

ChIMP

Construction Manual

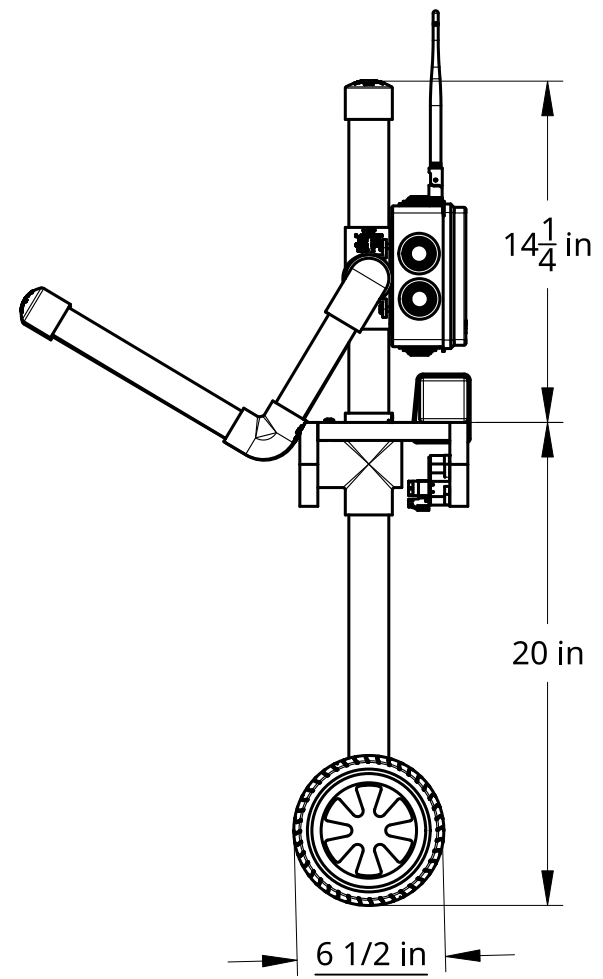
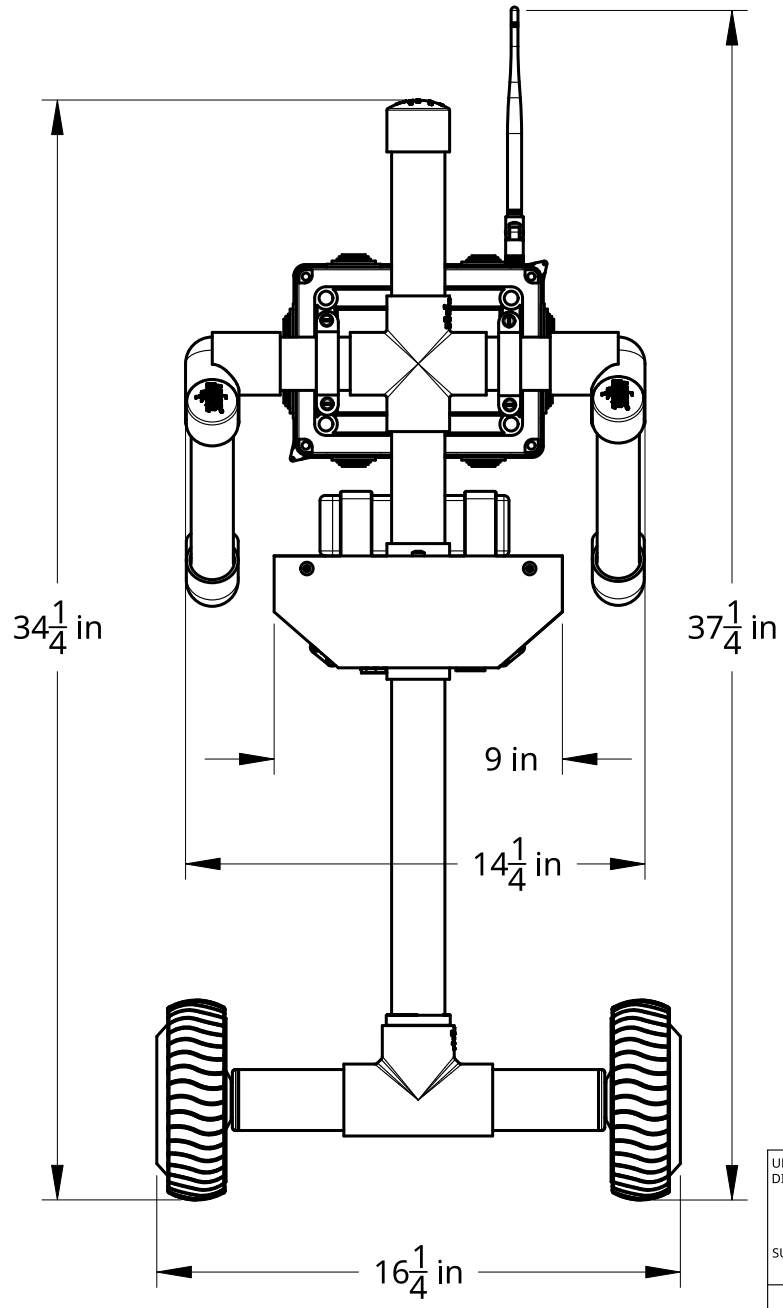


2

1

B

B





A

A

2

1

UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN MILLIMETERS ANGULAR = ± ° SURFACE FINISH  DO NOT SCALE DRAWING BREAK ALL SHARP EDGES AND REMOVE BURRS THIRD ANGLE PROJECTION 	NAME	DATE	TITLE <p style="text-align: center;">ChIMP Charleston Robotics</p>		
	DRAWN	CHRISTIAN VAN HORN			01/21/2022
	CHECKED				
	APPROVED				
MATERIAL	FINISH	SIZE	DWG. NO.	REV.	
		A		A	
		SCALE 1:6	WEIGHT	SHEET 1 of 1	

ChIMP FRAME BOM

PART	QUANTITY	NAME	NOTES	Purchase
1	1	1 1/2" PVC TEE	Readily available SCH 40 plumbing fixtures and pipe.	Home Depot, Lowes, Plumbing Supply, HRDW Store
2	2	1 1/2" X 4 5/8" PVC STUB AXLE	Readily available SCH 40 plumbing fixtures and pipe.	Home Depot, Lowes, Plumbing Supply, HRDW Store
3	1	1 1/2" TO 1 1/4" PVC ADAPTER	Readily available SCH 40 plumbing fixtures and pipe.	Home Depot, Lowes, Plumbing Supply, HRDW Store
4	1	1 1/4" X 13" PVC LEG	Readily available SCH 40 plumbing fixtures and pipe.	Home Depot, Lowes, Plumbing Supply, HRDW Store
5	1	1 1/4" PVC HIP CROSS	Readily available SCH 40 plumbing fixtures and pipe.	Home Depot, Lowes, Plumbing Supply, HRDW Store
6	1	1 1/4" X 6" PVC TORSO	Readily available SCH 40 plumbing fixtures and pipe.	Home Depot, Lowes, Plumbing Supply, HRDW Store
7	1	1 1/4" PVC CROSS	Readily available SCH 40 plumbing fixtures and pipe.	Home Depot, Lowes, Plumbing Supply, HRDW Store
8	1	1 1/4" X 7" PVC NECK	Readily available SCH 40 plumbing fixtures and pipe.	Home Depot, Lowes, Plumbing Supply, HRDW Store
9	1	1 1/4" PVC CAP	Readily available SCH 40 plumbing fixtures and pipe.	Home Depot, Lowes, Plumbing Supply, HRDW Store
10	2	1 1/4" X 4 1/2" PVC SHOULDER STUB	Readily available SCH 40 plumbing fixtures and pipe.	Home Depot, Lowes, Plumbing Supply, HRDW Store
11	2	1 1/4" TO 1" 90 DEG PVC SHOULDER	Readily available SCH 40 plumbing fixtures and pipe.	Home Depot, Lowes, Plumbing Supply, HRDW Store
12	2	1" X 6 1/2" PVC ARM	Readily available SCH 40 plumbing fixtures and pipe.	Home Depot, Lowes, Plumbing Supply, HRDW Store
13	2	1" 90 DEG ELBOW	Readily available SCH 40 plumbing fixtures and pipe.	Home Depot, Lowes, Plumbing Supply, HRDW Store
14	2	1" X 10 1/2" PVC FOREARM	Readily available SCH 40 plumbing fixtures and pipe.	Home Depot, Lowes, Plumbing Supply, HRDW Store
15	2	1" PVC CAP	Readily available SCH 40 plumbing fixtures and pipe.	Home Depot, Lowes, Plumbing Supply, HRDW Store
16	1	3/4" PVC FRONT PLATE	Cut from 1" x 4" PVC Trim Board	Home Depot, Lowes, Lumber Suppliers, Charleston Robotics
17	1	3/4" PVC HIP PLATE	Cut from 1" x 6" PVC Trim Board	Home Depot, Lowes, Lumber Suppliers, Charleston Robotics
18	1	3/4" PVC REAR PLATE	Cut from 1" x 4" PVC Trim Board	Home Depot, Lowes, Lumber Suppliers, Charleston Robotics
19	2	PLA PRINTED HUB ADAPTER SETS	3D Printed in PLA, 6 Wall, 50% Infill from our .STL File.	STLs, Charleston Robotics
20	2	6 1/2" HOVERBOARD MOTORS	Source new or used 22 volt 6 1/2" Hoverboard motors	Amazon (New), CraigsList, FaceBook Marketplace
21	5	HEX PAN HEAD ST4.8 X 36MM SCREW	Can substitute #10, Slotted, Phillips or Torx Drive	Amazon, Online Hardware Sellers
22	2	1 1/4" PIPE STRAP	Found at Lowe's, Home Depot, and Electrical Supply Stores.	Home Depot, Lowes, Electrical Supply, HRDW Store
23	4	M6 X 15 HEX PAN HEAD BOLT	Uses 4mm Hex Bit	Amazon, Online Hardware Sellers
24	4	M6 X 12 WASHER	ALL M6 can be replaced by 1/4" Hardware if necessary	Amazon, Online Hardware Sellers
25	4	M6 X 20 WASHER	ALL M6 can be replaced by 1/4" Hardware if necessary	Amazon, Online Hardware Sellers
26	4	M6 NYLON NUT	Uses 10mm Socket	Amazon, Online Hardware Sellers
27	10	BACKPACK INSERT	Comes Pre-installed in BACKPACK	Comes with LeMoTech Boxes, Amazon, Online Resellers
28	1	BACKPACK (200 X 155 X 80mm)	LeMoTech ABS Junction Box or Similar	Amazon, LeMoTech Online
29	6	L-SHAPED STANDOFF	Used in Gaming Consoles and PCB Mounting	Amazon, Adafruit, Online Electronics Sellers
30	6	M2.5 X 6 PAN HEAD MACHINE SCREW	Comes with L-SHAPED STANDOFF	Amazon, Adafruit, Online Electronics Sellers
31	6	M2.5 NYLON INSULATING WASHER	For insulating ODRIVE Mounting Screws	Amazon, Home Depot, Lowes, HRDW Store
32	6	M2.5 FLAT HEAD SELF TAPPING SCREW	Comes with L-SHAPED STANDOFF	Amazon, Adafruit, Online Electronics Sellers
33	1	ODRIVE VERSION 3.6	ODrive Brushless Motor Controller	ODrive.com

ChIMP FRAME INSTRUCTIONS

What follows are the basic ChIMP assembly instructions. In addition to these written instructions, the included Assembly Drawings and the online videos should make it clear that the ChIMP is truly simple to build with basic hand tools. We strongly advise you to watch the construction videos before beginning construction. If a picture is worth a thousand words, what is 30 frames per second worth? :-). Please take your time, have fun, and use the forum to get clarification on anything we haven't made perfectly clear.

