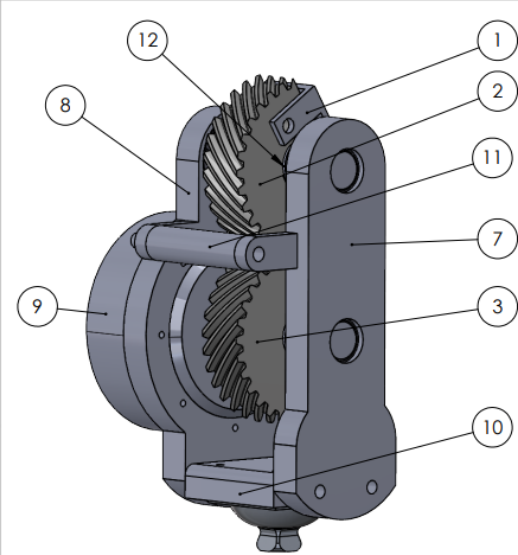

MECHANICAL HIP PROSTHETIC HORIZON HIP

AIDEN CAMISA, VICTORIA LYON,
MATT MARTINEZ, QUINN O'NEILL



Design Efforts – CAD



ITEM NO.	PART NUMBER	QTY.
1	Attachment Bracket	1
2	1030R Cut	1
3	1030R	1
4	LowerShaft	1
5	UpperShaft	1
6	7203B Bearing	3
7	Bearing Bracket	1
8	Motor Bracket	1
9	CubeMars ak80-64	1
10	LowerBar	1
11	StructuralBar	1
12	Retaining ring dsr-19	4
13	ShaftKeyV3	2

		UNLESS OTHERWISE SPECIFIED:	NAME	DATE	
		DIMENSIONS ARE IN INCHES	DRAWN		
		TOLERANCES:	CHECKED		TITLE:
		FRACTIONAL: ±	ENG APPR.		
		ANGULAR: MACH ± BEND ±	MFG APPR.		
		TWO PLACE DECIMAL ±	Q.A.		
		THREE PLACE DECIMAL ±	COMMENTS:		
		INTERPRET GEOMETRIC TOLERANCING PER:			
		MATERIAL:			
		FINISH:			
NEXT ASSY	USED ON				SIZE DWG. NO. REV
					A 100build
APPLICATION		DO NOT SCALE DRAWING			SCALE: 1:5 WEIGHT: SHEET 1 OF 1

PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <INSERT COMPANY NAME HERE>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> IS PROHIBITED.


```

Send to Rpi : 1 7F FE 80 27 FD 47 0 7D B8 Yaw: -36.05 Pitch: -3.20 Roll: -1.00
Send to Rpi : 1 7F FE 80 27 FD 47 0 7D B8 Yaw: -36.05 Pitch: -3.20 Roll: -1.00
Send to Rpi : 1 7F FE 80 27 FD 47 0 7D B8 Yaw: -36.05 Pitch: -3.20 Roll: -1.00

```

Design Efforts - Electrical

```

4 // I2C Communication
5 #include <Wire.h>
6 // Basic demo for readings from Adafruit BNO08X
7 #include <Adafruit_BNO08X.h>
8
9 // set up IMU
10 #define BNO08X_RESET -1
11 #define BNO08X_RESET -1
12
13 struct euler_t {
14     float yaw;
15     float pitch;
16     float roll;
17 } ypr;
18
19 Adafruit_BNO08X bno08x(BNO08X_RESET);
20 sh2_SensorValue_t sensorValue;
21
22 #define SDA_W1 (D0) //TX
23 #define SCL_W1 (D1) //RX
24
25 #define ADDR_0 (e4) //peripheral address
26 uint8_t state = 0x00;
27
28 // CANBUS Communication
29 #include <Adafruit_MCP2515.h>
30 #define CS_PIN (D19)
31 #define CAN_BPMODE (166)
32 #define MTR_ADDR (0x01)
33 #define MTR_FREQ (100)
34 #define CAN_LEN (8)
35
36 Adafruit_MCP2515 mcp(CS_PIN);
37
38 uint8_t rpi_rec[CAN_LEN] = {0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0};
39 uint8_t can_recv[CAN_LEN] = {0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0};
40
41 uint8_t enable_motor[CAN_LEN] = {0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF};
42 uint8_t disable_motor[CAN_LEN] = {0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF};
43 uint8_t zero_motor[CAN_LEN] = {0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF};
44
45 uint32_t mtr_cmd = 0x0;
46
47 uint cur_time = micros();
48 bool motor_enabled {false};
49
50 char x = 0;
51
52 #define i2c_send_LEN 10 // new
53 uint8_t i2c_send[i2c_send_LEN]; // new
54 // state
55 // state
56 void quaternionToEuler(float qr, float qi, float qj, float qk, euler_t* ypr) {
57     float sqr = qr * qr;
58     float sqi = qi * qi;
59     float sqj = qj * qj;
60     float sqk = qk * qk;
61
62     ypr->yaw = atan2(2.0 * (qj*qk + qk*qr), (sqi - sqj - sqk + sqr));
63     ypr->pitch = asin(-2.0 * (q1*qk - qj*qr) / (sqi + sqj + sqk + sqr));
64     ypr->roll = atan2(2.0 * (qj*qk + q1*qr), (-sqi - sqj + sqk + sqr));
65
66     ypr->yaw *= RAD_TO_DEG;
67     ypr->pitch *= RAD_TO_DEG;
68     ypr->roll *= RAD_TO_DEG;
69

