HardwareX 10 (2021) e00226

Contents lists available at ScienceDirect

HardwareX

journal homepage: www.elsevier.com/locate/ohx

The Crop Clamp – A non-destructive electromechanical pinch test to evaluate stalk lodging resistance

Andrew M. Stucker, Ethan Morris, Christopher J. Stubbs, Daniel J. Robertson [*]

AgMEQ Laboratory, Department of Mechanical Engineering, University of Idaho, 875 Perimeter Drive, Moscow, ID 83843, USA

ARTICLE INFO

Article history:
Received 7 April 2021
Received in revised form 17 August 2021
Accepted 18 August 2021

Keywords:
Lodging
Stalk
Stiffness
Transverse
Maize

ABSTRACT

Given the ever-increasing world population, maize plays a pivotal role in global food security. A major obstacle facing farmers is stalk lodging (the breakage of the stalk before harvest), which leads to substantial losses in annual yields. Weather, disease, and pest damage are major contributors to stalk lodging. Traditionally, evaluating a stalk's tendency to lodge was achieved with a 'pinch' test: pinching the stalk by hand to estimate its transverse stiffness. This test is inherently qualitative, and results vary from person to person. To combat these problems, a portable, battery-operated, non-destructive device for precisely measuring the transverse stiffness of maize stalks, known as the Crop Clamp, has been developed. The device is capable of recording over 100 measurements per hour and has been validated against laboratory tests.

Specifications table

Hardware name	The Crop Clamp
Subject area	• Mechanical Engineering • Agricultural Sciences • Open-Source Alternatives to Existing Infrastructure
Hardware type	• Measuring Physical Properties • Field Measurements and Sensors
Open-Source License	GNU General Public License V3
Cost of Hardware	$1,165.27
Source File Repository	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1

Hardware in context

Stalk lodging in maize (the breakage of the stalk before harvest) results in the loss of between 5% and 25% of global yields [1]. This creates instability in the global maize supply and results in substantial economic losses for farmers [2].

[*] Corresponding author.
E-mail address: danieljr@uidaho.edu (D.J. Robertson).

https://doi.org/10.1016/j.ohx.2021.e00226
2468-0672/© 2021 The Author(s). Published by Elsevier Ltd.

It is therefore desirable to breed for maize varieties that are more resistant to stalk lodging [3–5]. It has been well established that stalk lodging in maize is related to stalk structural features [2,3,6–14]. Obtaining accurate measurements of these structural features aids agronomists in selecting lodging-resistant maize varieties for breeding [3,15–18]. Plant scientist are particularly interested in measurements that can be taken in a field setting as measuring plants in or near the field significantly reduces costs associated with labor, transportation, and sample storage [19–23]. One of the structural features of interest in determining a maize stalk's susceptibility to lodging is the transverse stiffness of stalk [7,24]. Historically, transverse stiffness of maize stalks has been determined via the pinch test method: pinching a maize stalk by hand to estimate its stiffness [25]. While the pinch test has been accepted as valid in the agricultural community for years [26–29], it is based on manually estimating the stiffness, which leads to results varying from person to person. This has given rise to the need for a device capable of gathering accurate transverse stiffness measurements [3,7].

Currently, accurate transverse stiffness measurements can be performed on Maize using a Universal Testing System (UTS) [24]. A UTS can perform the necessary low-force compression procedures required to determine transverse stiffness, but they are expensive and cannot be deployed outside the laboratory (e.g., in a field setting) [30]. These restraints result in longer testing times and make it impractical to determine the transverse stiffness of plants in or near the field.

The lack of portable testing capabilities and the high cost of UTS machines led to the development of a portable, nondestructive device capable of accurate transverse stiffness measurements. This device, known as the Crop Clamp, combines the accuracy of UTS machines with portability, high testing speed, and low cost, thereby enabling plant scientist to collect transverse stiffness data in a field setting (using a portable table).

Hardware description

The Crop Clamp is a low-cost device for measuring the transverse stiffness of maize stalks. The device can be built for under $1200, compared to the $8700+ required for the traditional lab equipment required for transverse stiffness tests [30]. The Crop Clamp is portable and battery powered. When used on a tabletop the device can continuously test stalks for up to three hours. A single user can conduct approximately 100 test per hour on average with the Crop Clamp.

A summary of the device's key features is below:

- $1200 transverse stiffness measurement device
- 5-pound, portable device with a battery life of up to 3 h
- 100 + stiffness measurements per hour.
- Real-time, customizable transverse stiffness calculations with data plotting.
- Data resolution of 0.001 mm and 0.01 N.

Design files

Design files summary

Table 1

Fabrication summary

Table 2

CropClamp_OS.ino file is the operating software for the Crop Clamp. It can be opened and uploaded to the device using the Arduino IDE.

The Crop_Clamp_Buttons.ino file is used to configure the Crop Clamp's D-Pad before installing the actual operating software.

The Crop_Clamp_Full.png is the full electronics schematic for the Crop Clamp.

Bill of materials

Tables 3–6

Due to the complexity of the Bill of Materials, a full document with extra notes and purchase links has been included in the project repository. A simplified version of the Bill of Materials is included here.

Table 1
Design files with locations and file types.

Design file name	File type	Open source license	Location of the file
Electronics Housing	CAD	GNU GPL v3	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1
Battery Housing	CAD	GNU GPL v3	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1
Screen Housing	CAD	GNU GPL v3	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1
Screen Border	CAD	GNU GPL v3	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1
Electronics Cover	CAD	GNU GPL v3	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1
Wire Track	CAD	GNU GPL v3	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1
Load Cell Carriage	CAD	GNU GPL v3	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1
Battery Cover	CAD	GNU GPL v3	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1
D-Pad Housing	CAD	GNU GPL v3	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1
Button Cover	CAD	GNU GPL v3	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1
Calibration Piece	CAD	GNU GPL v3	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1
Load Cell Backplate	CAD	GNU GPL v3	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1
Metal Backer	CAD	GNU GPL v3	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1
CropClamp_OS	.ino	GNU GPL v3	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1
Crop_Clamp_Buttons	.ino	GNU GPL v3	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1
Crop_Clamp_Full	png	GNU GPL v3	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1
BOM Nice	.xlsx	GNU GPL v3	Laboratory, AgMeq (2021), "Crop Clamp Design Files", Mendeley Data, V1, https://doi.org/10.17632/p4r3khtht7.1

Table 2
Fabrication instructions for design files.

Design file name	Manufacturing Method	Material
Electronics Housing	3D Printing	ABS – 20% Infill
Battery Housing	3D Printing	ABS – 20% Infill
Screen Housing	3D Printing	ABS – 20% Infill
Screen Border	3D Printing	ABS – 20% Infill
Electronics Cover	3D Printing	ABS – 20% Infill
Wire Track	3D Printing	ABS – 20% Infill
Load Cell Carriage	3D Printing	ABS – 20% Infill
Battery Cover	3D Printing	ABS w/ Carbon Fiber Inlay (otherwise 40% ABS)
D-Pad Housing	3D Printing	ABS – 20% Infill
Button Cover	3D Printing	ABS – 20% Infill
Calibration Piece	3D Printing	ABS – 20% Infill
Load Cell Backplate	Milling	3/16″ Stock Aluminum
Metal Backer	Milling	3/16″ Stock Aluminum

Table 3
3D Printed Parts.

Part Name	Designator	Qty	Cost*	Material
Electronics Housing	CAD01	1	$3.39	ABS
Battery Housing	CAD02	1	$4.17	ABS
Screen Housing	CAD03	1	$3.26	ABS
Screen Border	CAD04	1	$0.16	ABS
Electronics Cover	CAD05	1	$2.96	ABS
Wire Track	CAD06	1	$1.75	ABS
Load Cell Carriage	CAD07	1	$4.18	ABS
Battery Cover	CAD08	1	$2.95	ABS
D-Pad Housing	CAD09	1	$1.97	ABS
Button Cover	CAD10	1	$0.17	ABS
Calibration Piece	CAD11	1	$0.04	ABS

*The cost of these parts was estimated using the mass of the part and the cost of ABS ($20/kg).

Table 4
Milled Parts.

Part Name	Designator	Qty	Cost*	Material
Load Cell Backplate	MILL01	1	$0.12	Aluminum
Metal Backplate	MILL02	1	$0.25	Aluminum

*The cost of these parts was estimated using the mass of the part and the approximate cost of Aluminum stock ($10/kg).

Table 5
Electronic Parts.

Part Name	Designator	Qty.	Cost
Arduino	Arduino Mega	1	$40.30
Stepper Motor	NEMA 17 Stepper Motor	1	$10.99
Button	microtivity Tact Switch	1	$5.29
Prototype Board	Elegoo Prototype Board Kit	1	$9.99
Breadboard	400 Pin Breadboard	1	$8.99
12 V Battery	REV Robotics Slim Battery	1	$50
Charger	REV Robotics Battery Charger	1	$32
12 V Connectors	XT30 Connectors	1	$8.99
Screen	Adafruit Product 1770	1	$29.95
microSD Card Reader	Adafruit Product 254	1	$7.50
Load Cell Amplifier	Sparkfun HX711	1	$9.95
Stepper Motor Driver	Pololu MP6500 Stepper Motor Driver	1	$5.95
Analog to Digital Converter	Adafruit Product 1085	1	$14.95
Voltage Regulator	eBoot Mini Adjustable Power Supply	1	$9.25
Linear Potentiometer	LMC-100 Linear Potentiometer 10kOhm +- 20% with + -0.7% linearity 100 mm stroke	1	$110
Load Cell	FUTEK Miniature S-Beam Jr. Load Cell 2.0 (25 lb capacity) with wire	1	$700*
Jumper Wire Kit	Jumper Wire Kit	1	$10.99
Electrical Tape	Electrical Tape	1	$3.98
Wire Nuts	Wires Nut Set	1	$5.50
Wire	Extra Wire	1	$12.99

*The cost of the FUTEK Load Cell is approximate because the price of the load cell is not available without a quote request.

Table 6
Hardware.

Part	Designator	Qty	Price/Unit	Total Price
8020 Beam	16″ 80–20 Beam (part # 25–5013)	1	$2.21	$2.21
Lead Screw System	Lead Screw Nut/ Shaft Coupler/ Lead Screw	1	$11.98	$11.98
Single T-Nut	8020 Single T Nut (8020 Part Number 3859)	2	$0.27	$0.54
Double T-Nut	8020 Double T Nut (8020 Part Number 3800)	4	$0.53	$2.12
M6 × 1.0 10 mm Bolt	M6 × 1.0 10 mm Bolt (8020 Part Number 13–6310)	10	$0.20	$2.00
M10 Flat Washer Pack	M10 Flat Washer (Should be ~ 1.66 mm thick)	1	$7.64	$7.64
Metric Bolt Set*	NINDEJIN 880 Pcs M2 M3 M4 M5 Assortment Kit	1	$25.89	$25.89

*The Metric Bolt Set included in Table 6 is the best way to get all the assorted nuts and bolts required for the Crop Clamp. However, a list of individual nuts and bolts can be found in the Bill of Materials in the project repository.

Build instructions

First, begin by manufacturing the 3D-printed parts and the milled parts as specified in Section 3: Design Files. These parts must be completed prior to device assembly.