Basic ChIMP frame construction instructions:

1. Using clear PVC cement, glue part #3, the 1 1/2" to 1 1/4" PVC ADAPTER into the top connection of part #1, the 1 1/2" PVC TEE. Please refer to construction drawings 001 and 002. Make sure you apply the cement to BOTH surfaces. Wipe away any stray cement, and hold the pieces together for at least a minute to allow the cement to begin to set.
2. Glue both part #2s, the 1 1/2" PVC STUB AXLES into part #1 as shown in construction drawings 001 and 002. Hold at least 1 minute to let glue set.
3. Temporarily insert part #4, the 1 1/4" X 13" PVC LEG into part #3. DO NOT GLUE!
4. Glue part #5, the 1 1/4" HIP CROSS into part #4. The HIP CROSS is identified by its having one shorter connection, which will face towards the rear as shown in construction drawings 002 and 003. This shorter connection allows the wiring to exit for attachment to the ODrive Motor Controller mounted on the #18 PVC REAR PLATE.
5. Glue the #17 PVC HIP PLATE down onto the top connection of #5 HIP CROSS, as shown in construction drawing 003. The beveled side of the 2" hole faces down. The HIP PLATE should be seated down fully onto the HIP CROSS, and the shorter connection must face the rear of the HIP PLATE (towards the battery strap slots). The longer connection on the HIP CROSS should face forward and be square with the front edge of the HIP PLATE.
6. Pre-drill and screw down the HIP PLATE onto the forward connector of the #5 HIP CROSS as shown on drawing 003. This is used to strengthen the hip connection and to prevent rotation.

ChIMP FRAME INSTRUCTIONS

7. Glue part #6, 1 1/4" x 6" TORSO into the top connection of part #5, the 1 1/4" HIP CROSS.
8. Glue part #7, the 1 1/4" PVC CROSS onto the #6 TORSO, making sure the cross is perpendicular to the #5 HIP CROSS as shown in construction drawing 002.
9. Insert part #8, the 1 1/4" x 7" PVC NECK into the top of part #7. DO NOT GLUE. Pre-drill and screw.
10. Glue part #9, the 1 1/4" PVC CAP onto part # 8.
11. Glue both part #10s, the 1 1/4" x 4 1/2" SHOULDER STUBS into part # 7, the 1 1/4" PVC CROSS, as shown in drawings 001 and 002.
12. Glue both part #15, 1" PVC CAPs onto their respective part #14s, their 1" x 10 1/2" PVC FOREARMS.
13. Glue both #14 1" x 10 1/2" PVC FOREARMS onto their respective part #13s, the 1" 90 DEG PVC ELBOWS.
14. Glue both part # 12s, 1" x 6 1/2" PVC ARMS onto their respective #13 1" 90 DEG PVC ELBOWS.
15. Slide both #11 1 1/4" to 1" 90 DEG PVC SHOULDERS, and secure them with masking tape, in the position shown in drawings 001 and 002.
16. Glue both part # 12, 1" x 6 1/2" PVC ARMS into the taped on #11 1 1/4" to 1' 90 DEG PVC SHOULDERS, making sure the arms are square and perpendicular as shown in drawings 001 and 002.
17. Pre-drill and install both part #11s onto the SHOULDER. These should be installed at a good angle to protect the ChIMP during a forward fall, and should both be mounted at the same angle.
18. Referring to construction drawing 004, remove the BACKPACK rear cover and drill the four 6mm holes in the BACKPACK, as shown. Using a Dremel or knife and chisel, remove the molded ribbing from the shaded areas in the drawing. This will allow the straps and washers to lay correctly while mounting.

ChIMP FRAME INSTRUCTIONS

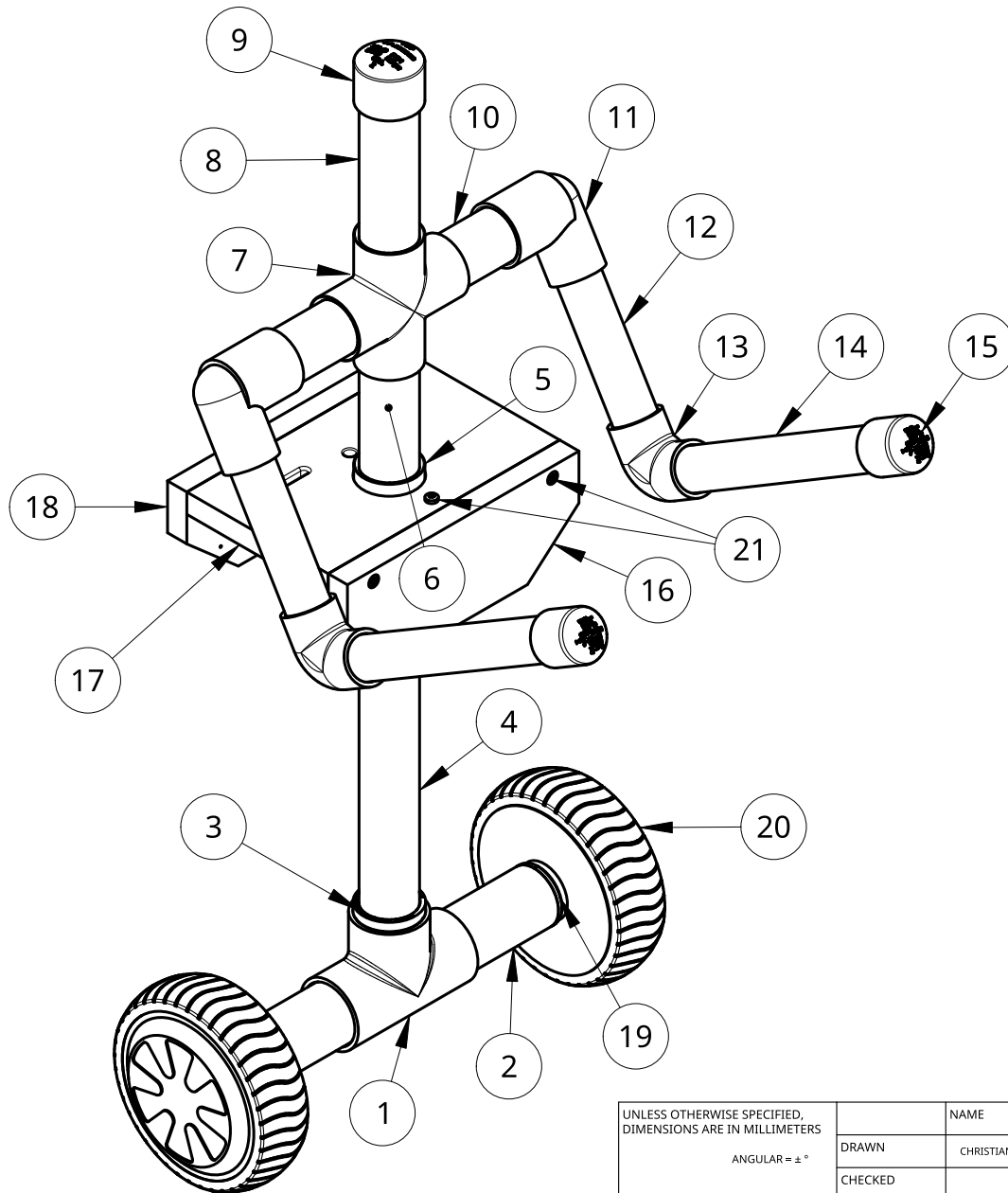
19. Mount the BACKPACK as shown in drawing 005. Use of a 4mm hex wrench and a 10mm socket on an extension is recommended, but a simple hex wrench and a pair of pliers would work in a pinch.

20. It's now time to fit the part #19 PLA PRINTED HUB ADAPTER SETs. First, install a set of adapters onto a Hoverboard Axle. Remove any excess material as necessary to make the two halves of the adapter meet and form a tight fit. The adapter halves must touch when strong pressure is applied. Repeat on the second adapter set.

21. Once this fit is accomplished, use sandpaper to slightly bevel the inner edge of the part #2 1 1/2" x 4 5/8" PVC AXLE STUBs, so that it will help with the Adapter installation.

22. Check the fit of the first adapter set into an AXLE STUB. They should be too tight to insert, but not too far off. Clamp a pair of Adapters to a work bench vertically. Using a length of 1" - 2" wide abrasive cloth tape (used by plumbers) smooth the outside of the Adapter until it can be inserted. This should be a TIGHT FIT that requires some force to fully insert. These Adapters are NOT GLUED but held in by friction and a single screw acting as a set screw. Repeat with second Adapter set. Do not install fully until the motor wiring is fully extended and ready for final assembly. The adapters should be installed with their seam running vertically, straight up and down. Pre-drill and install screws at the time of final assembly.

This concludes the assembly of the basic ChIMP frame. Please proceed by viewing the Electronics Videos and downloading the Electronics Assembly Manual



1	1 1/2" PVC TEE
2	(2) 1 1/2" x 4 5/8" PVC STUB AXLE
3	1 1/2" to 1 1/4" PVC ADAPTER
4	1 1/4" x 13" PVC LEG
5	1 1/4" PVC HIP CROSS
6	1 1/4" x 6" PVC TORSO
7	1 1/4" PVC CROSS
8	1 1/4" x 7" PVC NECK
9	1 1/4" PVC CAP
10	(2) 1 1/4" X 4 1/2" PVC SHOULDER STUB
11	(2) 1 1/4" to 1" 90 DEG PVC SHOULDER
12	(2) 1" x 6 1/2" PVC ARM
13	(2) 1" 90 DEG PVC ELBOW
14	(2) 1" x 10 1/2" PVC FOREARM
15	(2) 1" PVC CAP
16	3/4" PVC FRONT PLATE
17	3/4" PVC HIP PLATE
18	3/4" PVC REAR PLATE
19	(2) PLA PRINTED HUB ADAPTER SETS
20	(2) 6 1/2" HOVERBOARD MOTORS
21	(5) HEX PAN HEAD ST4.8 x 36mm

SCALE 1:5

UNLESS OTHERWISE SPECIFIED,
DIMENSIONS ARE IN MILLIMETERS

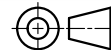
ANGULAR = ± °

SURFACE FINISH

DO NOT SCALE DRAWING

BREAK ALL SHARP EDGES AND
REMOVE BURRS

THIRD ANGLE PROJECTION



	NAME	DATE
DRAWN	CHRISTIAN VAN HORN	01/17/2022
CHECKED		
APPROVED		
	MATERIAL	FINISH

TITLE		
ChIMP Frame Charleston Robotics		
SIZE	DWG NO.	REV.
A	001	A
SCALE	WEIGHT	SHEET
1:6		1 of 1