```

```

72 void setReports() {
73     if (!bno08x.enableReport(SH2_GAME_ROTATION_VECTOR)) {
74         Serial.println("Could not enable game rotation vector");
75     }
76 }
77
78 void setup() {
79     Wire.setSDA(SDA_W1);
80     Wire.setSCL(SCL_W1);
81
82     Serial.begin(115200);
83
84     Wire.begin(ADDR_0);
85     Wire.onRequest(requestEvent);
86     Wire.onReceive(receiveEvent);
87
88     if (!mcp.begin(CAN_BAUDRATE)) {
89         while(1) delay(10);
90     }
91
92     cur_time = micros();
93
94     while (!Serial) delay(10);
95
96     if (!bno08x.begin_I2C(BNO08X_I2CADDR_DEFAULT, &Wire)) {
97         while (1) { delay(10); }
98     }
99
100 void loop() {
101     delay(10);
102
103     if (bno08x.wasReset()) {
104         setReports();
105     }
106
107     if (!bno08x.getSensorEvent(&sensorValue)) {
108         return;
109     }
110
111     switch (sensorValue.sensorId) {
112         case SH2_GAME_ROTATION_VECTOR:
113             quaternionToEuler(
114                 sensorValue.un.gameRotationVector.real,
115                 sensorValue.un.gameRotationVector.i,
116                 sensorValue.un.gameRotationVector.j,
117                 sensorValue.un.gameRotationVector.k,
118                 &ypr
119             );
120
121             // pitch conversion to 16 bit
122             pitch16 = (int16_t)((ypr.pitch + 180.0) * (65535.0 / 360.0));
123
124             Serial.print("Yaw: "); Serial.print(ypr.yaw);