The remainder of the build instructions are summarized in the Crop_Clamp_Full.png electronics schematic found in the project repository. It may be helpful to reference the full electronic schematic throughout the build process to ensure that no wiring mistakes have been made.

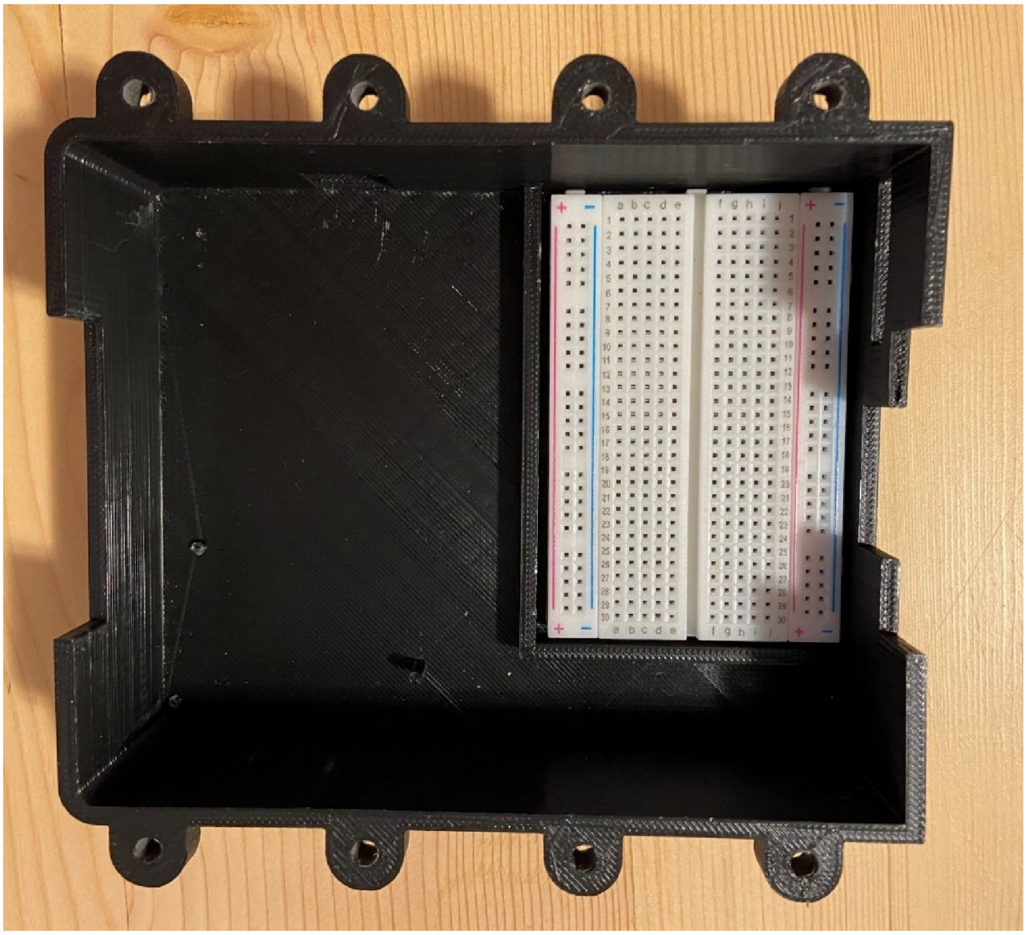


Fig. 1. Breadboard placed within Electronics Housing.

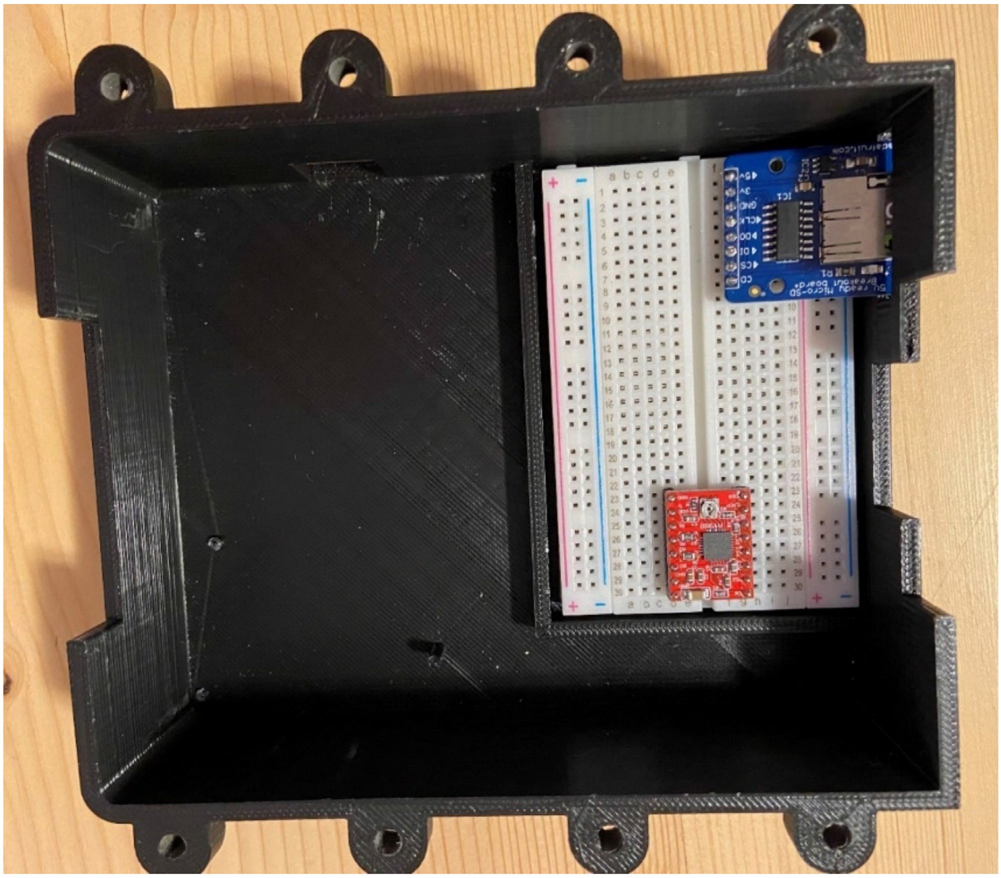


Fig. 2. microSD Card Reader and Stepper Motor Driver are in place.

Part A: Basic Electronics

When the custom parts have been manufactured, the first step in assembling the Crop Clamp is to use the double-sided tape on the back of the Breadboard to connect the Breadboard to the Electronics Housing (CAD01) as seen in Fig. 1.

Next, place the microSD Card Reader and the Stepper Motor Driver into place on the Breadboard as seen in Fig. 2. Note that the microSD Card Reader will be angled upwards slightly and stick out of the Electronics Housing through the rectangular slot when installed.

Next, place the power and ground jumper wires for both the microSD Card Reader and the Stepper Motor Driver as seen in Fig. 3. The left side of the Breadboard will provide power for all the breakout boards in the Crop Clamp. Notice that the jumper wire powering the microSD Card Reader is bent out of the way to make room for future wires. Also add 9 jumper wires to the board for the Load Cell Amplifier. The amplifier will be placed on top of these wires later. The location of these wires can be seen in Fig. 3.

Add the power lines for the Load Cell Amplifier (there are two 5V+ lines, the ground line will be added later) and add a small jumper wire connecting SLP and RST on the Stepper Motor Driver. Also connect the positive and ground lines from the right side of the Breadboard to the stepper motor driver's VMOT and GND pins, respectively. The added wires can be seen in Fig. 4.

Add seven 10 K resistors to the board (it will be necessary to trim both ends of the resistor wires so that the resistor rests on the Breadboard). These will act as step-down resistors for the D-Pad ground lines. Next, place the Load Cell Amplifier in its correct spot so that its pins occupy rows 12–16 on the Breadboard. See these changes in Fig. 5.

Add the Arduino Mega and begin connecting it to the Breadboard. It will slide into place on the left side of the Electronics Housing. The USB Port of the Arduino Mega will fit into the hole in the Electronics Housing. It is possible that one or two of the 3D printed pegs on the bottom of the electronics cover will snap off (or need to be snapped off) to get the Mega into place properly. The seated Arduino Mega can be seen in Fig. 6. Now connect the bottom ground on the left side of the Arduino Mega to the second column on the Breadboard (the blue column). Also connect the top 5 V+ pin on the bottom of the Arduino Mega to the first column of the Breadboard (the red column). See all changes in Fig. 6.

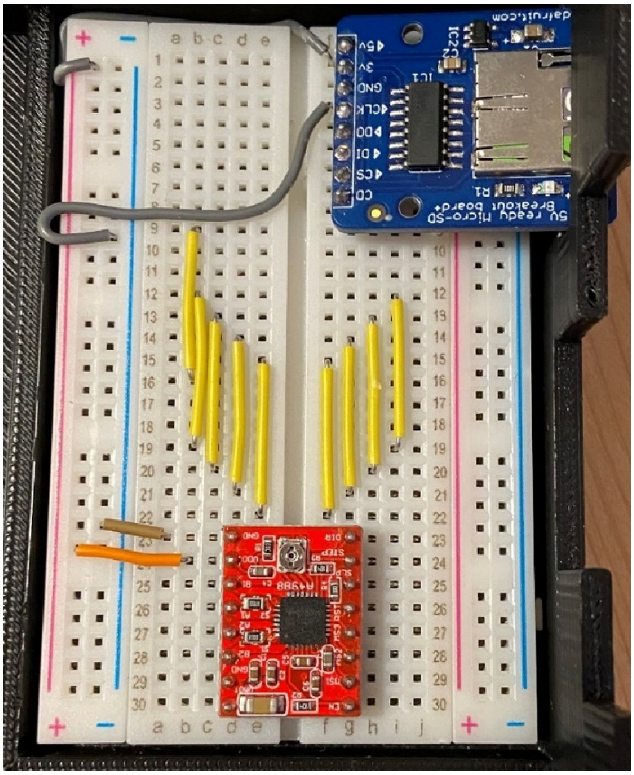


Fig. 3. The first group of wires has been added.