2

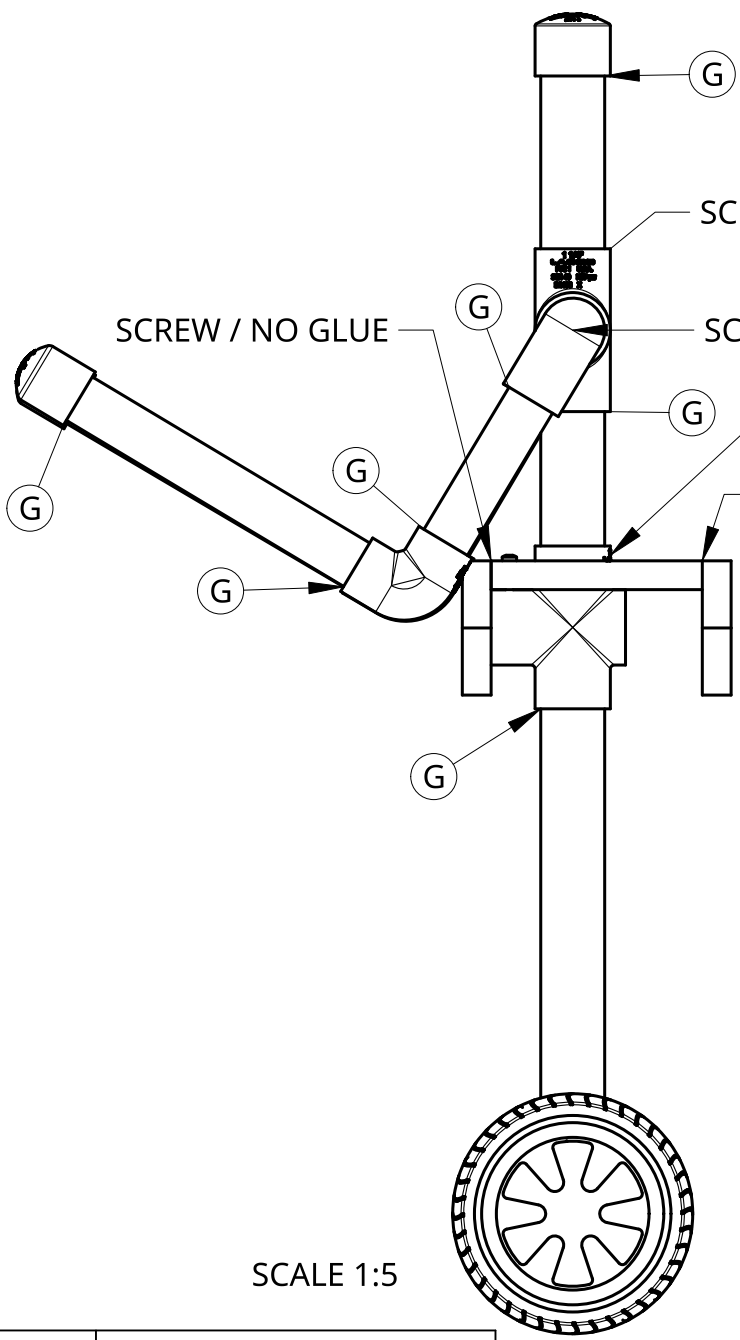
1

B

B

A

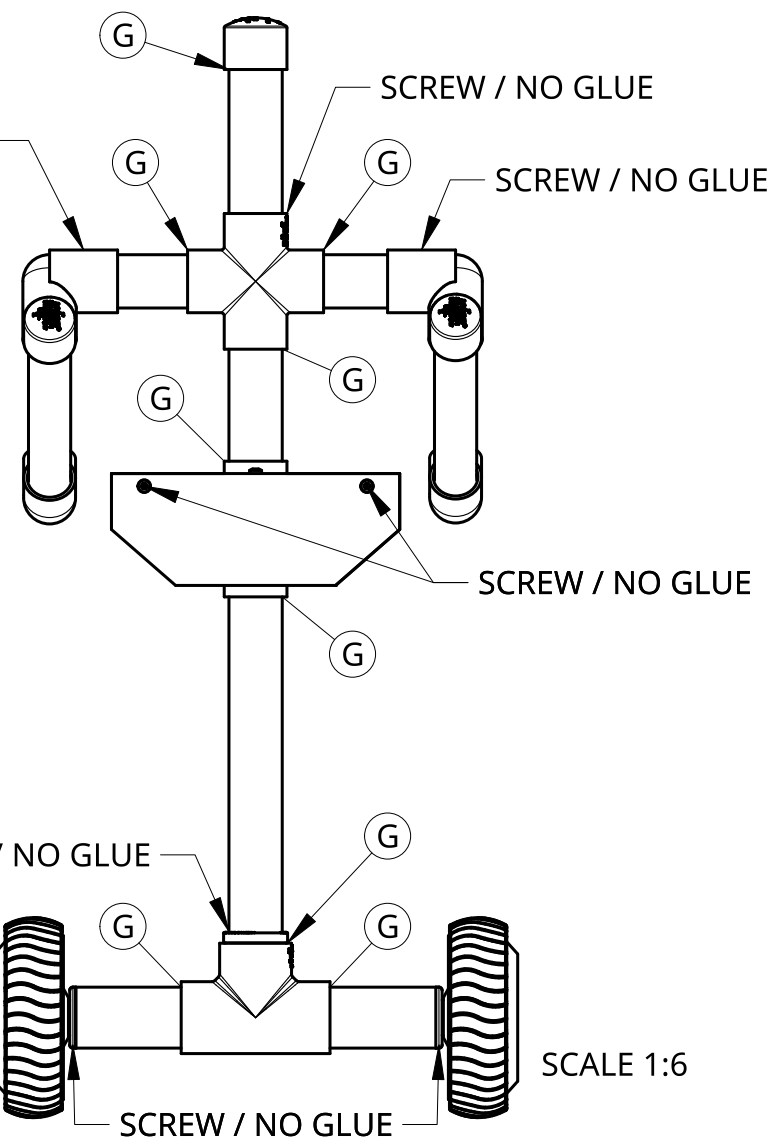
A



SCALE 1:5

SCREW: NECK, SHOULDERS 90s ONTO SHOULDER STUBS, LEG INTO BOTTOM TEE, FRONT AND BACK PLATES ONTO HIP PLATE, MOTOR ADAPTERS INTO STUB AXLES.

2

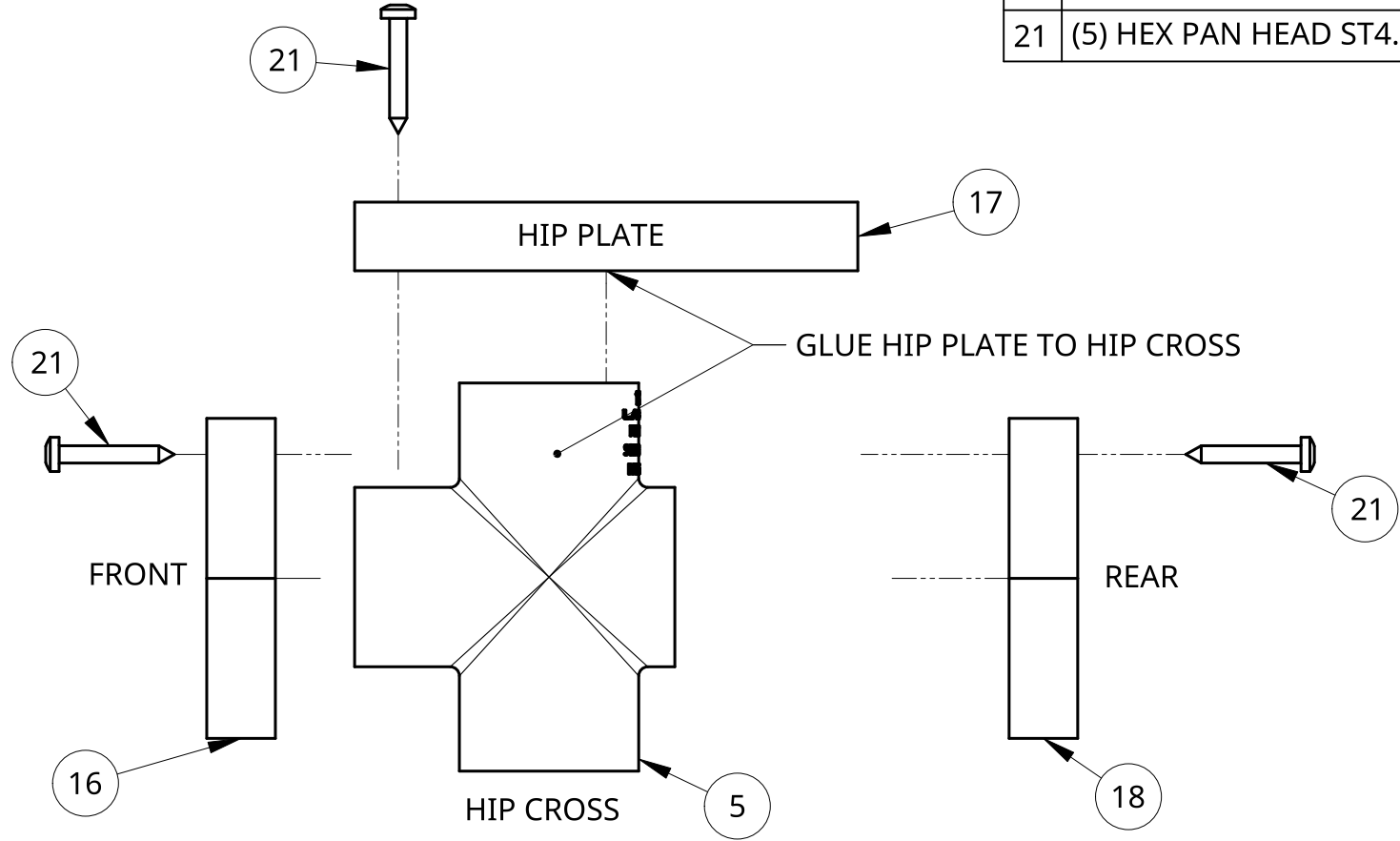




SCALE 1:6

1

UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN MILLIMETERS ANGULAR = ± ° SURFACE FINISH	DRAWN	CHRISTIAN VAN HORN	01/18/2022	TITLE ChIMP FRAME Charleston Robotics	
	CHECKED				
	APPROVED				
	DO NOT SCALE DRAWING				
BREAK ALL SHARP EDGES AND REMOVE BURRS				SIZE	A
THIRD ANGLE PROJECTION	MATERIAL	FINISH		DWG. NO.	002
				SCALE	1:5
				WEIGHT	
				SHEET	1 of 1
				REV.	A