```

```

129     Serial.print("Yaw: "); Serial.print(ypr.yaw);
130     Serial.print(" Pitch: "); Serial.print(ypr.pitch);
131     Serial.print(" Roll: "); Serial.println(ypr.roll);
132     break;
133 }
134
135 int peak_time = micros();
136 if ((peak_time - cur_time) > 1e6 / MTR_FREQ) {
137     cur_time = peak_time;
138
139     switch(state){
140         case 0x00:
141             if (motor_enabled == true) {
142                 motor_enabled = false;
143                 CAN_Send(MTR_ADDR,disable_motor,8);
144             }
145             break;
146         case 0x01:
147             if (motor_enabled == false) {
148                 motor_enabled = true;
149                 CAN_Send(MTR_ADDR,enable_motor,8);
150             }
151             break;
152         case 0x02:
153             if(motor_enabled == true) {
154                 CAN_Send(MTR_ADDR, rpi_rec ,8);
155             }
156             break;
157         case 0x03:
158             CAN_Send(MTR_ADDR,zero_motor,8);
159             break;
160     }
161
162     int packetSize = mcp.parsePacket();
163     if (packetSize) {
164         byte i = 0;
165         while (mcp.available()) {
166             can_recv[i] = mcp.read();
167             i++;
168             if(i >= CAN_LEN){
169                 break;
170             }
171         }
172     }
173 }
174
175 }
176
177

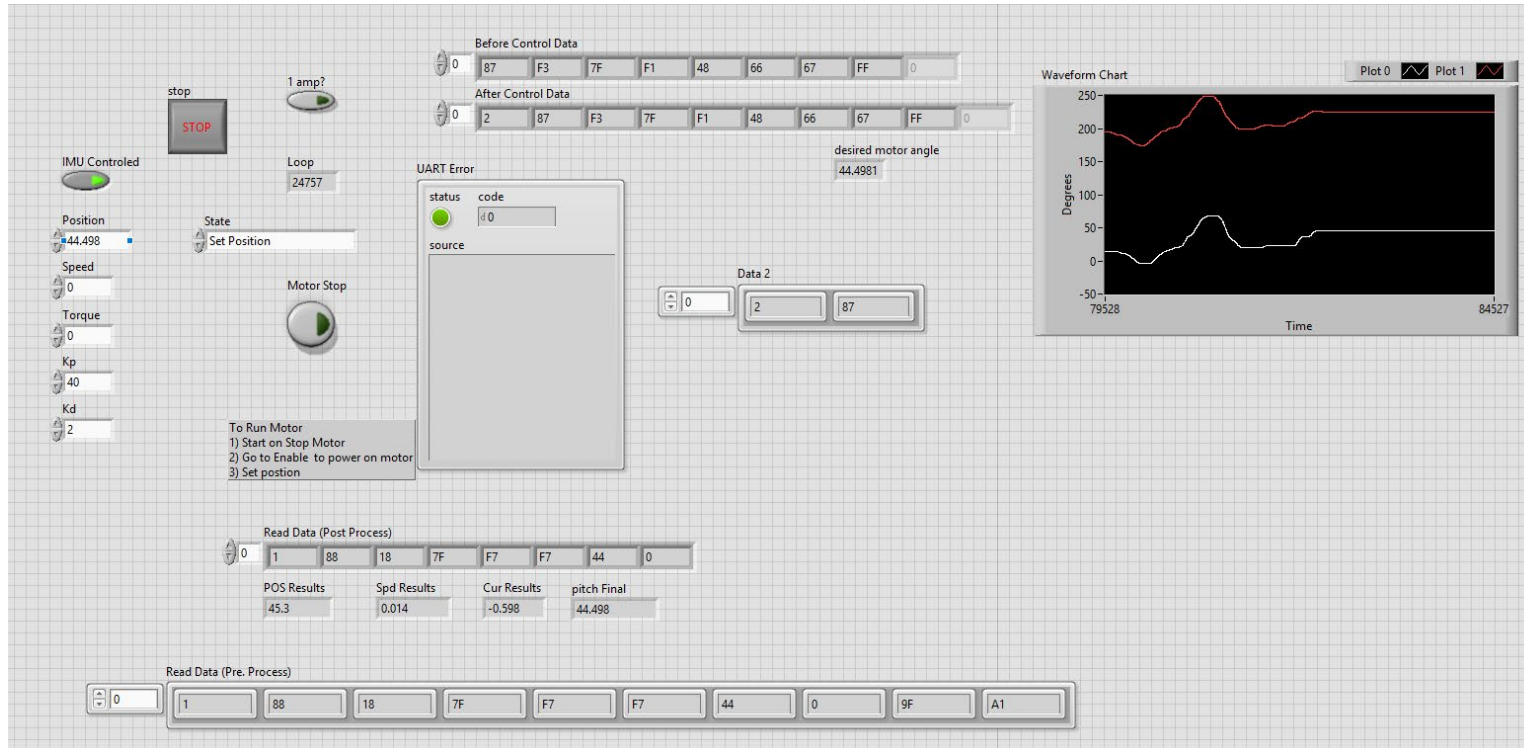
```

```

179 void CAN_Send(uint32_t cmd_id, uint8_t *data_addr, int len){
180     mcp.beginPacket(cmd_id, len, false);
181     for(byte i = 0; i < len; i++){
182         mcp.write(*(data_addr+i));
183     }
184     mcp.endPacket();
185 }
186
187 void requestEvent() {
188     // sending over 10 bytes (new)h16
189     for (int i = 0; i < 8; i++) {
190         i2c_send[i] = can_recv[i]; // original data
191     }
192     // adding new pitch
193     i2c_send[8] = (pitch16 >> 8) & 0xFF; //high
194     i2c_send[9] = pitch16 & 0xFF; //low
195
196     Wire1.write(i2c_send, i2c_send_LEN);
197
198     Serial.print("Send to Rpi : ");
199     for(int i = 0; i < i2c_send_LEN; i++) {
200         Serial.print(i2c_send[i], HEX);
201         Serial.print(' ');
202     }
203 }
204
205 void receiveEvent(int howMany){
206     int i = 0;
207     while(Wire1.available()) {
208         char c = Wire1.read();
209         if (i == 0){
210             state = c;
211         } else if (i > 0){
212             rpi_rec[i-1] = c;
213         }
214         i++;
215     }
216 }
217