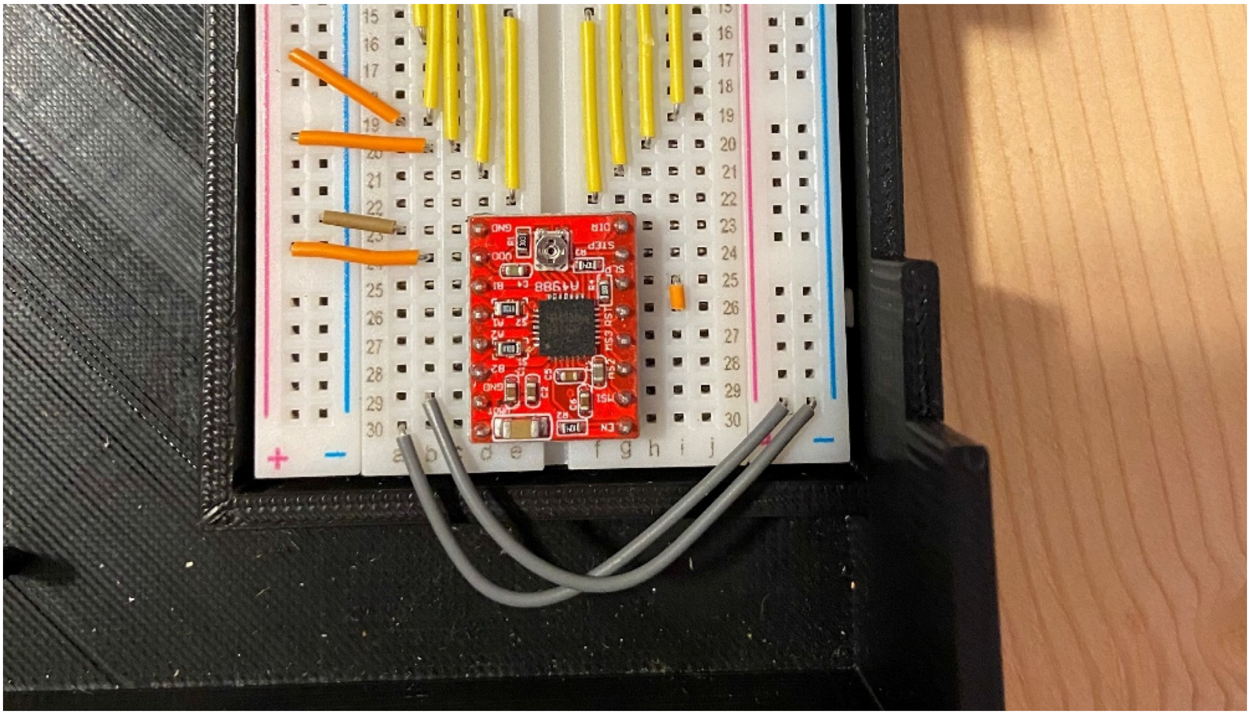


Fig. 4. Wires for the Stepper Motor Driver and the Load Cell Amplifier has been added.

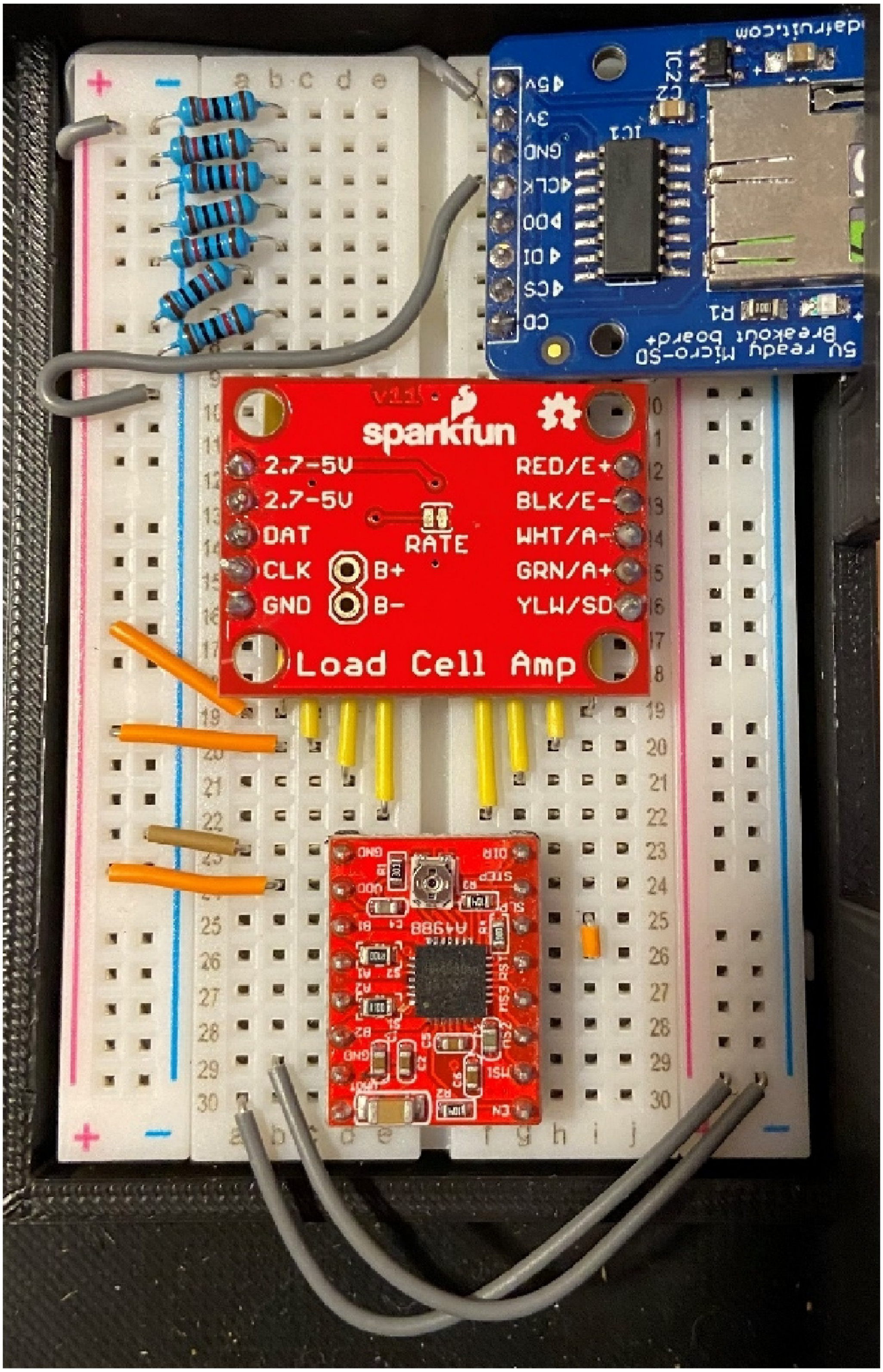


Fig. 5. Seven 10 K resistors have been added, as well as the Load Cell Amplifier.

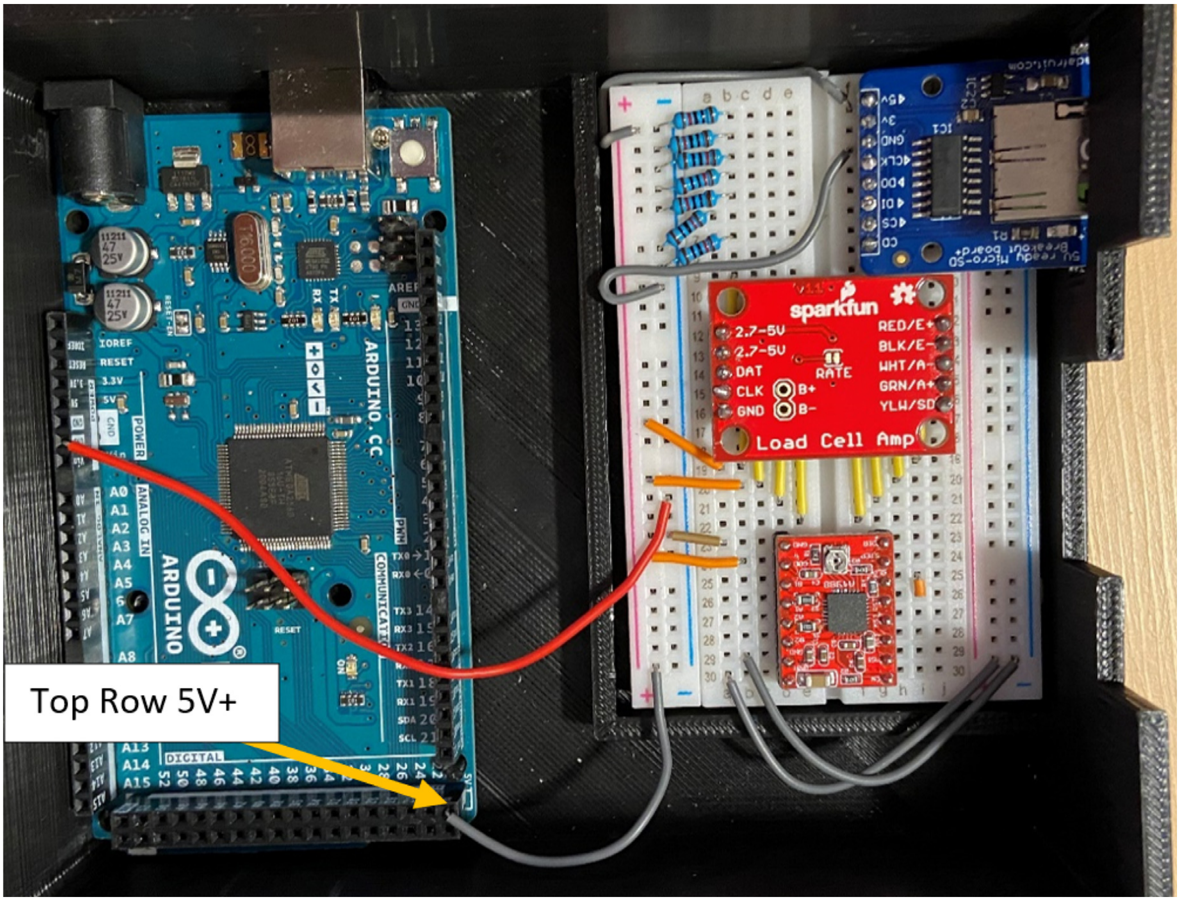

Top Row 5V+

Fig. 6. Arduino Mega added with 5 V+ and GND lines attached to Breadboard.

Next, connect pin 6 on the Arduino Mega to the DIR pin (Breadboard row 23) of the Stepper Motor Driver. Connect pin 5 on the Arduino Mega to the STEP pin (Breadboard row 24) of the Stepper Motor Driver. Also, add a jumper wire connecting the Load Cell Amplifier (Breadboard row 9) to the GND pin at the top right of the Arduino Mega. See these additions in Fig. 7.

Next, add the power and ground lines for the Arduino Mega itself. The two rightmost columns of the Breadboard (red and blue) will be powered by 12 V, and this is where the Arduino will get its power. Connect the Arduino's Vin pin to the positive column on the right of the Breadboard and the uppermost GND line on the left of the Arduino Mega to the rightmost Breadboard column. See the added wires in Fig. 8.

Add the signal lines for the microSD Card Reader. Connect Arduino Pin 50 to microSD Card Reader DO pin, Arduino Pin 51 to microSD Card Reader DI pin, and Arduino Pin 52 to microSD Card Reader CLK pin. If there are not any jumper wires long enough to reach between the pins, it may be necessary to solder two smaller jumper wires together as seen in Fig. 9.