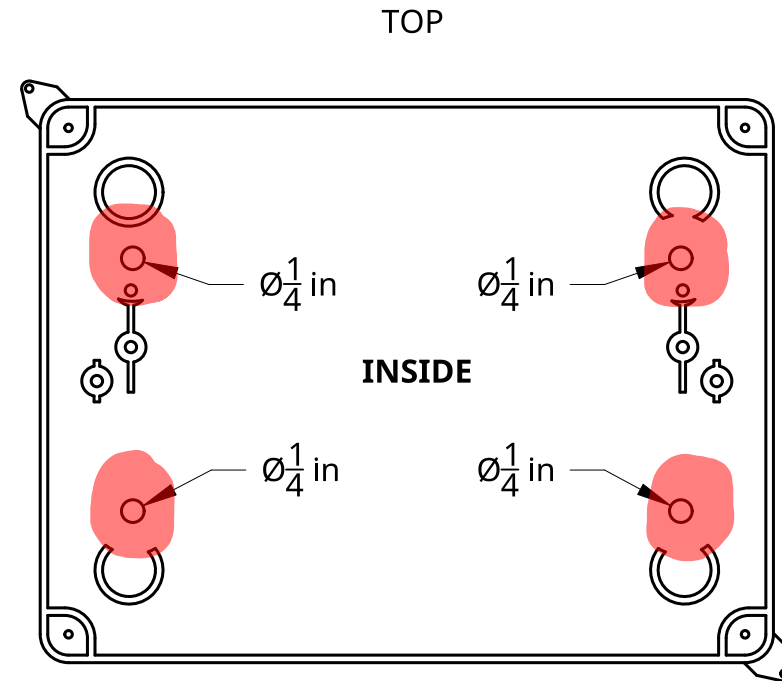
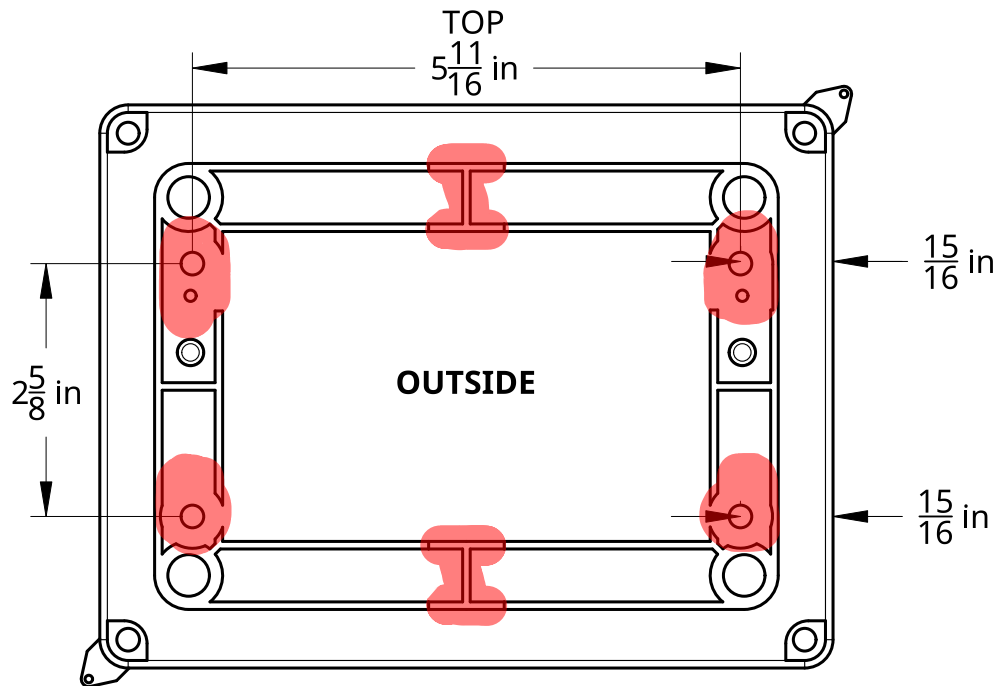
17	1" X 6" PVC HIP PLATE
18	1" X 4" PVC REAR PLATE
16	1" X 4" PVC FRONT PLATE
5	1 1/4" PVC HIP CROSS
21	(5) HEX PAN HEAD ST4.8 X 36



UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN MILLIMETERS ANGULAR = ± ° SURFACE FINISH  DO NOT SCALE DRAWING BREAK ALL SHARP EDGES AND REMOVE BURRS THIRD ANGLE PROJECTION 	DRAWN	CHRISTIAN VAN HORN	02/03/2022	TITLE Hip Exploded Drawing Charleston Robotics		
	CHECKED					
	APPROVED					
	MATERIAL	FINISH		SIZE A	DWG NO. 003	REV B
	SCALE 1:2		WEIGHT		SHEET 1 of 1	


2

1



DRILL (4) 1/4" HOLES FOR MOUNTING BOLTS AS INDICATED.

REMOVE ALL RIBS FROM SHADED AREAS TO ALLOW STRAPS & WASHERS TO LAND.

UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN MILLIMETERS ANGULAR = ± ° SURFACE FINISH $\sqrt{\quad}$ DO NOT SCALE DRAWING BREAK ALL SHARP EDGES AND REMOVE BURRS THIRD ANGLE PROJECTION 	DRAWN	NAME CHRISTIAN VAN HORN	DATE 01/19/2022	TITLE BACKPACK MODIFICATIONS Charleston Robotics		
	CHECKED					
	APPROVED					
		MATERIAL	FINISH	SIZE A	DWG NO. 004	REV. A
				SCALE 1:2	WEIGHT	SHEET 1 of 1

2

1

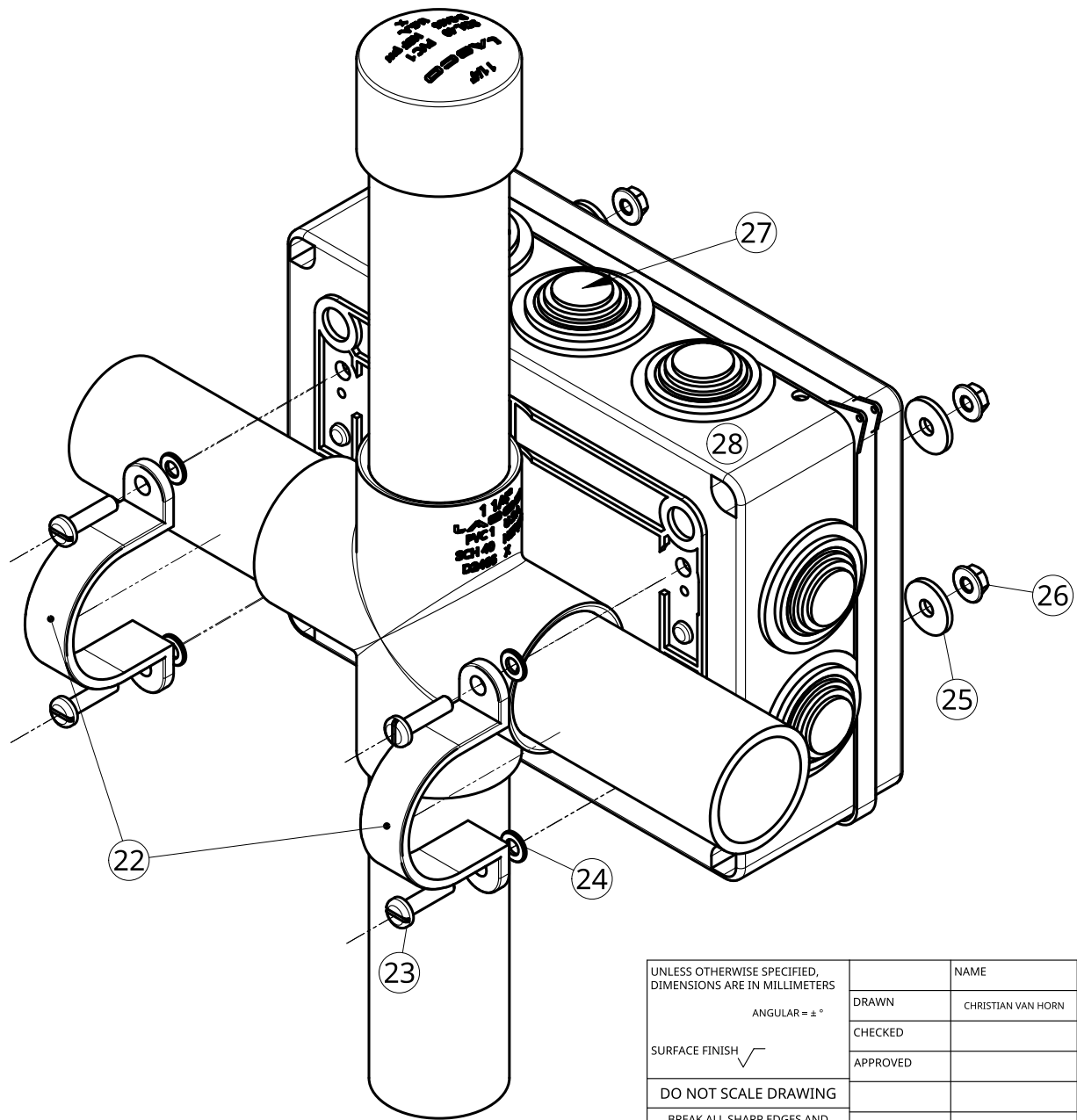
2

1

22	(2) 1 1/4" PIPE STRAP
23	(4) M6 x 15 HEX PAN HEAD BOLT
24	(4) M6 x 12 WASHER
25	(4) M6 x 20 WASHER
26	(4) M6 NYLOK NUT
27	(10) BACKPACK INSERT
28	BACKPACK (200 x 155 x 80)

B

B



A

A

UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN MILLIMETERS		NAME	DATE		
ANGULAR = ± °		DRAWN	CHRISTIAN VAN HORN	01/19/2022	
SURFACE FINISH $\sqrt{\quad}$		CHECKED			TITLE
DO NOT SCALE DRAWING		APPROVED			ChIMP BACKPACK Charleston Robotics
BREAK ALL SHARP EDGES AND REMOVE BURRS					
THIRD ANGLE PROJECTION		MATERIAL	FINISH	SIZE	DWG NO.
				A	005
				SCALE	WEIGHT
				1:2	
				SHEET	1 of 1
				REV.	A