```

Design Efforts - Electrical



Purchasing Plan - Budget

Hip Prosthetic Bill of Materials									
Category	Item No	Description	Primary Vendor	Unit Price	Quantity	Make/Buy	Manufacturer	Lead Time	Part Status
Main Assembly	1	AK80-64 KV80 Motor	CubeMars	\$ 911.77	1	Buy	CubeMars		In-Hand
Main Assembly	2	Angular Contact Bearing	MiMotion	\$ 10.83	3	Buy	Timken	3-4 Weeks	In-Hand
Main Assembly	3	1030L Gear	Zoro	\$ 200.00	1	Buy	Boston Gears	2-3 Weeks	In-Hand
Main Assembly	4	1030R Gear	Zoro	\$ 195.00	1	Buy	Boston Gears	2-3 Weeks	In-Hand
Main Assembly	5	Upper Shaft	NAU Machine Shop	Custom	1	Buy			In-Hand
Main Assembly	6	Lower Shaft	NAU Machine Shop	Custom	1	Buy			In-Hand
Main Assembly	7	Retaining Ring	DSR	\$ -	2	Buy	Hillman	1 Week	In-Hand
Main Assembly	8	Shaft Key	Amazon	\$ 5.56	2	Buy		3-4 Weeks	In-Hand
Main Assembly	9	Frame (Motor Side)	McMaster-Carr	\$ 46.01	1	Make		3-4 Weeks	In-Hand
Main Assembly	10	Frame (Bearing Side)	McMaster-Carr	\$ 31.55	1	Make		3-4 Weeks	In-Hand
Main Assembly	11	Lamination Plate	NextStep Prosthetics	\$ -	1	Buy	OttoBock	5 days	In-Hand
Main Assembly	12	Base Plate	McMaster-Carr	\$ 16.02	1	Make		3-4 Weeks	In-Hand
Main Assembly	13	Male Pyramid Adapter	Ebay	\$ 25.00	1	Buy		N/A	In-Hand
Main Assembly	14	Structure Enforcing Bar	McMaster-Carr		1	Make		3-4 Weeks	In-Hand
Hardware	15	M6-1x25 Bolt	HomeDepot	\$ 3.75	4	Buy	Everbilt	1 Week	In-Hand
Hardware	16	M6-1x20 Connector Bolt	Amazon	\$ 0.89	2	Buy	ACCU	10 days	In-Hand
Hardware	17	M3x12 Socket Cap Head Screw	Amazon	\$ 0.23	8	Buy	ACCU	10 days	In-Hand
Hardware	18	M4x10 Socket Cap Head Screw	Amazon	\$ 0.24	6	Buy	ACCU	10 days	In-Hand
Hardware	19	M6x15 Socket Cap Head Screw	Amazon	\$ 0.84	2	Buy	ACCU	10 days	In-Hand
Electronics	21	Power Supply 60 V, 5A	Amazon	\$ 71.99	1	Buy	Jesverty	5 days	In-Hand
Electronics	23	Adafruit CAN Controller	Adafruit	\$ 19.95	1	Buy	Adafruit	2-3 Weeks	In-Hand
Electronics	24	MicroSD Card	Adafruit	\$ 13.69	2	Buy	Adafruit	2-3 Weeks	In-Hand
Electronics	25	Buck Converter	Amazon	\$ 15.99	1	Buy	YABOANG	1 Week	In-Hand
Electronics	26	Breadboard Jumper Wire	Amazon	\$ 10.99	1	Buy	TODOELEC	5 days	In-Hand
Electronics	27	IMU Sensor	Adafruit	\$ 6.99	2	Buy	HiLetgo	5 days	In-Hand
Electronics	28	USB-C to USB-C	Amazon	\$ 5.00	1	Buy	Orseoose	5 days	In-Hand
Electronics	29	USB to USB-C	Amazon	\$ 7.99	1	Buy	Basesailor	5 days	In-Hand
Electronics	30	RUBIK Link V2.0	CubeMars	\$ 40.00	1	Make	CubeMars		In-Hand
Electronics	31	CAN Bus HAT	Waveshare	\$ 39.99	1	Buy	Waveshare		In-Hand
Electronics	32	36V Battery	Amazon	\$ 32.83	1	Buy	Amazon		In-Hand
Electronics	33	Battery Adapter	Amazon	\$ 5.82	1	Buy	Amazon		In-Hand
Electronics	34	Adapter	NAU Surplus	\$ 3.21	1	Buy	N/A		In-Hand
Electronics	35	Micro HDMI cable	BestBuy	\$ 31.81	1	Buy	Best Buy		In-Hand