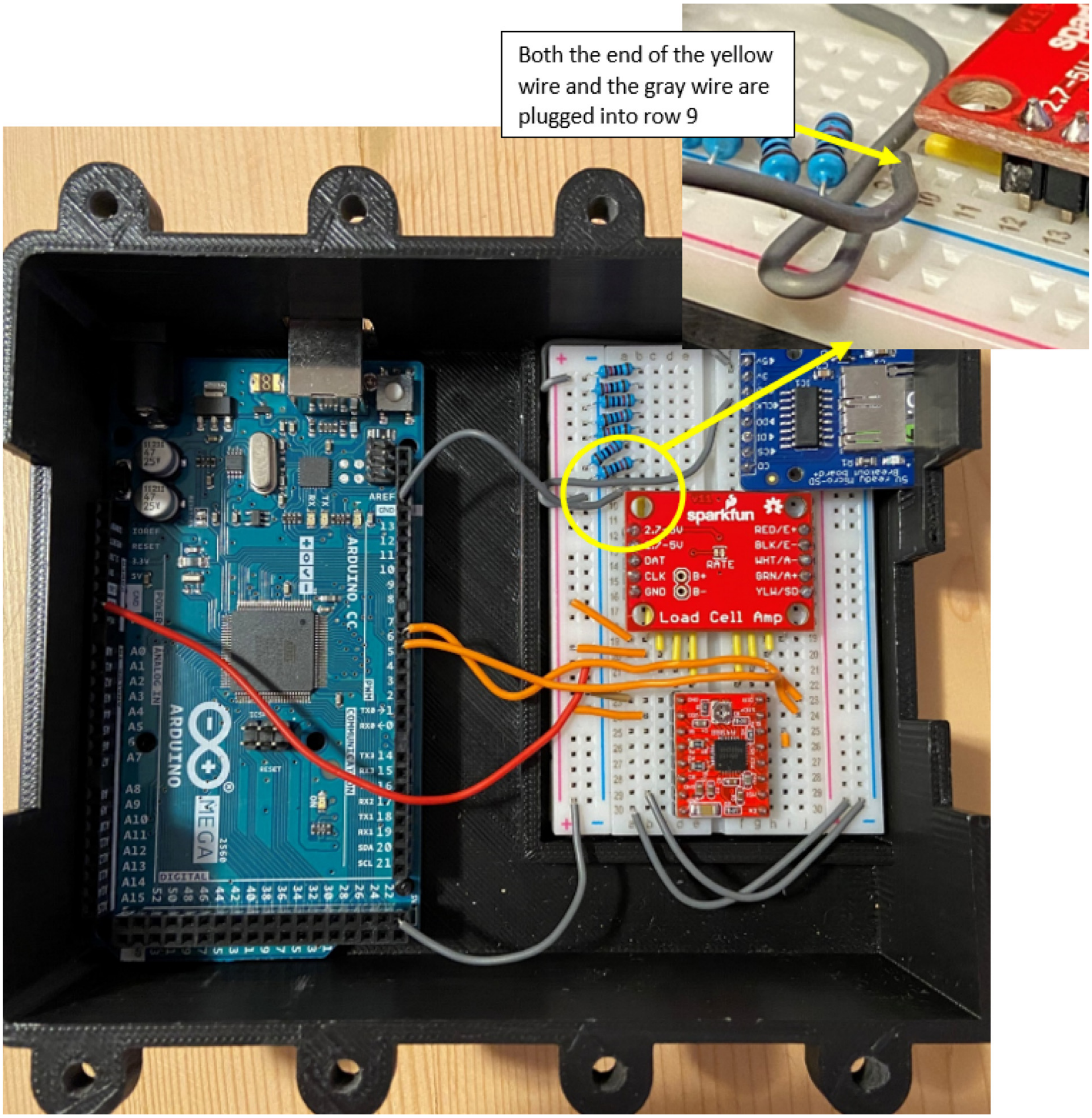
Next, add the jumper wires running between the seven 10 K Ohm resistors and their corresponding Arduino pins. Connect Arduino pins 10–13 to the top four resistors as shown in Fig. 10. These wires will fit better if they are placed below the signal lines going to the microSD Card Reader.

Next, add a jumper wire running between Arduino pin 4 and the microSD Card Reader CS pin as seen in Fig. 11.

Next, the CD (chip detect) pin of the microSD Card Reader must be connected to Arduino pin 3. However, a 10 K Ohm pull-up resistor must also be placed on this line. To accomplish this, it is necessary to create the custom wire seen in Fig. 12.

Place this wire, or a similar setup, as seen in Fig. 13. The single wire (in this case, the grey wire) must be plugged into the 5 V+ row of the Breadboard, while the two other wires (red in this case) must be connected to Arduino pin 3 and the CD pin of the microSD Card Reader.

Take 3 jumper wires and connect the remaining 10 K resistors to Arduino pins 14–16. Take two more jumper wires and connect Arduino pin 9 to the DAT pin of the Load Cell Amplifier and Arduino pin 8 to the CLK pin of the Load Cell Amplifier. These changes can be seen in Fig. 14.



Both the end of the yellow wire and the gray wire are plugged into row 9

Fig. 7. Added DIR and STEP lines for the Stepper Motor Driver and GND line for Load Cell Amplifier.

Part B: D-Pad Assembly

Before the rest of the Crop Clamp can be assembled, the custom D-Pad controller must be created using 7 Buttons and 1 Prototype Board. This part is complex, and it will take a substantial amount of time to create. This D-Pad will be used to control the Crop Clamp's Graphical User Interface (GUI). To begin, place the 7 Buttons in the correct location on the prototype board. If these Buttons are not positioned properly, the 3D printed Button Cover (CAD10) will not fit properly. See the Button layout in Figs. 15 and 16.

Note that the pins for each Button straddles two holes. That is, the right pins of the Button sit three columns over from the left pins. When the Buttons are positioned correctly, bend the end of the pins back slightly to hold the buttons in place. The board is now ready for soldering.

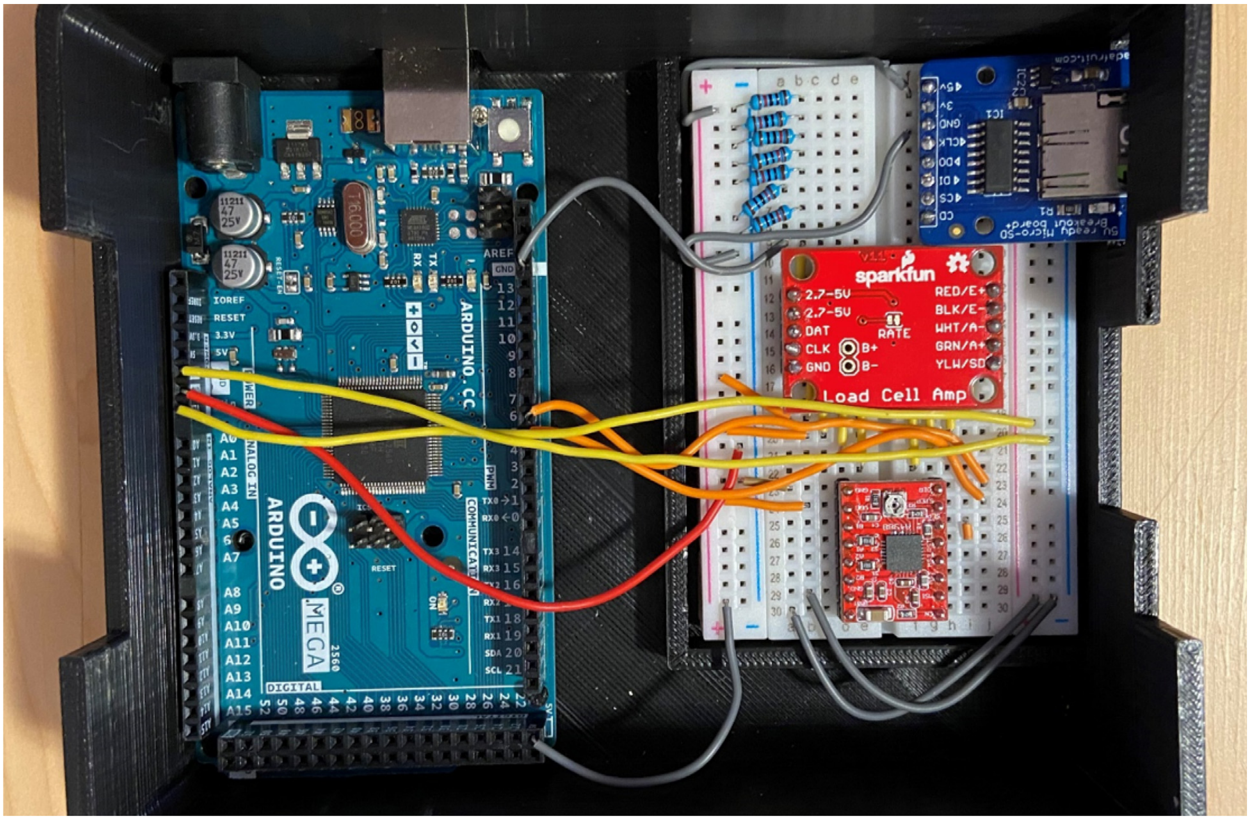


Fig. 8. Added the power and ground lines for the Arduino Mega.

One side of each Button will receive a 5 V+ signal from the Arduino, while the opposite side of the Button will have a unique GND pin running to the Arduino. For simplicity, all the Buttons should share a common 5 V+ line but have individual GND lines. To aid understanding, Fig. 17 shows how the Buttons are wired.

Fig. 18 shows the back of the D-Pad assembly with annotations indicating which Buttons the wires connect to. It is not necessary to exactly replicate this layout if all Buttons are positioned correctly. All seven buttons share a power line, but they all have unique ground lines. The ground wires should be roughly 24 in. long.

Fig. 19 is a complete electrical schematic of the D-Pad assembly included for reference.

Part C

Now that the D-Pad Assembly has been created, the bulk of the Crop Clamp can be assembled. Begin by taking the Battery and cutting off the connector and fuse. *Do not cut both wires at once, as this may short-circuit the battery*. Then solder a male XT60 connector to the remaining wires coming from the Battery. The XT60 connector has defined positive and negative leads; be sure to solder red to positive and black to negative. Place the Battery into the Battery Housing (CAD02) as shown in Fig. 20.

Next, place the custom D-Pad electronics board into the D-Pad Housing as in Fig. 21. The D-Pad assembly will fit snugly into the D-Pad Housing (CAD09). The D-Pad wires will have to be fed through the opening in the bottom of the D-Pad Housing.

Secure the Button Cover (CAD10) over the D-Pad assembly using two 3 M × 0.5 8 mm bolts and two 3 M × 0.5 nuts as shown in Fig. 22. There is a nut and bolt through the top and bottom of the Button Cover.

Next, take the Battery Cover (CAD08) and superglue the Metal Backplate (MILL02) to the flat square area. See Fig. 23 for reference.

Next, connect two 18″ wires to a female XT60 connector and secure everything with heat-shrink tubing or electrical tape. It should look something like the wire in Fig. 24. These wires will be referred to as the XT60 Connector, and it will be helpful to mark these two wires with tape so that they remain easy to identify as additional wires are added later.

Take the female end of the XT60 Connector and place it in the rectangular opening of the Battery Housing. Now position the wires from the D-Pad assembly so that they pass through the large rectangular slot in the left side of the Battery Housing.