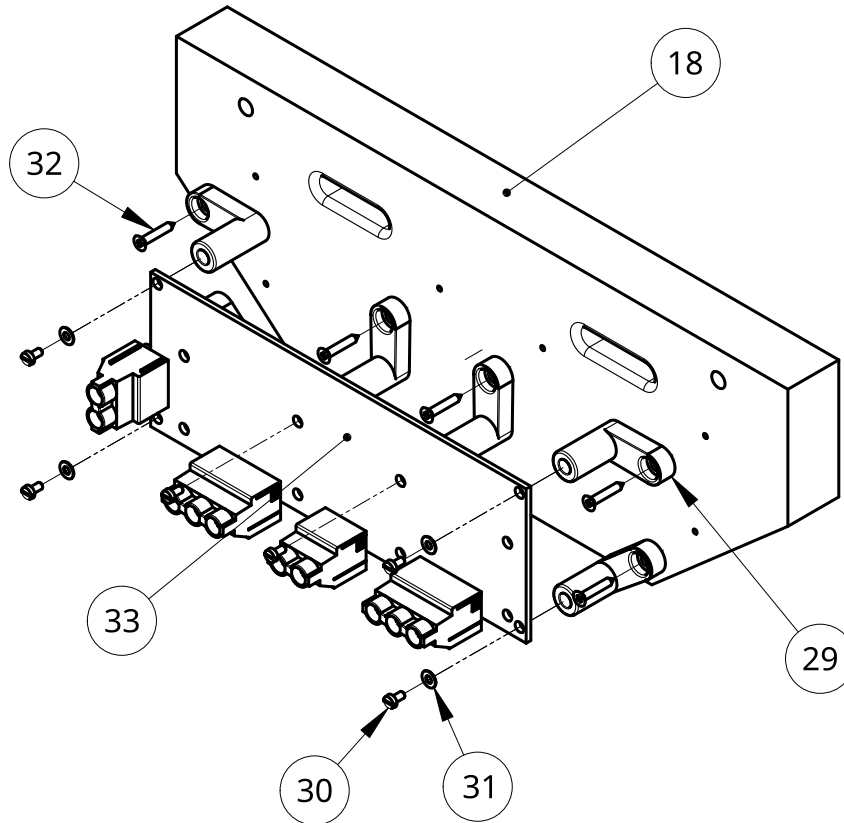
2



1

2

1

18	3/4" PVC REAR PLATE
29	(6) L-SHAPED STANDOFF
30	(6) M2.5 X 6 PAN HEAD MACHINE SCREW
31	(6) M2.5 NYLON INSULATING WASHER
32	(6) M2.5 FLAT HEAD SELF TAPPING SCREW
33	ODRIVE VERSION BOARD V3.6



UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN MILLIMETERS ANGULAR = ± ° SURFACE FINISH  DO NOT SCALE DRAWING BREAK ALL SHARP EDGES AND REMOVE BURRS THIRD ANGLE PROJECTION 		NAME	DATE	TITLE ODRIVE ASSEMBLY CHARLESTON ROBOTICS			
	DRAWN	CHRISTIAN VAN HORN	01/21/2022				
	CHECKED						
	APPROVED				SIZE	DWG NO.	REV.
					A	006	
	MATERIAL	FINISH		SCALE	WEIGHT	SHEET	
				1:2		1 of 1	

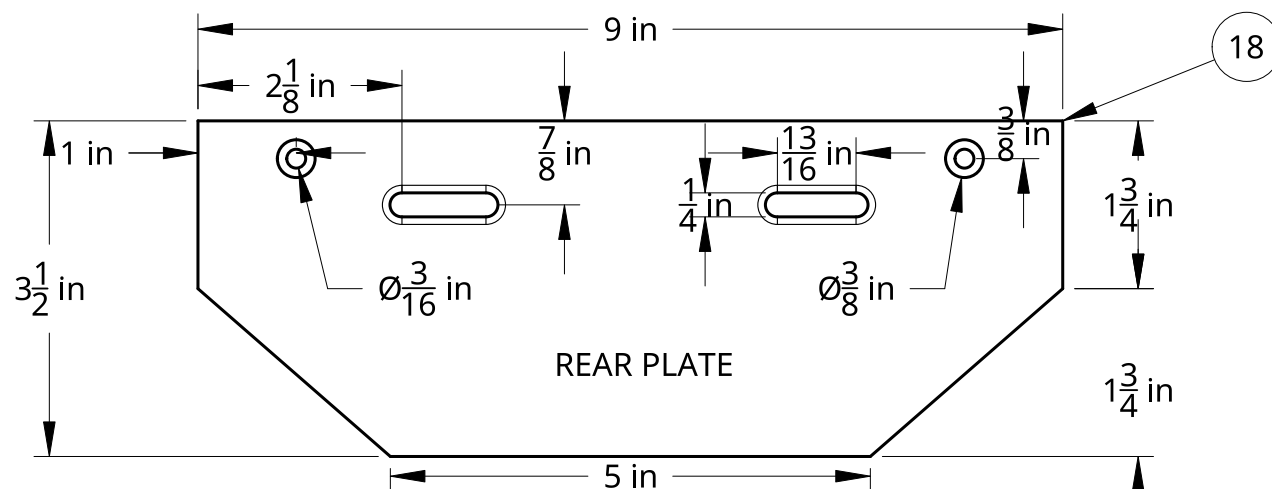
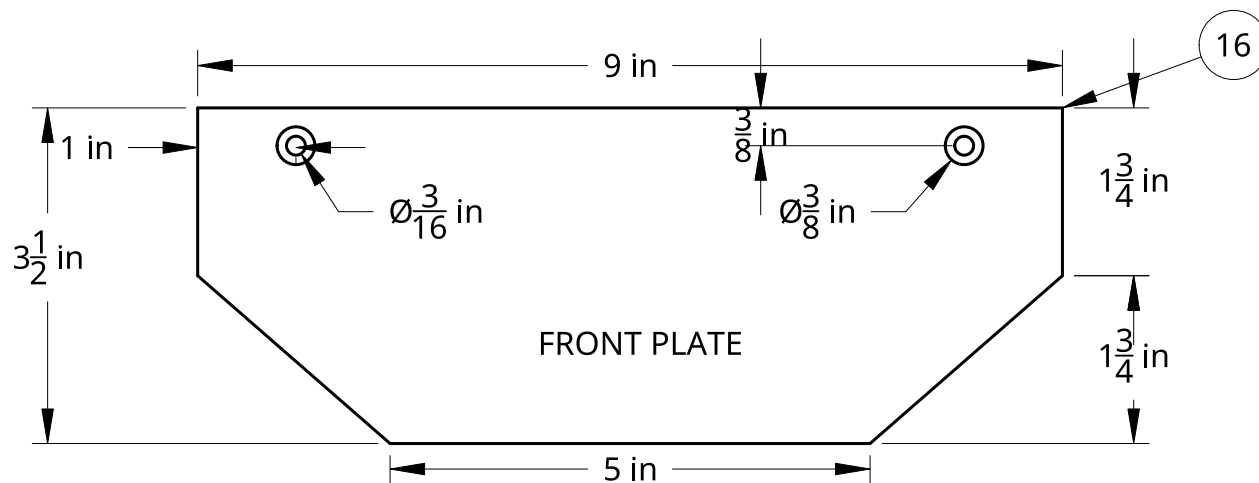
2

1

2

1

16	1" X 4" PVC FRONT PLATE
18	1" X 4" PVC REAR PLATE

UNLESS OTHERWISE SPECIFIED,
DIMENSIONS ARE IN MILLIMETERS

ANGULAR = ± °

SURFACE FINISH

DO NOT SCALE DRAWING

BREAK ALL SHARP EDGES AND
REMOVE BURRS

THIRD ANGLE PROJECTION



NAME DATE

DRAWN CHRISTIAN VAN HORN 02/03/2022

CHECKED

APPROVED

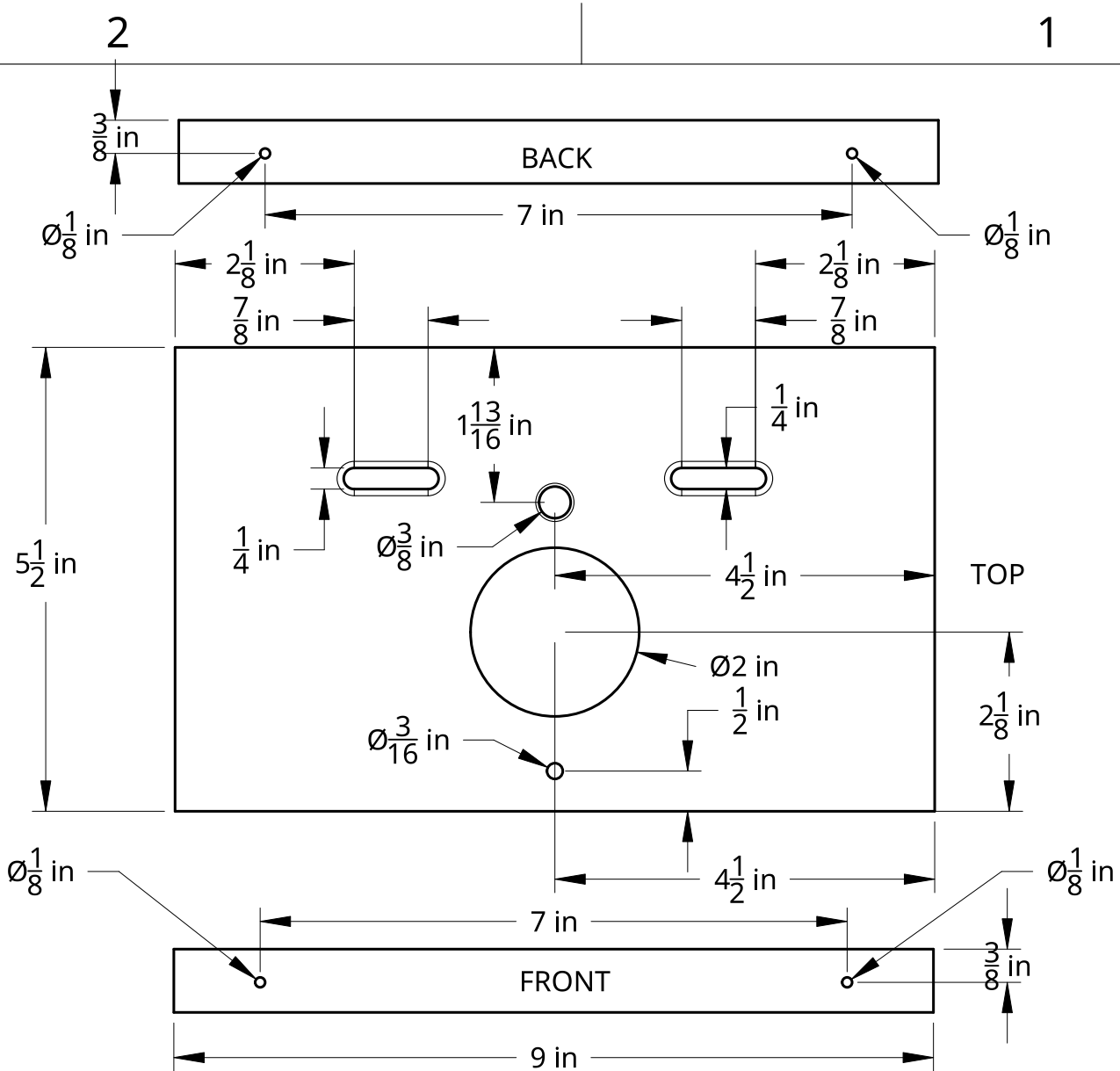
TITLE
FRONT AND REAR PLATES
CHARLESTON ROBOTICS