Total Parts:	56
Total Purchased:	100%
Total In-Hand:	100%

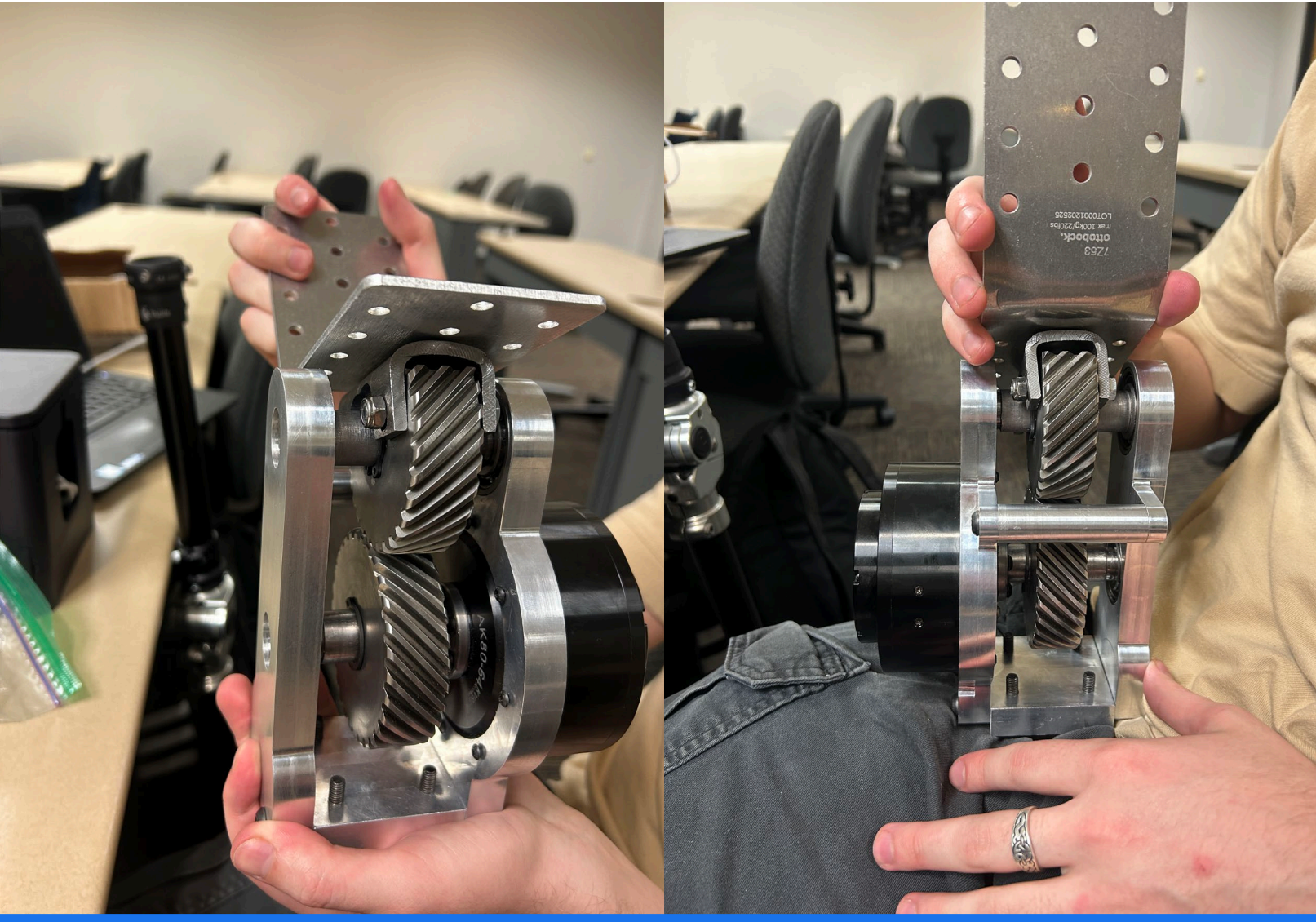
Expenses	\$	3,077.06
Available Balance	\$	1,422.94