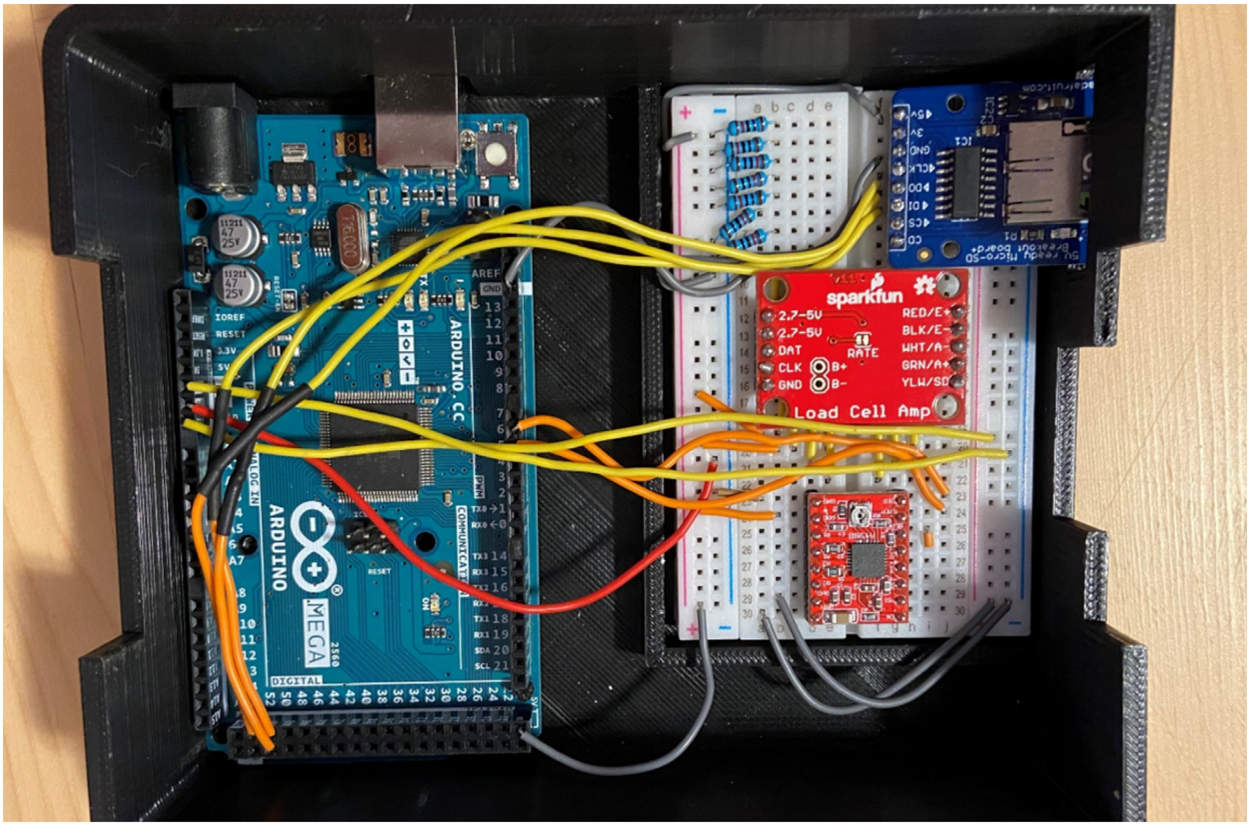


Fig. 9. Added lines between the microSD Card Reader's DO, DI, and CLK pins and the Arduino Mega.

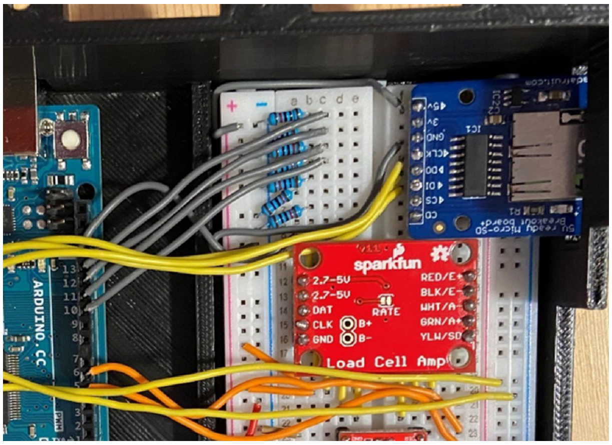


Fig. 10. Arduino Mega pins 10–13 connected to the 10 K resistors.

The XT60 Connector wires will run alongside the D-Pad wires, so it is important to mark the XT60 Connector wires with tape (or something similar) so they can be differentiated later. Refer to Fig. 25 for both changes.

Now place the 8020 Beam across the Battery Housing. The 8020 Beam should be flush with the right side of the Battery Housing and the D-Pad assembly should be positioned as seen in Fig. 26. The XT60 Connector and the wires from the D-Pad should all be underneath the 8020 Beam and must not be pinched between the beam and the Battery Housing.

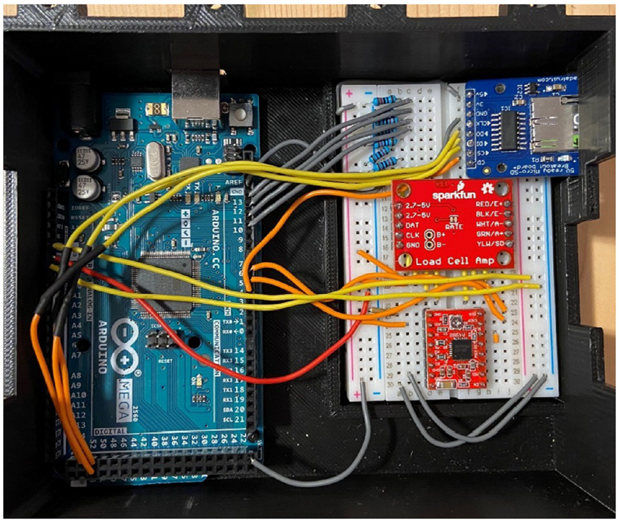


Fig. 11. A wire has been added running between Arduino Mega pin 4 and the microSD Card Reader CD.

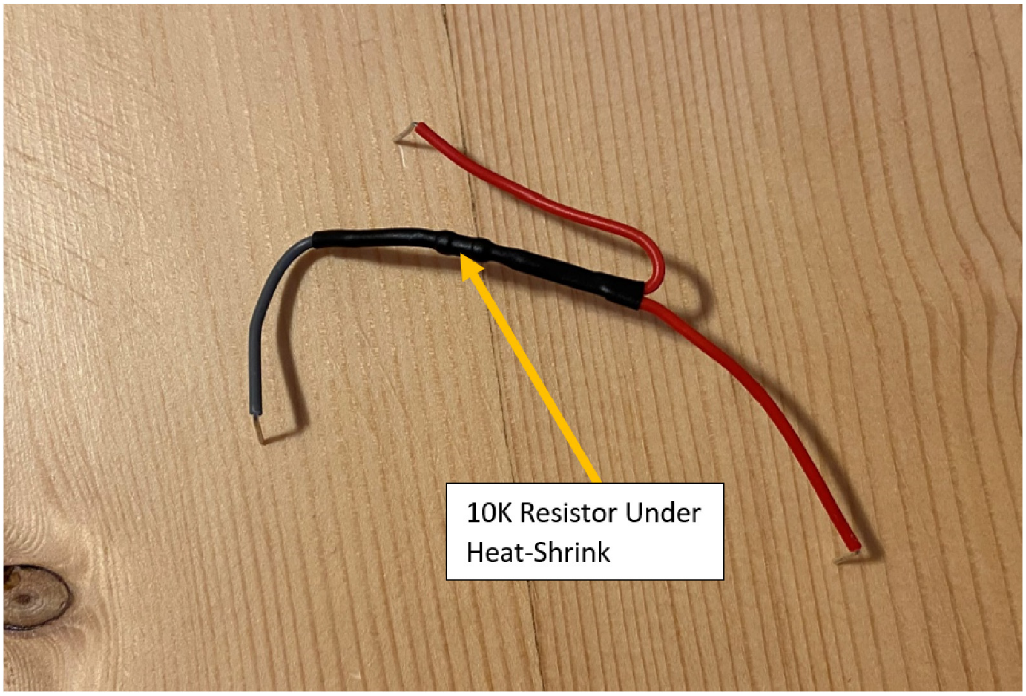


Fig. 12. A three-way wire with a built-in 10 K Ohm Resistor.

Now take two Double T-Nuts and place them into the grooves in the 8020 Beam as shown in Fig. 27. These nuts will be used to attach the Battery Cover to the 8020 Beam.

Use four M6 × 1.0 10 mm bolts to connect the Battery Cover to the Double T-Nuts in the 8020 Beam as seen in Fig. 28. The D-Pad wires should easily fit through the groove in the Battery Cover.

The Battery Cover must now be bolted to the Battery Housing using five M4 × 0.7 20 mm bolts as seen in Fig. 29. Keep in mind that the Battery Cover has already been attached to the 8020 Beam in the last step.

Next, connect the D-Pad to the Battery Cover in 5 places. Every exposed hole on the D-Pad Assembly should have a bolt. See Fig. 30 for reference. Be careful not to overtighten the two bolts in the bottom of Fig. 30.

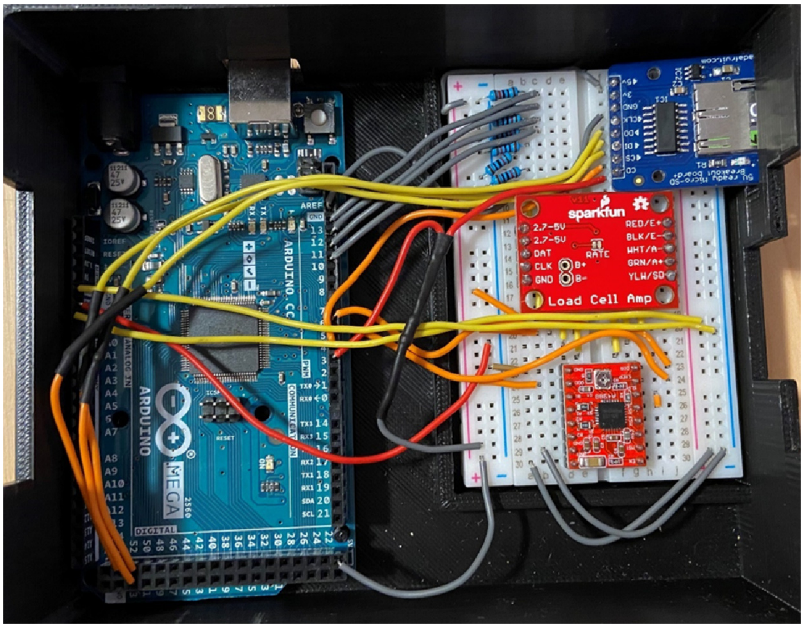
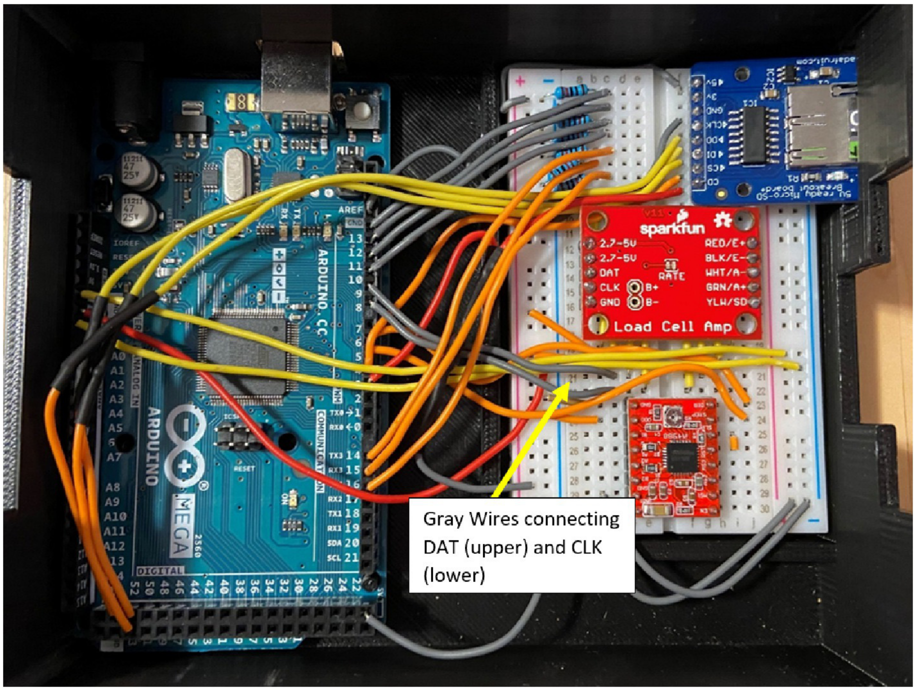


Fig. 13. The custom 3-way wire has been attached to the Breadboard 5 V+ column, pin 3 of the Arduino, and the CD pin of the microSD Card Reader.