SIZE A DWG NO. 007 REV. A

SCALE 1:2 WEIGHT SHEET 1 of 1

2

1



UNLESS OTHERWISE SPECIFIED,
DIMENSIONS ARE IN MILLIMETERS

ANGULAR = ± °

SURFACE FINISH

DO NOT SCALE DRAWING

BREAK ALL SHARP EDGES AND
REMOVE BURRS

THIRD ANGLE PROJECTION

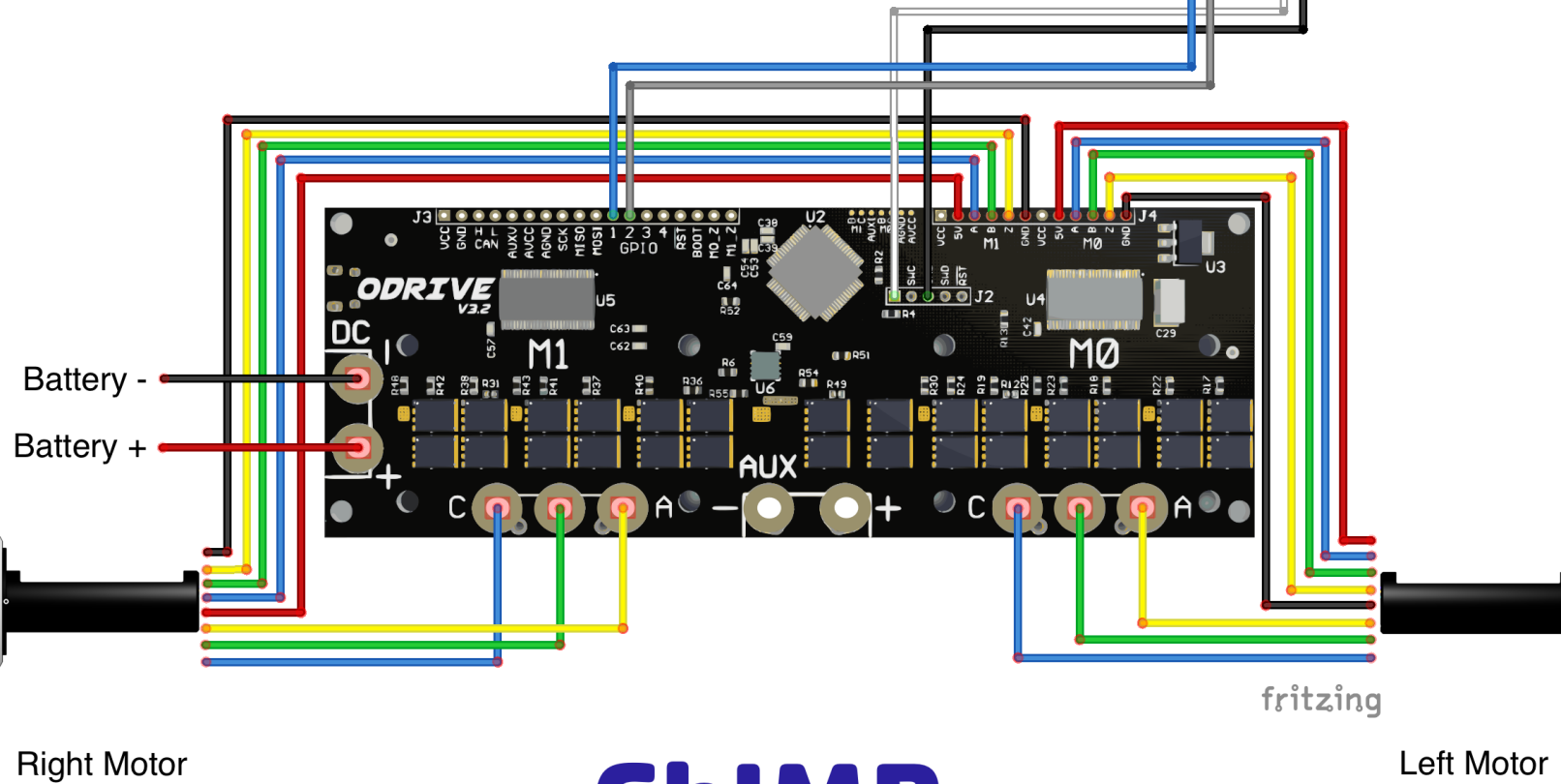
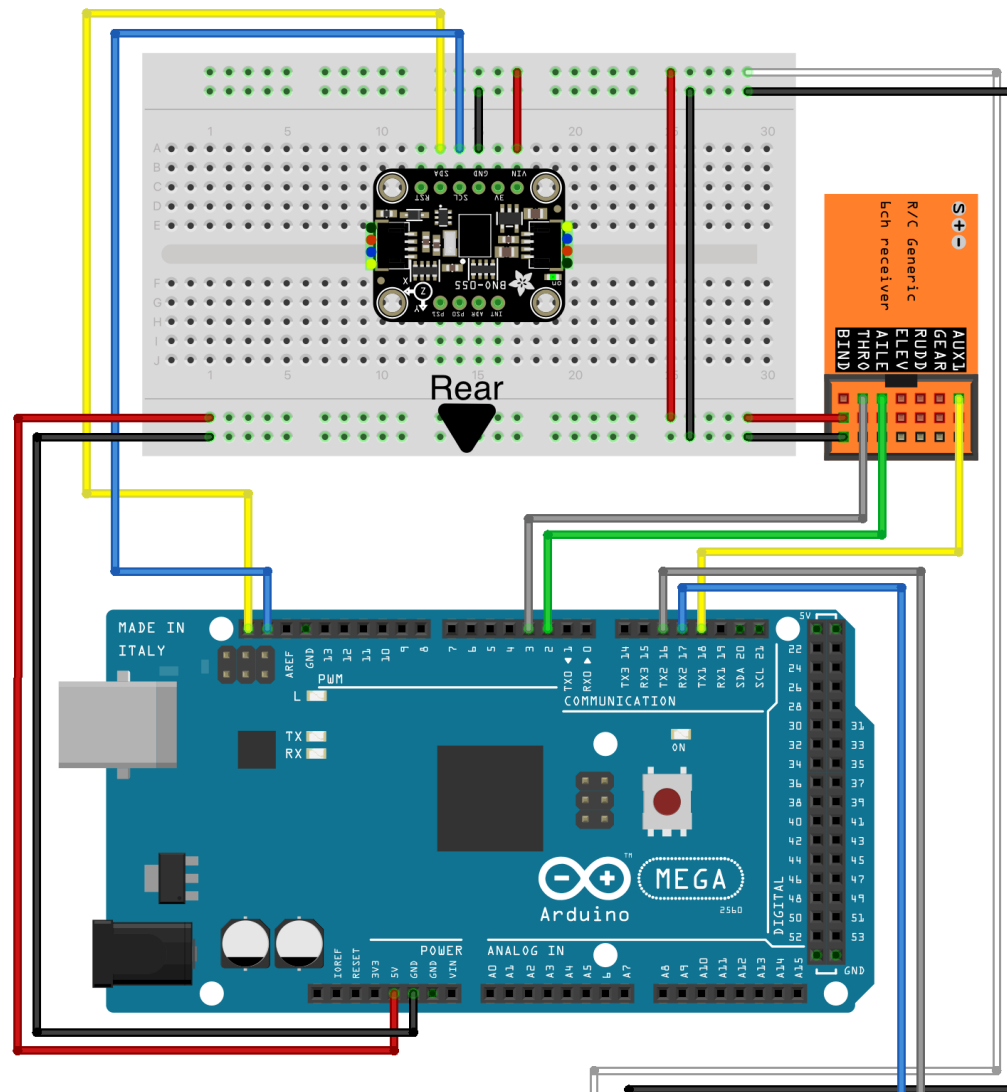
	NAME	DATE
DRAWN	CHRISTIAN VAN HORN	02/03/2022
CHECKED		
APPROVED		
MATERIAL	FINISH	

TITLE		
Hip Plate Charleston Robotics		
SIZE	DWG. NO.	REV.
A	008	A
SCALE	WEIGHT	SHEET
1:2		1 of 1

ChIMP BASIC ELECTRONICS BOM

PART	QUANTITY	NAME	NOTES	PURCHASE
1	1	ODRIVE MOTOR CONTROLLER VER. 3.6	We are using a genuine ODrive, but clones do exist.	ODrive.com
2	1	ARDUINO MEGA MICROCONTROLLER R3	Genuine or clone version of the Mega	Amazon, Arduino, Seed Studio
3	1	ADAFRUIT BNO055 IMU BREAKOUT BOARD	The BNO055 is available from many sources, including Amazon	Amazon, Adafruit
4	1	ASSORTED DUPONT WIRES	Purchase an assortment of lengths and connector combinations	Amazon, Adafruit, Online electronics sellers
5	15'	5 CONDUCTOR 22AWG RIBBON WIRE	Multi-colored stranded wires for extending Hoverboard sensor wires	Amazon, Digikey, Newark Electronics
6	10'	14GA BLUE STRANDED SILICONE WIRE	For extending Hoverboard motor power wires	Amazon, Digikey, Newark Electronics
7	10'	14GA GREEN STRANDED SILICONE WIRE	For extending Hoverboard motor power wires	Amazon, Digikey, Newark Electronics
8	10'	14GA YELLOW STRANDED SILICONE WIRE	For extending Hoverboard motor power wires	Amazon, Digikey, Newark Electronics
9	10'	22GA RED SOLID WIRE	Used for breadboard wiring	Amazon, Digikey, Newark Electronics
10	10'	22GA BLACK SOLID WIRE	Used for breadboard wiring	Amazon, Digikey, Newark Electronics
11	1	3D PRINTED ARDUINO MEGA MOUNT	STL Available on website	Charleston Robotics
12	1	400 PIN, 3 1/4" X 2 1/8" BREADBOARD	Adhesive backed, with buss strips lengthwise	Amazon, Adafruit, Online electronics sellers
13	2	6 1/2" 24 VOLT HOVERBOARD MOTOR	Salvaged or purchased new	Craigslist, FaceBook Marketplace
14	1	5000Ah - 6000Ah 22.2 VOLT LIPO BATTERY	XT60 Connector, 40C or higher discharge rate	Amazon, Radio Control Sellers, Hobby Shops
15	1	XT60 PIGTAIL WITH SILICONE WIRES	Used to connect battery to ODrive board	Amazon, Radio Control Sellers, Hobby Shops
16	1pk	3M COMMAND LRG PICTURE HANGERS	Holds up to 16 lbs, several pairs for mounting Arduino, etc.	Amazon, Home Depot, Lowes, HRDW Stores
17	1	6S LIPO CHARGER	AC or DC 6 cell Li-Poly charger w/ XT60 connector	Amazon, Radio Control Sellers, Hobby Shops
18	1	2.54mm DUPONT CONNECTOR KIT	With crimper and 1, 2, 3, 4, 5, and 6 pole connectors	Amazon, Digikey, Newark Electronics
19	1	2.4Ghz 5.8Ghz WiFi ANTENNA	U.FL/IPEX to RP-SMA female stub antenna	Amazon, Adafruit, Online Electronics Sellers