Purchasing- Manufacturing Plan

Hip Prosthetic Bill of Materials [MANUFACTURING]

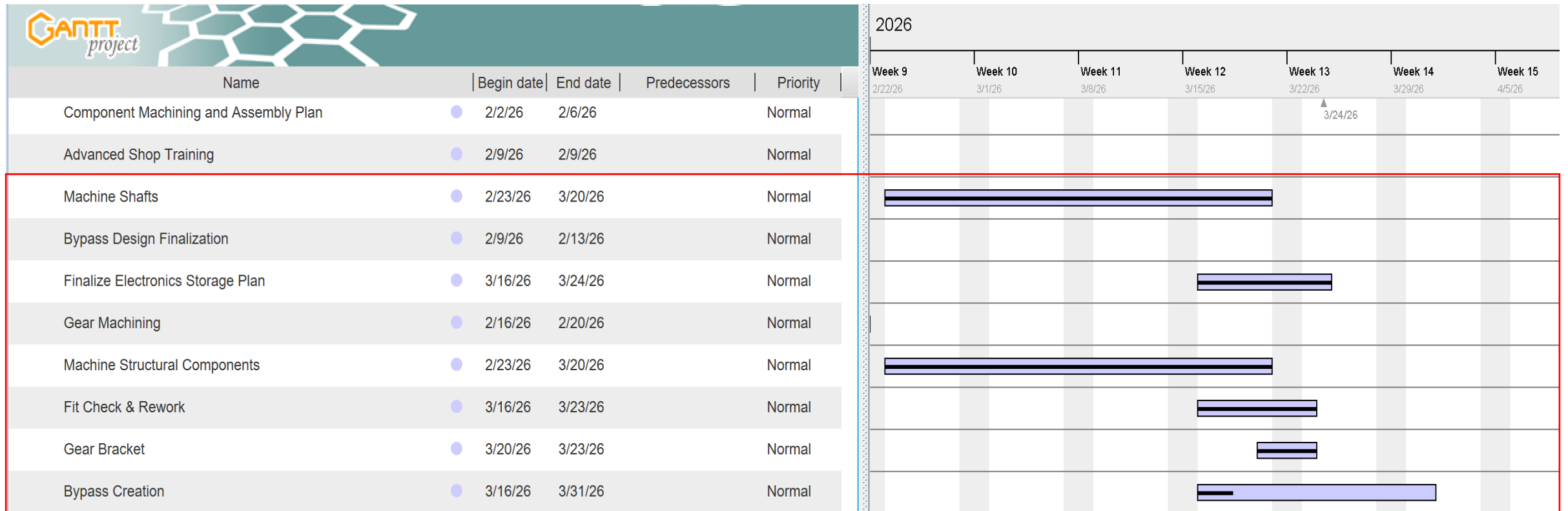
Item No	Description	Primary Vendor	Location	Quantity	Machinist	Process	Time Spent (Hr)	Estimated Tim	Progress %	Part Status
9	Frame (Motor Side)	McMaster-Carr	NAU Machine Shop	1	Ryan, Red Rock	CNC	1	1	100%	In-Hand
10	Frame (Bearing Side)	McMaster-Carr	NAU Machine Shop	1	Ryan, Red Rock	CNC	1	1	100%	In-Hand
12	Base Plate	McMaster-Carr	NAU Machine Shop	1	Matt/Aiden	Mill	5	3	100%	In-Hand
14	Structure Enforcing Bar	McMaster-Carr	NAU Machine Shop	1	Victoria	Lathe	4	2	100%	In-Hand
5	Upper Shaft	NAU Machine Shop	NAU Machine Shop	1	Quinn / Aiden	Lathe, Mill	5	3	100%	In-Hand
6	Lower Shaft	NAU Machine Shop	NAU Machine Shop	1	Victoria / Aiden	Lathe, Mill	7	4	100%	In-Hand
11	Mounting Bracket	McMaster-Carr	NAU Machine Shop	1	Matt	Mill	5	1	100%	In-Hand
3	1030L Gear	Zoro	East Valley Precision Machining	1	East Valley	Wire EDM Cutting	1	1	100%	In-Hand
3	1030L Gear	Zoro	NAU Machine Shop	0	Matt	Mill	1	1	100%	In-Hand