Gray Wires connecting DAT (upper) and CLK (lower)

Fig. 14. The last three resistors have been connected to the Arduino, as have the DAT and CLK lines of the Load Cell Amplifier.

Part D: Load Cell Carriage

Before the wiring can be completed, the Load Cell Carriage (CAD07) must be added to the middle of the Crop Clamp. Begin by placing the Lead Screw Nut into the Load Cell Carriage and securing it with four M3 × 0.5 16 mm bolts as seen in Fig. 31. If the Lead Screw Nut will not fit, using a drill bit to slightly widen the hole in the Load Cell Carriage.

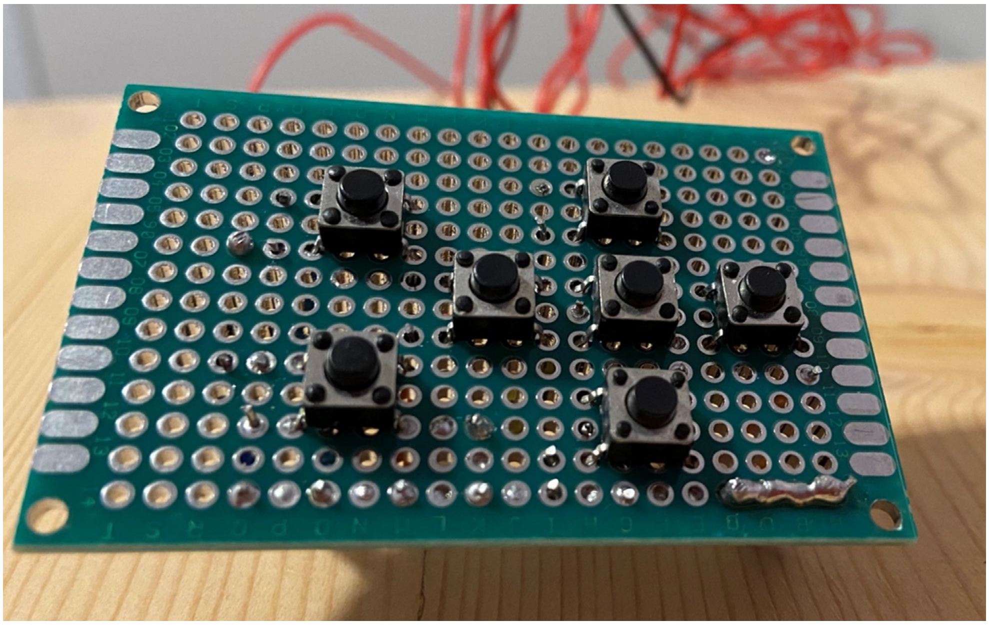


Fig. 15. The Buttons placed onto the prototype board.

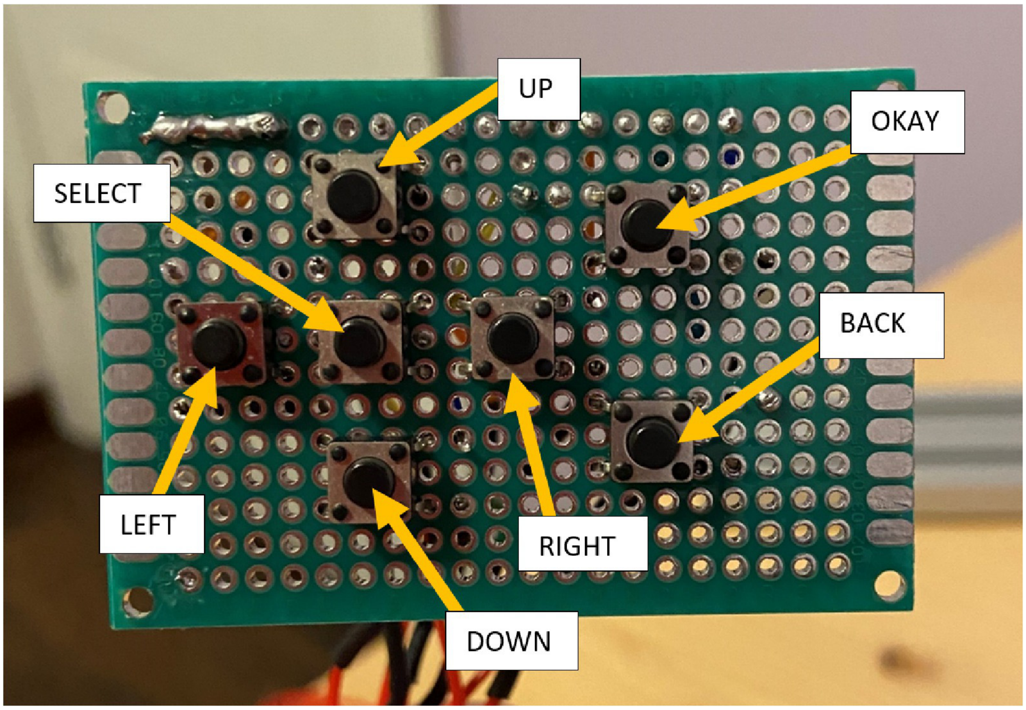


Fig. 16. Another view of the 7 Buttons with the naming conventions for the various Buttons. This is rotated 180 degrees from Fig. 15.

Now attach the Load Cell to the Load Cell Backplate (MILL02) using one M3 × 0.5 8 mm bolt. Note that the small gap in the bottom of the Load Cell must face outwards. See Fig. 32 for reference.

Next, place the Metal Backplate into position on the Load Cell Carriage as shown below. It might take some force to get the plate into position because of the bolt securing the Load Cell to the Load Cell Backplate. Next, use two M3 × 0.5 12 mm bolts to connect the Load Cell Backplate to the Load Cell Carriage as seen in Fig. 33.

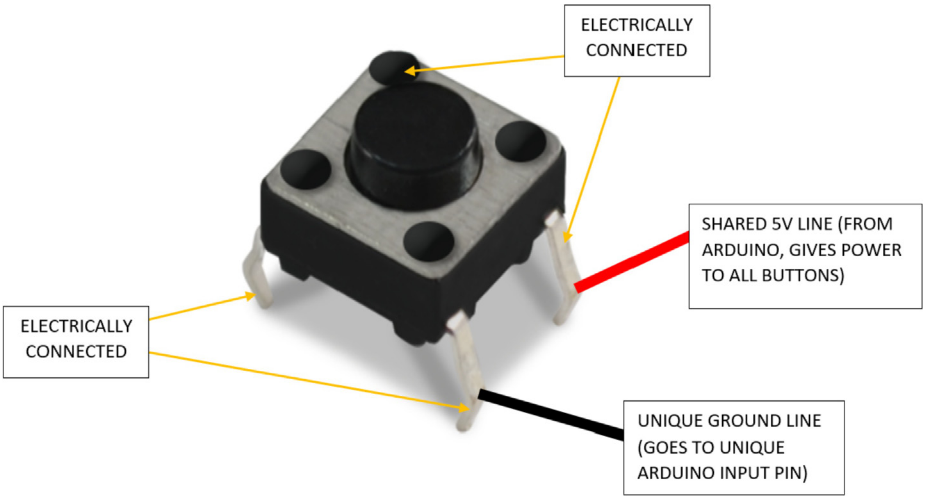

ELECTRICALLY CONNECTED
SHARED 5V LINE (FROM ARDUINO, GIVES POWER TO ALL BUTTONS)
ELECTRICALLY CONNECTED
UNIQUE GROUND LINE (GOES TO UNIQUE ARDUINO INPUT PIN)

Fig. 17. Button electrical layout.

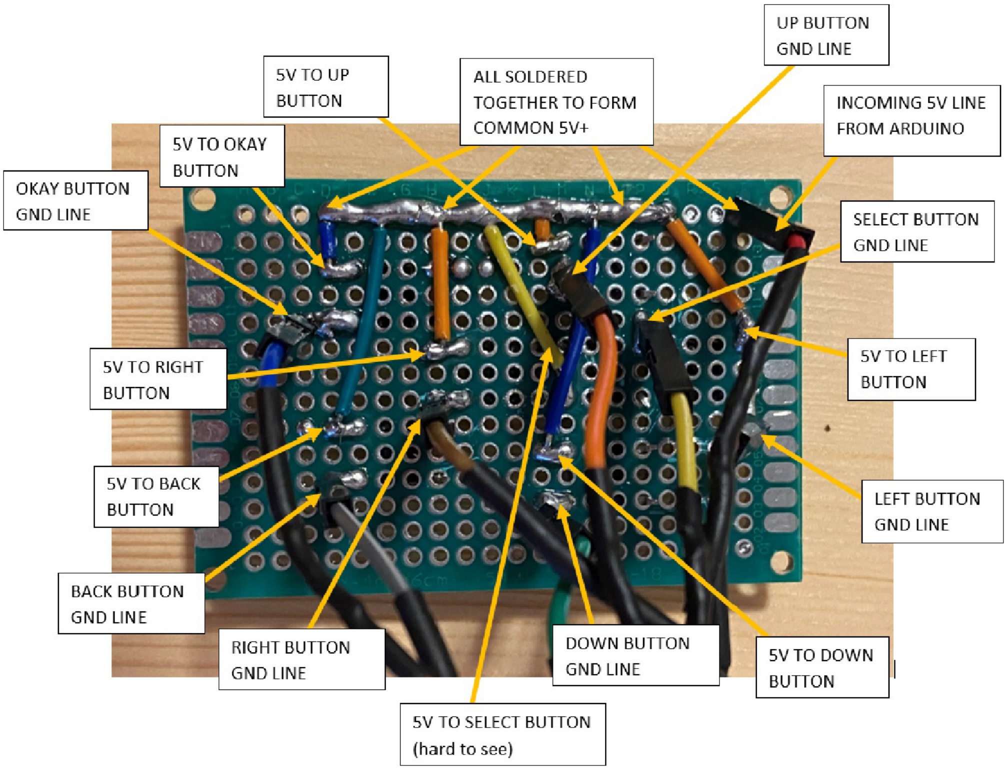


Fig. 18. Back of the D-Pad Assembly with annotations.