Charleston Robotics



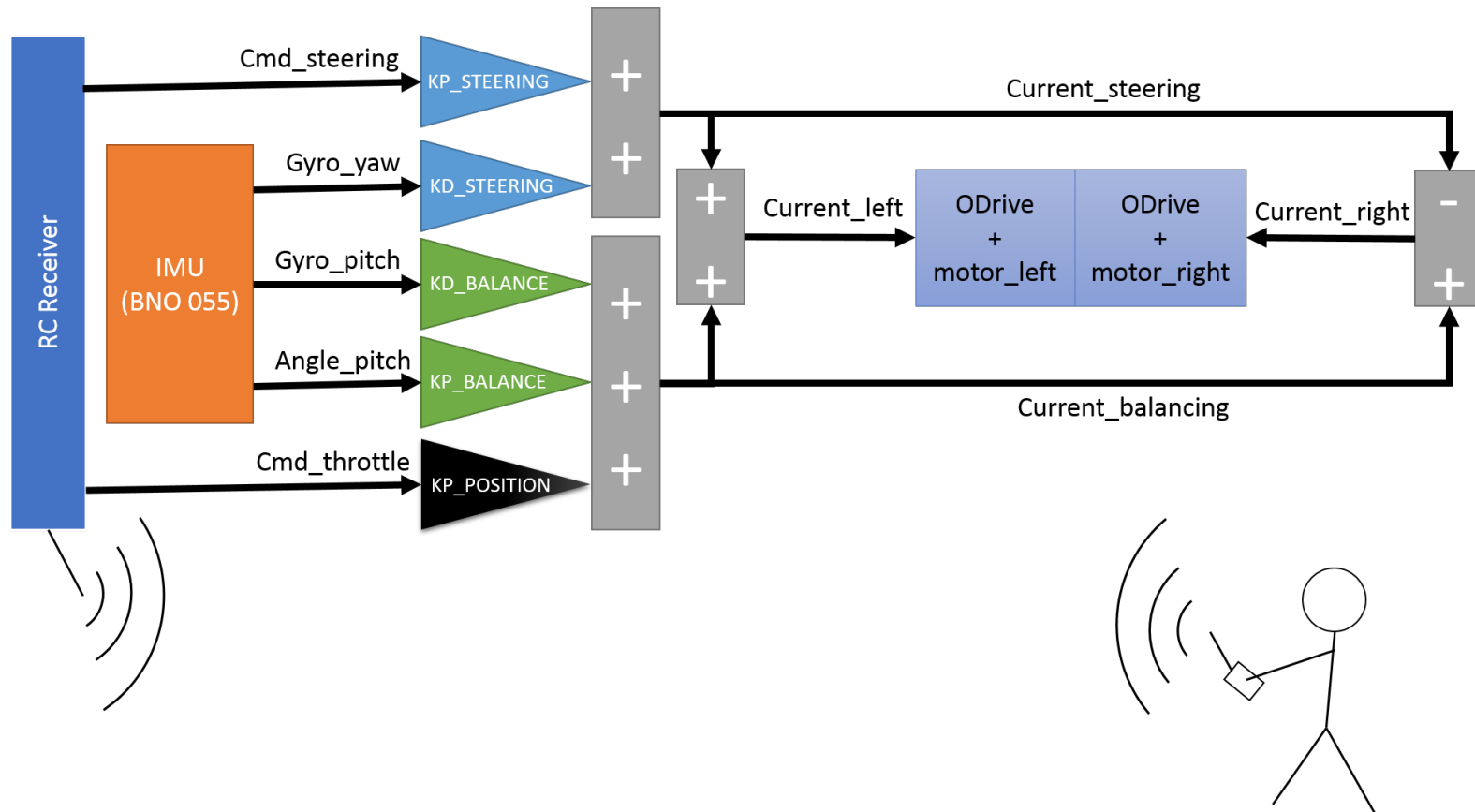
ChIMP

ChIMP BASIC ELECTRONICS INSTRUCTIONS

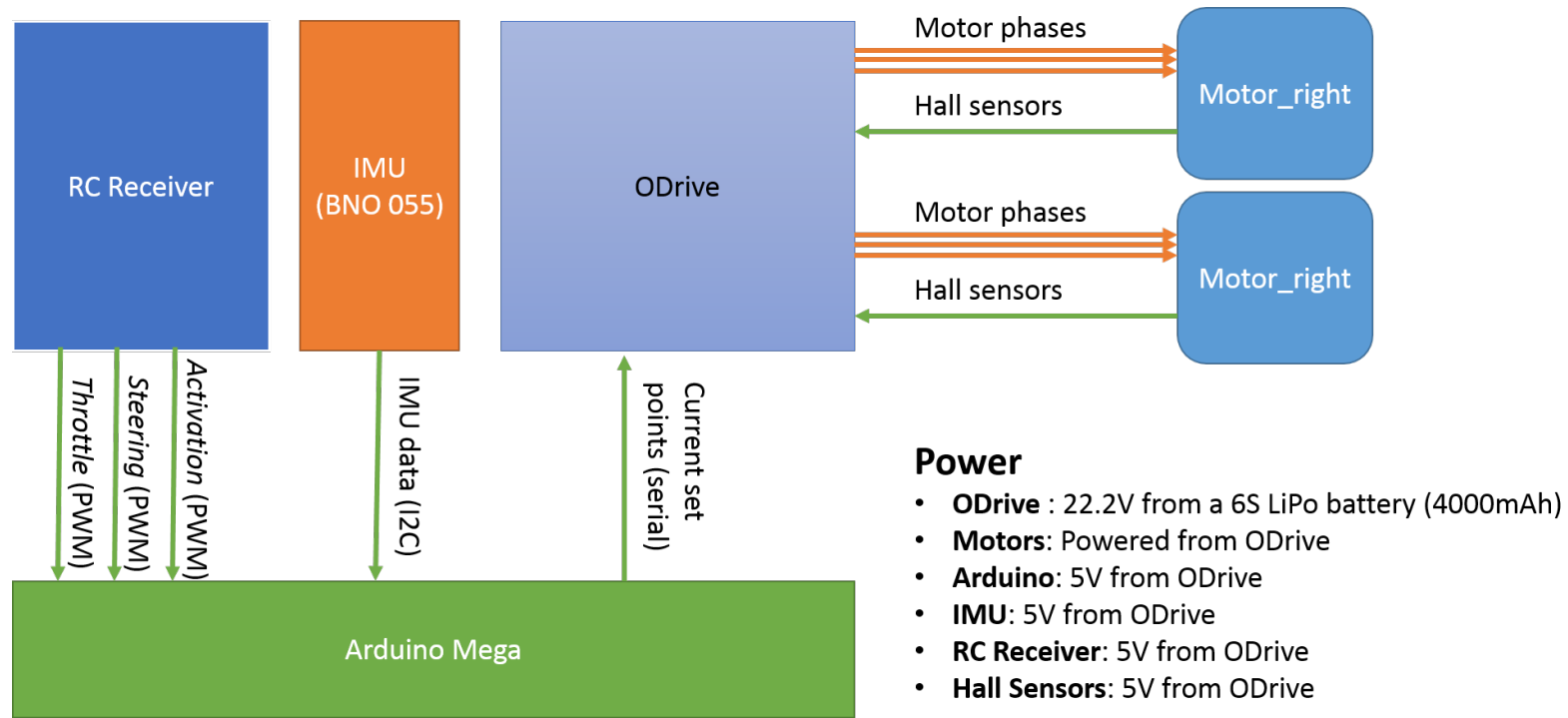
What follows is a description of Chimp's electronic systems and basic wiring instruction. Please read the Construction Manual thoroughly, and watch both electronics videos to familiarize yourself with the ChIMP wiring before wiring up your ChIMP! Take your time, and don't hesitate to visit the forum with any questions you might have.

In a nutshell: Inside ChIMP's backpack is an Arduino Mega (or clone) microcontroller, communicating with a Bosch BNO055 Inertial Measuring Unit. The Arduino is running a custom Sketch (program) and receives (PWM) control signals from a **Radio Control** receiver. Using the info from the IMU along with commands from the receiver, it instructs the motor drive board (ODrive 3.6) to control the brushless Hoverboard motors in such a way that the robot remains upright and balanced while driving as commanded.

A Block Diagram of the ChIMP Control System:



A Block Wiring Schematic:



Notes about following the Chimp Wiring Schematic:

The Fritzing Wiring Diagram is included in this construction manual. Here are additional notes, listed from the top of the diagram down, starting with the Adafruit BNO-055 IMU:

1. The Adafruit Bosch BNO-055 IMU comes in several varieties, and **pin placement varies!** Please connect the Arduino Mega Pin #21, (the far left pin closest to the USB connector) to the **SCL** Pin on the IMU with a yellow wire. Connect the Arduino Pin #20 (right next to pin 21) to the **SDA** Pin on the IMU. **These labels for its I2C connections override any other diagram or instructions.**

2. The Radio Control Receiver takes its 5 Volt power from the breadboard as well. Run a red wire to the Positive Buss, and a Black wire to the Negative. ALL the Positive pins on the receiver are wired together, as are all of the Negative Pins. You can connect your black and red power wires to their respective pins anywhere on the receiver.
3. The Signal pins on the Receiver are each marked with a servo channel or purpose. On the Chimp, the Steering channel is usually marked Steering or Channel 1 on the receiver. No matter which channel you choose for steering, that channel must be wired to **Pin #2 on the Arduino**.
4. Throttle is often Channel 2 or Throttle on the receiver. No matter which you choose for Throttle, it must be wired to **Arduino Pin # 3**.
5. Finally, the Enable channel, which could be any channel on the receiver that can be controlled by a switch or a button, must be wired to **Arduino Pin #18**. The Enable is used as a Failsafe, and must be activated in order for the Chimp to function.
6. The Arduino gets its power from the Buss on the Breadboard, which in turn gets its power from the ODrive's 5 Volt Regulator. This is kept at a constant 5 Volts no matter the condition of the battery.
7. The ODrive 3.6 power pins are on the **J2** Header. The White wire connects to the 5V Pin, and the Black wire connects to GND. These run up and connect to the Buss on the breadboard.
8. The ODrive UART connections are on the **J3** Header, with the Blue wire connecting to **GPIO Pin #1**, and Grey wire connecting to **GPIO Pin #2**. Interestingly, the UART Transmit Pin (TX2) on the Arduino connects to the ODrive's Receive (RX) Pin, set here for GPIO 2. The Arduino's Receive Pin (RX2) is connected to the ODrive's Transmit Pin, GPIO 1.
9. On the ODrive 3.6, there are two separate sets of motor connectors, **M1** and **M0**, The ChIMP's right hand motor (HIS right hand) utilizes M1, with the left hand motor wired to M0. The heavier Phase Wires are connected to the bottom terminals, marked C, B, and A, in any combination. I used Blue, Green, and Yellow on the Wiring Diagram.
10. The five lighter gauge wires from the motor are for the Hall Sensors, which tell the ODrive the position of the Motor Rotor. It is a known issue that the Hall Sensor signals can be disturbed by interference due to the longer wire run and the close proximity of the higher current Phase Wires. Charleston Robotics has designed a PC Board that has filter capacitors installed, as well as a pin header and connection points for the extension wires. The extension wires are usually soldered onto the motor Hall wires, with the Red wire going to 5V on the tiny extension board, and Black going to GND . The other three wires can attach to A ,B, and Z in any order, but you should keep the pattern the same on both motors. In reality, the Auto-Setup Script takes care of both the Phase Order and the Hall Sensor order!
11. Connect an XT60 pigtail into the ODrive 3.6 Battery Connector, marked "DC". Black to "- ", and Red to "+".
12. Finally, the ChIMP's Battery can be anywhere between 3500mAh and 6000mAh or more, This is a 6 Cell 22.2VDC LiPoly Pack. The Hoverboard Motors can be 24V OR 36V...Both will run with this battery. Make sure the battery charger you have can support "6S", six cell LiPo batteries.