Total Parts:	8
Total Time Spent:	29
Total Progress:	100%

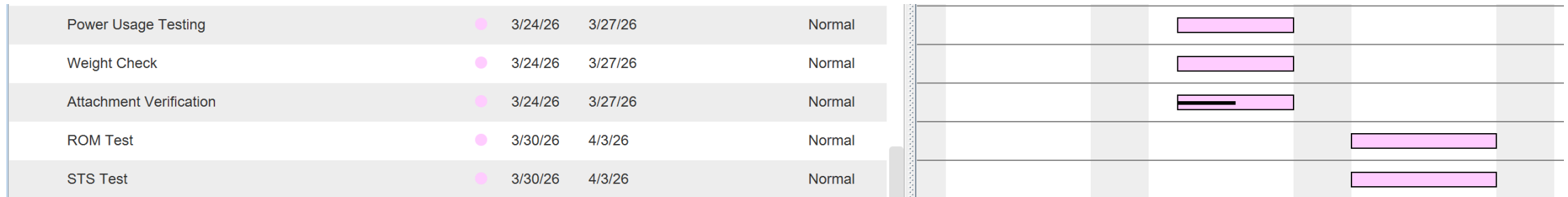


Physical Hardware Demonstration

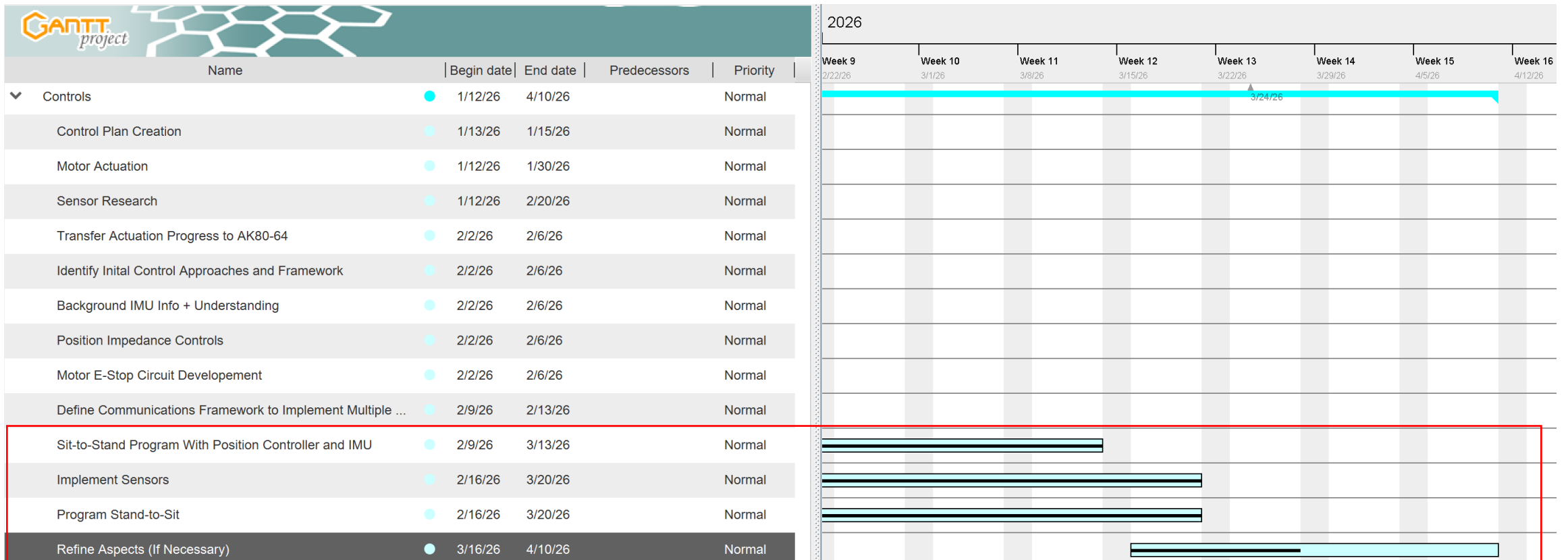
Gantt Chart (Manufacturing)



Gantt Chart (Testing)



Gantt Chart (Control)



Next Steps

- Finish assembling bypass for testing
- Complete motor activity protocol
- Begin testing!
 - System weight
 - Range of motion
 - Load
 - Motor performance

Thank you