Slide the Load Cell Carriage onto the 8020 Beam. The plastic grooves on the Load Cell Carriage may have to be sanded before the part will slide onto the 8020 Beam smoothly depending on the quality of the 3D print. The Load Cell should face towards the Battery Cover. See Fig. 34 for reference.

Part E: Screen Wiring and Assembly

To begin the screen assembly, take 15 male-to-female jumper wires (8″-12″) and attaching them to the Screen as seen in Fig. 35.

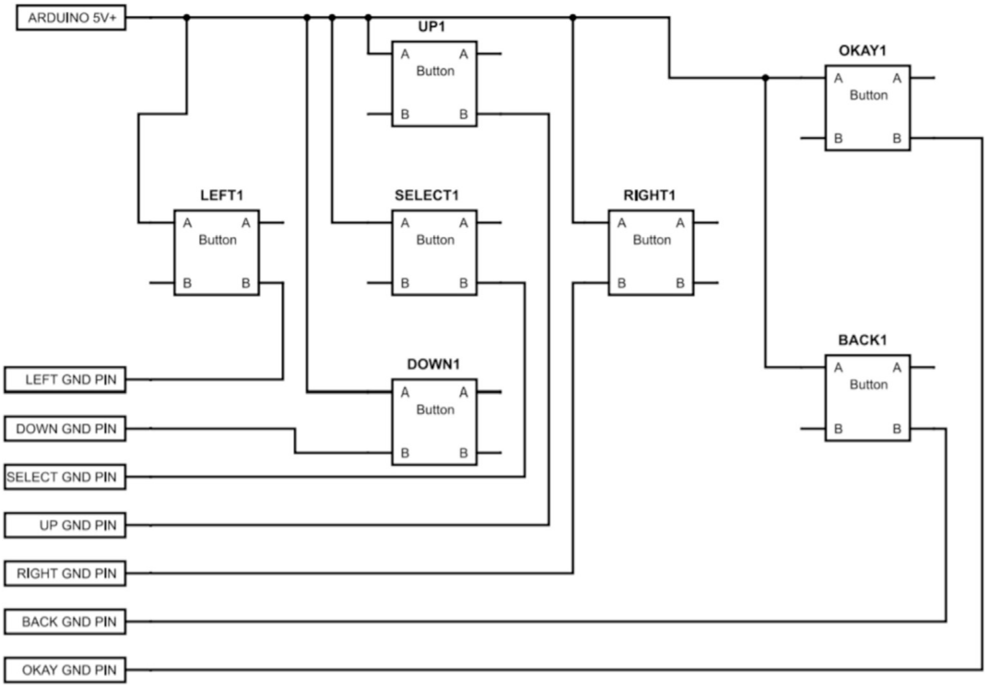


Fig. 19. Electrical schematic of the D-Pad assembly.

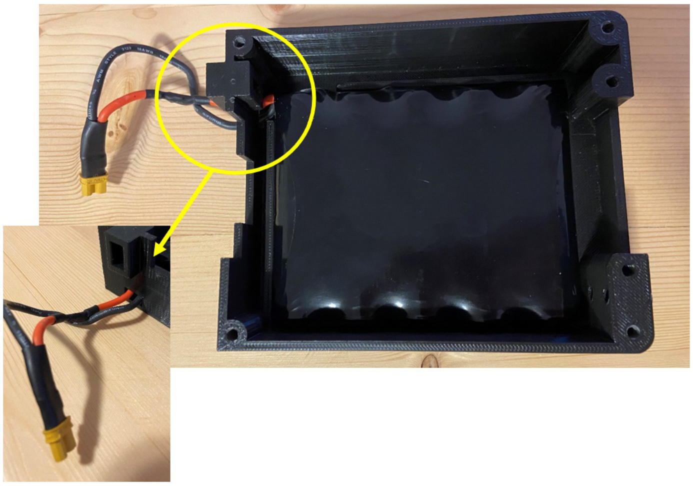


Fig. 20. The Battery placed into the Battery Housing. The male XT60 connector should come out of the opening in the bottom of the housing.

It is convenient to zip tie the RST and 3–5 V wires together in one bundle, and all the other wires together in another bundle. Connect the RST wire directly to the RESET pin of the Arduino Mega and the 3–5 V wire directly to the Arduino Mega's 3.3 V+ pin. See Fig. 36 for reference.

Now the remaining screen control pins can be connected. Connect screen pin CS to Arduino pin 32, C/D to Arduino pin 33, WR to Arduino pin 34, and RD to Arduino pin 35. Screen pins D0-D7 plug into Arduino pins 22–29, respectively. It may be helpful to reference the Crop Clamp's electronic schematic when completing the screen wiring.

Next, secure the Screen into place using the Screen Border (CAD04) and four M3 × 0.5 12 mm bolts. If there are no pins soldered on the right side of the screen, the right side of the Screen Border can be pushed farther into the Screen Housing than the left. To solve this problem, take a thin strip of velcro (or similar material) and place it on the right side of the Screen Border. If there are soldered pins on both sides of the screen, ignore the velcro suggestion. See both the velcro and the final assembly in Fig. 37.

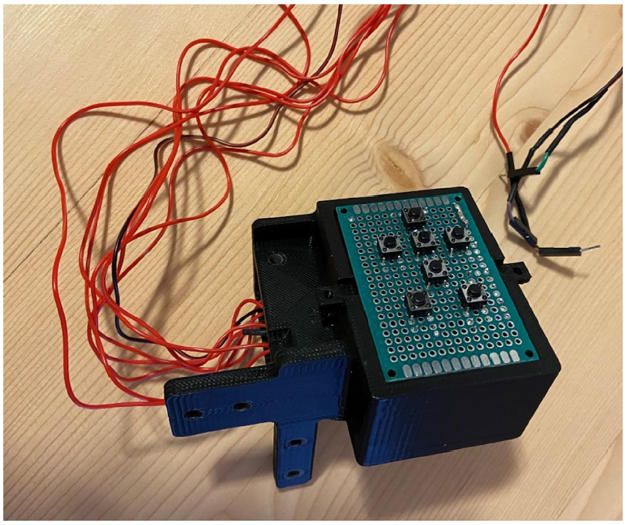


Fig. 21. The D-Pad protoboard has been placed into the D-Pad Housing.

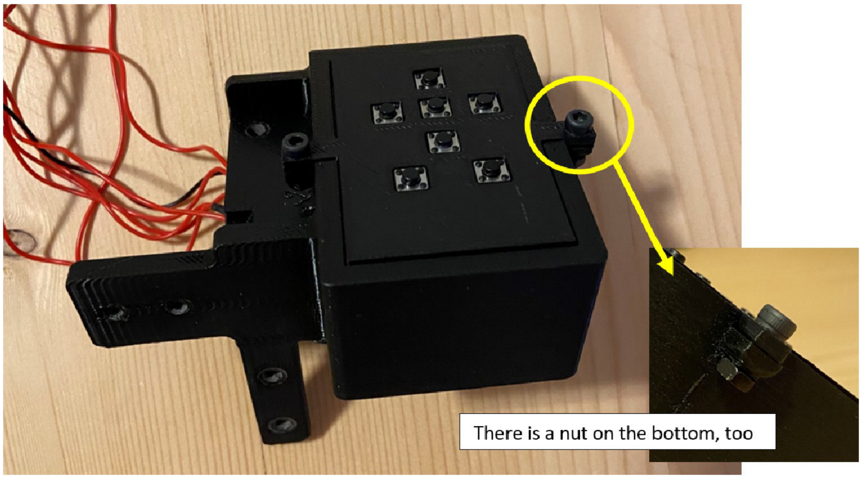


Fig. 22. The Button Cover attached to the D-Pad Housing.

Part F: Potentiometer Assembly

Next, prepare the potentiometer assembly. Start by soldering positive, negative, and signal wires to the Linear Potentiometer as seen below. Put a female pin on the signal line, but leave the positive and negative lines as bare wires. Also take a 9 V Battery Connection and solder 10″ wires onto it. Lastly, take a Voltage Regulator and solder wires to it. Use a female connector for both the OUT+ and OUT− locations and bare wires for the IN+ and IN− lines. All wires should be 4–5″ long. See all these components in Fig. 38.

Now take the Linear Potentiometer and pull the wires down through the top of the Electronics Cover (CAD05). Taking the ADS1115 Analog-to-Digital Converter and the Voltage Regulator, connect the signal line of the Linear Potentiometer to the A0 pin of the ADS1115, connect the OUT+ and OUT− lines of the Voltage Regulator to the ADS1115′s VDD and OUT- pins, respectively. See Fig. 39 for reference.

Now take the positive line from the 9 V connector, the IN+ line from the Voltage Regulator, and the positive line of the Linear Potentiometer and connect them using a small wire nut as seen in Fig. 40.

Now take a 6″ jumper wire with a male connector pin and strip the opposite end as seen in Fig. 41.

Take the bare end of the jumper wire from Fig. 41 and connect it to the negative line of the 9 V Battery Connector, the negative line of the Linear Potentiometer, and the IN- line from the Voltage Regulator with a small wire nut as seen in Fig. 42.

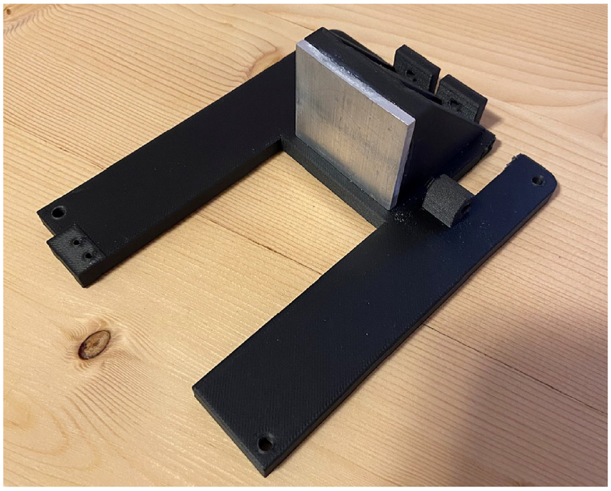


Fig. 23. The Metal Backplate has been attached to the Battery Cover.

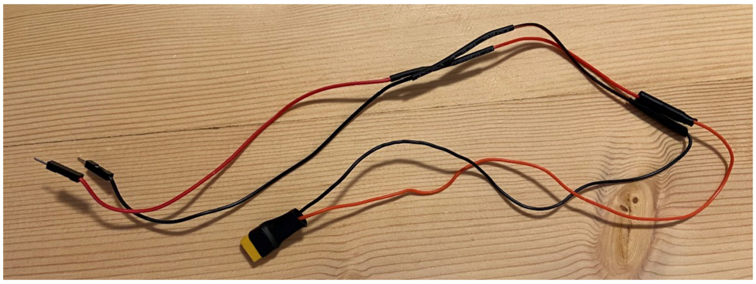


Fig. 24. 18″ wires soldered to a female XT60 connector.