Programming, Testing, and Setting up a ChIMP

If this seems foreign, do not fear. Our community is there to support you. No one is born with this knowledge!

THE ARDUINO MEGA:

Arduino IDE (Integrated Development Environment) Installation and setup:

The first step is to download the latest Arduino IDE from [here](#), and install it on your PC. This program allows you to modify and upload programs (Sketches in Arduino speak) onto your Arduino Microcontroller.

After installation, connect your Arduino Mega to the PC via USB and restart the IDE. Under Tools > Board, select “Arduino/Genuino Mega or Mega 2560” and under Tools > Port select the port your Arduino is connected to.

Open the Serial Monitor under Tools > Serial Monitor and set the baudrate to 115200 (just to make sure there won't be issues with this later). Close the Serial Monitor again.

Test your general setup:

Under File > Examples < 01.Basics select ”Blink”. The IDE will display the respective Code in a new window. Click on “Upload” (the right pointing arrow in the top left corner in the IDE). The code should get compiled and then be uploaded to the Arduino. After successful

upload, the Arduino's LED should blink steadily. You can change the values of delay() in lines 34 and 36 (times in milliseconds) and re-upload to change the blinking pattern. **This test is powered by the USB connection...NO BATTERY NEEDED, AND IT SHOULD REMAIN UNCONNECTED!**

Test the IMU:

After following the Electronics videos, and wiring up your ChIMP as shown in the included Fritzing Wiring Diagram (and then double checking EVERYTHING), it's time to test the IMU.

To easily use this relatively complicated sensor we need two software libraries that will do all the heavy lifting for us. Those are also provided by Adafruit [here](#) (sensor driver) and [here](#) (unified sensor interface). **Installing and using libraries from GitHub is explained in Annex I.**

To test if the sensor is working, start (or restart) your Arduino IDE, connect the Arduino with attached IMU and open the sensorapi example (as described [here](#)). Compile and download the code to the Arduino, open the Serial Monitor (to be found under tools; make sure the baudrate matches that in the code) and watch the stream of orientation data coming in as you tilt and turn your ChIMP! You are getting there! **This test is powered by the USB connection...NO BATTERY NEEDED, AND IT SHOULD REMAIN UNCONNECTED!**

Test the RC receiver:

Standard RC receivers run on 5 Volts, so they require 5V and GND from a power supply. The connectors of a receiver typically have three pins for each servo, with one for 5V, one

for GND and one for the signal. The Positive and Ground power pins are each wired in parallel (all together), hence GND can be connected to any GND pin and 5V can be connected to any 5V pin.

The signal for each channel is encoded using [PWM \(pulse width modulation\)](#). For standard RC equipment, a pulse is sent 50 times per second. The duration of this pulse, which typically ranges from 1 to 2 ms long, is the signal. ChIMP needs three signals (throttle, steering and enable), thus three signal pins of the receiver are connected to the Arduino. You choose the 3 channels based on how your RC Transmitter is laid out, and how you want your controls to work. There are different ways of reading PWM input signals on an Arduino, but the most efficient one is to use [interrupts](#). This ensures that the Arduino's processor is never busy waiting for the next pulse. By using interrupts, special hardware inside the Arduino's microcontroller is used for triggering an [interrupt service routine](#) only when the processor really needs to do something (i.e. calculating the duration of the pulse). The downside is that this only works on dedicated pins, **of which smaller Arduinos only have two**. This is the reason that ChIMP needs an Arduino Mega which provides six interrupt pins.

For testing, wire up the receiver as described in the Chimp Electronics Videos and the Wiring Diagram. Connect the three channels you want to use to control the robot to the Arduino pins 2, 3 and 18 (as defined in the [config file](#)). Normally Pin 2 is for Steering, Pin 3 is for Throttle, and Pin 18 is for the Enable (used as a failsafe, the Enable must be "On" for the Chimp to balance and operate)

Upload the [receiver test](#) sketch in the tests subfolder to the Arduino. This test is powered by the USB connection. **NO BATTERY IS NECESSARY FOR THIS TEST, AND IT SHOULD REMAIN UNCONNECTED!**

Once the sketch is uploaded and running, open the Serial Plotter in the Arduino IDE (Tools > Serial Plotter). As you move the sticks on the RC transmitter, you should see three curves react accordingly. They show the three signals' pulse widths in microseconds and should be centered roughly around 1500, ranging from 1000 to 2000.

ODrive Motor Controller Wiring:

Power and Motors:

The ODrive is powered from the battery via its DC Terminals, and each motor's three phases (A, B, C, Blue, Yellow and Green) are connected to one of its motor output terminals (M0 and M1). The ChIMP's righthand motor (his right hand) is connected to M1, his left connected to M0. The order in which the motor wires are connected does not matter, as the ODrive figures this out by itself during calibration.

Auxiliary resistor:

Since the ChIMP is running on batteries, generally no auxiliary braking resistor is required (the ODrive dumps recuperated energy back into the battery). However, depending on the type of battery you are using, the resistor might help avoid issues on the power bus that could lead to the ODrive unexpectedly shutting down.

Hall sensors:

Each hoverboard motor comes with internal Hall (magnetic) sensors that tell the ODrive when to power which of the three motor phases. Each Hall sensor set has five wires: Two for power (5V, usually red and GND, usually black) and three for signal (often blue, yellow and green). A motor's Hall sensors are connected using extension cables to the corresponding sensor inputs (M0 and M1) on the ODrive's J4 connector. The three signal lines are connected to inputs A, B and Z where the order again does not matter. You will probably need to make an extension cable (the shorter the better for the signal). Note that it is necessary to add a 22nF capacitor between each of the three signals lines and ground for signal integrity. Charleston Robotics has developed a neat little PC board that incorporates these capacitors along with a header and pads to solder your extension wires. Look for them shortly on our website!

Configuring the ODrive:

To configure the ODrive for the ChIMP you need to first install the ODrive tools software and setup the necessary drivers. There is a comprehensive explanation of how to do this for different operating systems [here](#).

The tools include the odrivetool interactive shell and a Python3 library.

Once you have installed all the tools following these [Instructions](#), this script takes care of setting up the ODrive. This will only work if the wiring of the motors and Hall sensors is correct.

All you need to do is:

1. Wire up the ODrive and motors as directed in this Construction Manual.
2. Make sure both motors are **ABSOLUTELY FREE TO TURN**. Secure the robot on a stand.
3. Connect the ODrive to a computer via USB.
4. Power up the ODrive with the ChIMPs battery.
5. Run this script [ODrive Setup](#) using a Python3 interpreter. On Linux this will look like so for motor M0:

```
./odrive_setup.py -a 0
```

And like so for motor M1:

```
./odrive_setup.py -a 1
```

You can use the interactive odrivetool to check if the motors are working as expected, or you can go right to the next step!

Remote control the motors via the Arduino:

Now that the ODrive is ready to use and the Arduino receives the RC signals, you can control the motors via the remote control.

Wiring:

The Odrive provides enough power from its 5V regulated output to power the Arduino and the receiver. You also need to connect the Arduino's serial interface to the ODrive - an overview over all necessary connections is shown on the ChIMP Wiring Diagram.

Code:

Once the ODrive, the motors, the Arduino and the receiver are wired up, you can upload [this test project](#) to the Arduino Mega.

Test:

MAKE SURE THE ROBOT IS PROPPED UP AND THE WHEELS CAN TURN FREELY!

Connect the battery to the ODrive. You should see the Arduino's LED flashing, indicating that the 5V power supply from the ODrive is working.

The motors should be in idle mode. Activate them by sliding/switching/turning the respective knob (Stick) on the RC transmitter. After that, check if they move when you move the sticks. Applying throttle should make them turn in the same direction, applying steering should make them turn in opposite directions. If that's not the case, adjust one of the MOTORDIR definitions in the config.h, recompile and reupload.

Control the motors by tilting the IMU:

At this stage, you can go ahead and compile and upload the complete HoverBot.ino file onto your robot. **For testing, again make sure that the wheels are free to turn!**

Once you turn on the robot, the blinking LED indicates that communication with the IMU is established and the system is ready to go. If the LED does not blink, something is wrong with the IMU.

In addition to reacting to the RC commands, once activated the wheels will now also react to the tilt of the vehicle. Tilting forward should make them turn forward, tilting backward should result in the wheels turning backward. Think of balancing a ruler in your palm: if it falls away from you, you must quickly move your hand forward to catch its fall...Rotating the robot around the vertical axis should make the wheels counter-rotate - this helps with driving straight.

Controller tuning:

The controller is based on five parameters:

- **KP_BALANCE:** Determines how much the robot reacts to leaning forward or backward. A high value might lead to tilt oscillations
- **KD_BALANCE:** Determines how much the robot reacts to the rate of tilt change. It generally dampens oscillations, but a value too high might lead to increased oscillations (mainly due to time lags in the control loop)
- **KP_POSITION:** Determines how strongly the robot reacts to RC throttle commands

- `KP_STEERING`: Determines how strongly the robot reacts to RC steering commands
- `KD_ORIENTATION`: Determines how much the robot tries to counteract turning around the vertical axis - higher values make going straight easier but turning slower

Tuning these parameters is not trivial and generally is a tedious, iterative process (test, change, compile, upload, repeat). However, the default parameters should be a very good start and if anything you should only need to change the first one (`KP_BALANCE`). If the robot does not seem to balance well, try a **(slightly!)** higher value. If it oscillates, try a **(slightly!)** smaller value.

Change the parameter in the `config.h` accordingly, recompile and reupload.

Annex 1. Installing Arduino Libraries from Github:

Installing Libraries:

Software libraries are an efficient and common way to encapsulate and distribute pieces of reusable code (such as sensor drivers) that are not expected to need to be changed by the user. A library typically consists of two files, a header file (`.h`) and a source file (`.cpp`). Some libraries consist only of a header file, some contain additional files or even subfolders.

Using a library in an Arduino project generally requires two steps:

- Placing all files (.h and .cpp) in a subfolder of your Arduino IDE's libraries folder. The libraries folder is created automatically upon installation of the IDE. The subfolder needs to have the same name as the library. This step is referred to as “manual installation” in the [official Arduino tutorial on the topic](#).
- At the top of your Arduino sketch, including the library with a precompiler instruction that looks something like `#include <myLibrary.h>`. This line of code alternatively gets created automatically when you click Sketch>Include Library>myLibrary as is shown at the very bottom of the [tutorial](#).

Getting the Code from GitHub:

Many open-source projects (such as our ChIMP) are hosted on GitHub, which is an online platform building on the powerful and widely used software version control tool Git. On any project main page, there is a green button captioned “Clone or download”.

If you are familiar with Git you can simply clone the entire project into the right folder, e.g. into a subfolder of the Arduino libraries folder in case of an Arduino library. If not, simply click on “Download ZIP” and, once downloaded, extract the archive into a subfolder of the libraries folder. **Alternatively, you can use the Arduino IDE's built-in functionality to include ZIP libraries, described [here](#).**

Note: You will need to restart your Arduino IDE after installing a library before you can use it in your code.