode)<0||ioctl(fd,
        SPI_IOC_WR_BITS_PER_WORD, &bits)<0||ioctl(fd,
        SPI_IOC_WR_MAX_SPEED_HZ, &speed)<0){ // Write the SPI speed
        printf("Failed!");
        close(fd);
        return 1;
    }
    while (1) {

        gpio_write(0, 0);     // enable Slave Select
        unsigned char a = spi_transfer(fd, 0x01);  //read first byte from
            reguster
        unsigned char b = spi_transfer(fd, 0x02);  //read second byte
            from reguster
        unsigned char c = spi_transfer(fd, 0x03);  //read third byte from
            reguster

        gpio_write(0, 1);     // disable Slave Select
        delay_func(1000); //  delay
        gpio_write(0, 0);                     // enable Slave Select

        int numberOfTurns = (int) c;            // convert to integer
        int finAngle = ((int)a<<8)|(int) b;   // construct the 2 byte
            integer

        gpio_write(0, 1);     // disable Slave Select

        printf("Number of Turns: %d\n", numberOfTurns); //print found
            values
        printf("Angle from 0 to 360: %d\n", finAngle);
        delay_func(1000); // delay
    }
    close(fd);
    return 0;
}
```

- **20 May - 24 May 2024**

  Made a dashboard
  Made a dashboard to display the readings from the encoder (turned angle and the number of turns it turned). It was made using Node-RED. The turns count and angle value are published using Mqtt server in it.

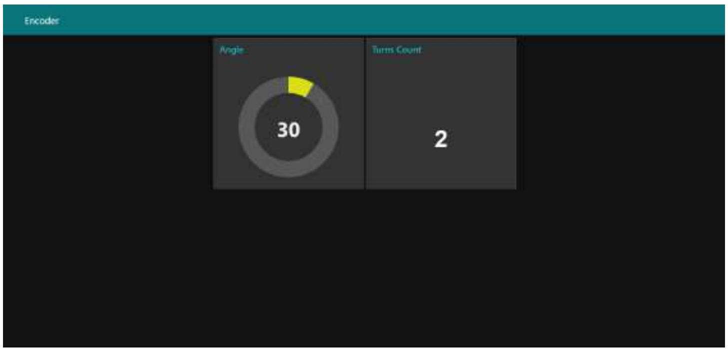


Figure 34: *Node-RED Dashboard*

Previously Used Code With Libraries

```
#include <avr/io.h>
#include <avr/interrupt.h>
#include <util/delay.h>
#include <Wire.h>
// Default I2C address of AS5600 Hall effect IC
#define AS5600_ADDRESS 0x36
volatile char highByt;
volatile char lowByt;
float numberOfTurns = 0; // number of turns
int fin;
int quadrantNumber = 0; // quadrant IDs
int previousQuadrantNumber = 0;
// commanda to say what to do with incoming data
volatile byte command = 0;
void setup() {
    // have to send on master in, *slave out*
    DDRB |= (1 << DDB4); // Set MISO as OUTPUT
    // turn on SPI in slave mode
    SPCR |= (1 << SPE);
    // turn on interrupts
    SPCR |= (1 << SPIE);
    Wire.begin();
    }
void getTurns(){
/*
//Quadrants:
4 | 1
--|---
3 | 2
*/
// Quadrant 1
```

```
    if (fin >= 0 && fin <= 90) {
    quadrantNumber = 1;
    }
    // Quadrant 2
    if (fin > 90 && fin <= 180) {
    quadrantNumber = 2;
    }
    // Quadrant 3
    if (fin > 180 && fin <= 270) {
    quadrantNumber = 3;
    }
        // Quadrant 4
    if (fin > 270 && fin < 360) {
    quadrantNumber = 4;
    }
    if (quadrantNumber != previousQuadrantNumber){ // if we changed quadrant
        if (quadrantNumber == 1 && previousQuadrantNumber == 4) {
            numberOfTurns++; // 4 --> 1 transition: CW rotation
        }
        if (quadrantNumber == 4 && previousQuadrantNumber == 1) {
            numberOfTurns--; // 1 --> 4 transition: CCW rotation
        }
        previousQuadrantNumber = quadrantNumber; //update to the current
            quadrant
    }
    totalAngle = (numberOfTurns * 360) + fin; //number of turns (+/-) plus the
        actual angle within the 0-360 range
    }
}
// SPI interrupt routine
ISR(SPI_STC_vect) {
    byte c = SPDR;
    switch (command) {
        // no command? then this is the command
        case 0:
            command = c;
            SPDR = 0;
            break;
    // first byte
        case   1   :
            SPDR = highByt;
            break;
    // second byte
        case   2   :
            SPDR = lowByt;
            break;
    // third byte - turns count
        case   3   :
            SPDR = lowByt;
            break;
    } // end of switch
```

```
} // end of interrupt service routine (ISR) SPI_STC_vect
void loop() {
        // if SPI not active, clear current command
    if (PINB & (1 << PINB2)) { // Read SS pin
        command = 0;
    }
    _delay_ms(100);
    Wire.beginTransmission(AS5600_ADDRESS);
    Wire.write(0x0D); // Send the address of the raw angle register
    Wire.endTransmission();
    Wire.requestFrom(AS5600_ADDRESS, 1); // Request 1 byte of data
    _delay_ms(100);
    while (Wire.available()) {
        lowByt = Wire.read(); // Read the low byte
    }
    _delay_ms(100);
    Wire.beginTransmission(AS5600_ADDRESS);
    Wire.write(0x0C); // Send the address of the raw angle register
    Wire.endTransmission();
    Wire.requestFrom(AS5600_ADDRESS, 1); // Request 1 byte of data
    _delay_ms(100);
    while (Wire.available()) {
        highByt = Wire.read(); // Read the high byte
    }
    _delay_ms(100);
}
int main() {
    setup();
    // Enable global interrupts
    sei();
    while (1) {
        loop();
    }
    return 0;
}
```

This documentation has been reviewed by Group H - Obstacle Avoidance System for Warehouse AMR and Obstacle Avoidance System for Warehouse AGV.

Name	Signature
1. Kuruppu M. P.	
2. Peiris D. L. C. J.	
3. Herath B. H. M. K. S. B.	
4. Madushan I. D.	