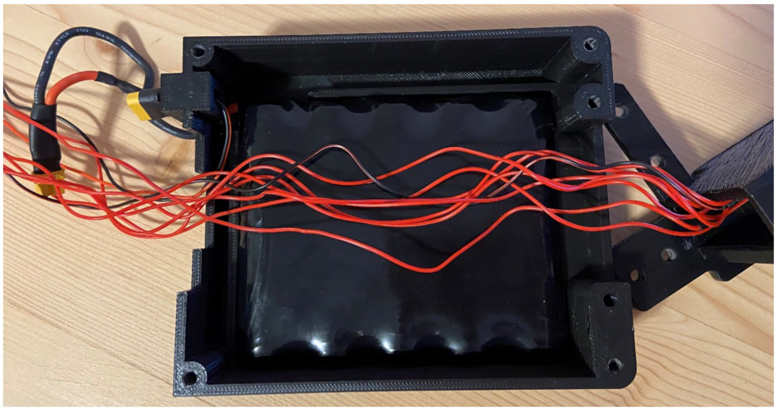


Fig. 25. The female XT60 Connector has been placed through its opening in the Battery Housing, and the wires from the D-Pad assembly are running to the left through the slot in the Battery Housing.

Next take 3 short male-to-female jumper wires and connect them to the SCL, SDA, and ADDR pins of the ADS1115 as seen in Fig. 43.

Part G: Final Hardware Assembly

To begin the final assembly, attach the Wire Track (CAD06) to the 8020 Beam using double-sided tape. Before attaching the Wire Track, though, get the wires from the D-Pad assembly, the XT60 Connector wires, and the 9 V connector wires cor-

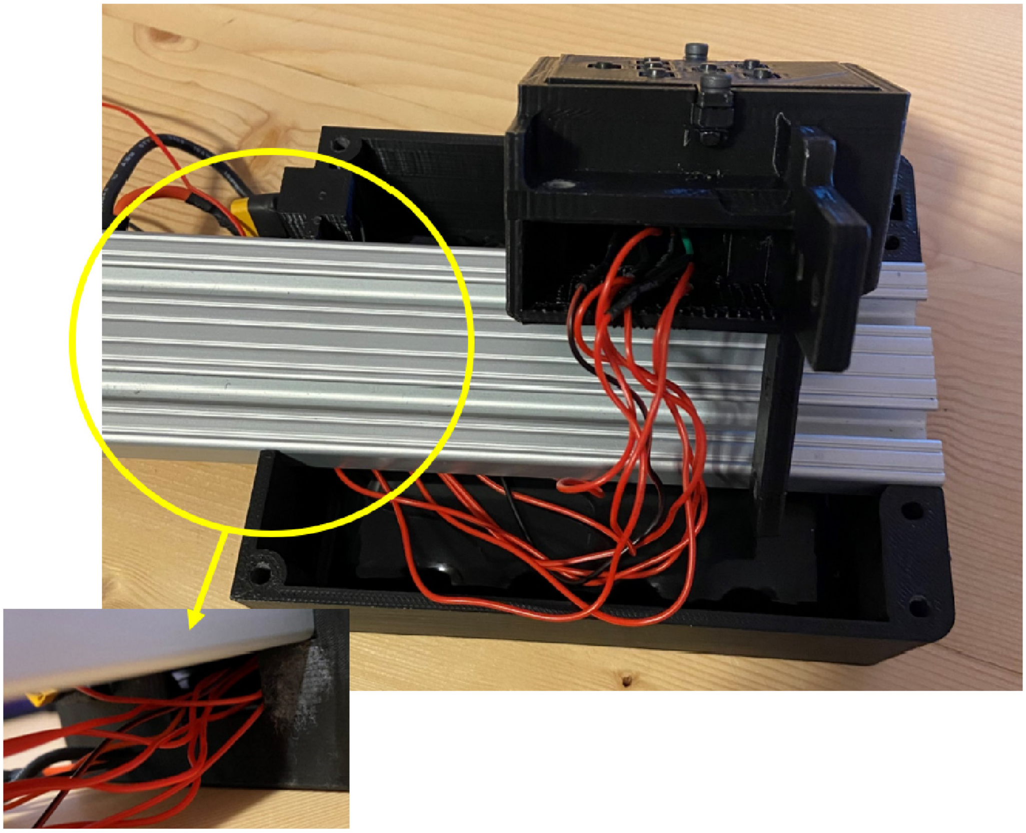


Fig. 26. The 8020 Beam has been placed across the Battery Housing.

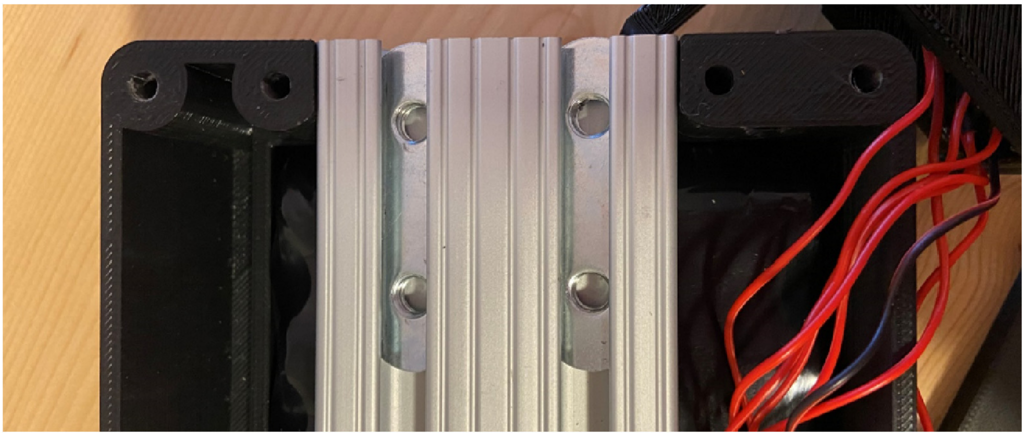


Fig. 27. The Double T-Nuts have been inserted into the 8020 Beam.

rectly into position. See Fig. 44, where the D-Pad and XT60 Connector wires run through the Wire Track while the 9 V connector only goes partway through the wire track.

Now take 4 small pieces of double-sided tape and attach them to the Wire Track as seen in Fig. 45. It may be helpful to zip tie the D-Pad/XT60 Connector wires into one bundle to keep them managed. Be sure to use industrial-strength doubled-sided tape for this step.

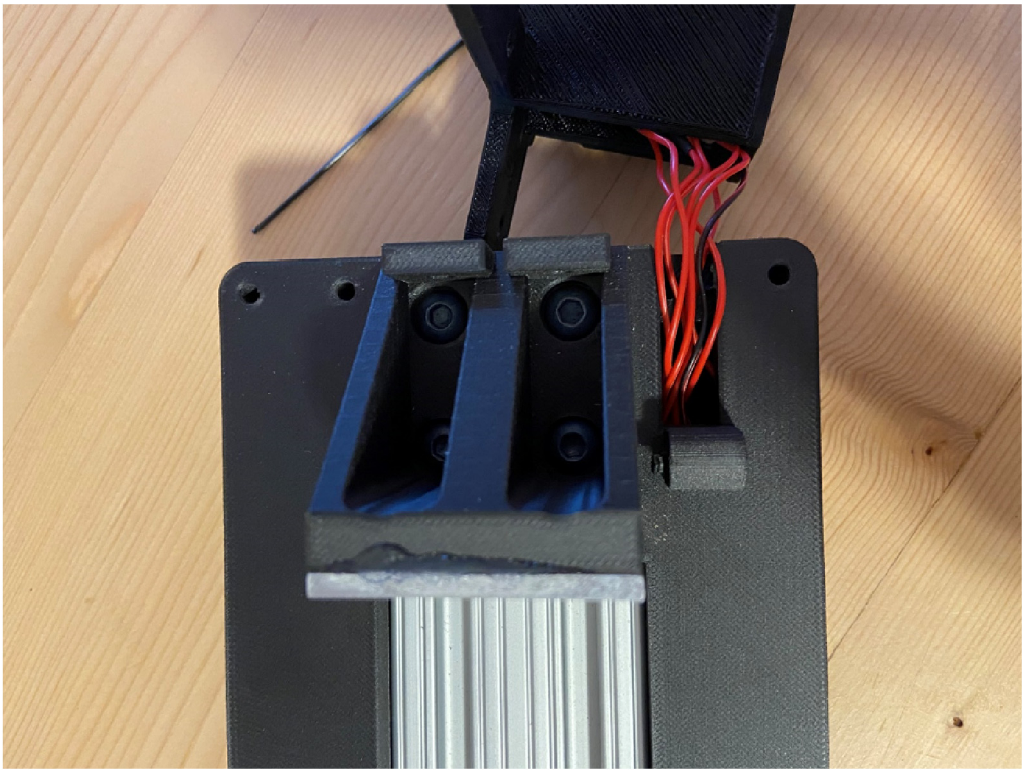


Fig. 28. The Battery Cover has been bolted to the 8020 Beam. The D-Pad wires can clearly be seen in the groove of the Battery Cover.

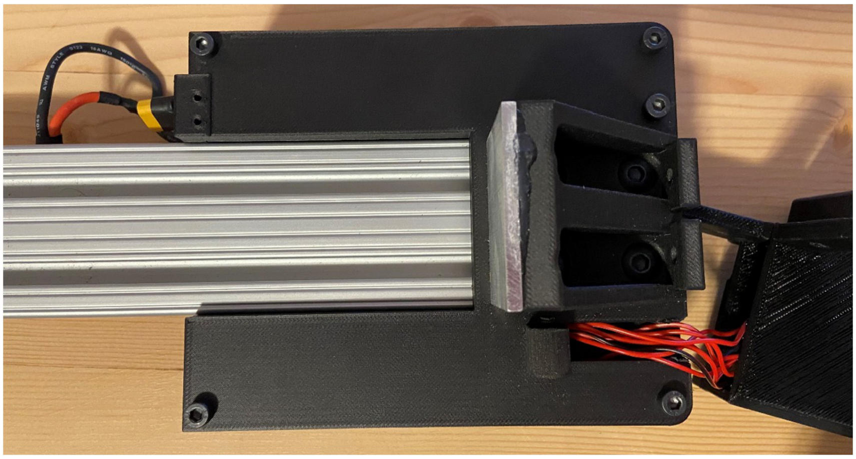


Fig. 29. The Battery Cover has been secured to the Battery Housing.

Carefully remove the double-sided tape protectors and attach the Wire Track to the bottom of the 8020 Beam. The Wire Track should be flush with the Battery Housing. Double-check that none of the wires are pinched while installing the Wire Track. See Fig. 46 for reference.

Take the Electronics Housing and plug the XT60 Connector wires into the right side of the Breadboard. These wires will bring 12 V+ to the Breadboard and will power the Stepper Motor. Also take the common 5 V+ line from the D-Pad and plug it into the left side of the Breadboard. See Fig. 47 for reference.

Now take the 7 unique ground lines from the D-Pad and plug them into the first seven rows of the Breadboard where the 10 K Ohm resistors are (see Fig. 48). The order is irrelevent, as it will be determined later using software.

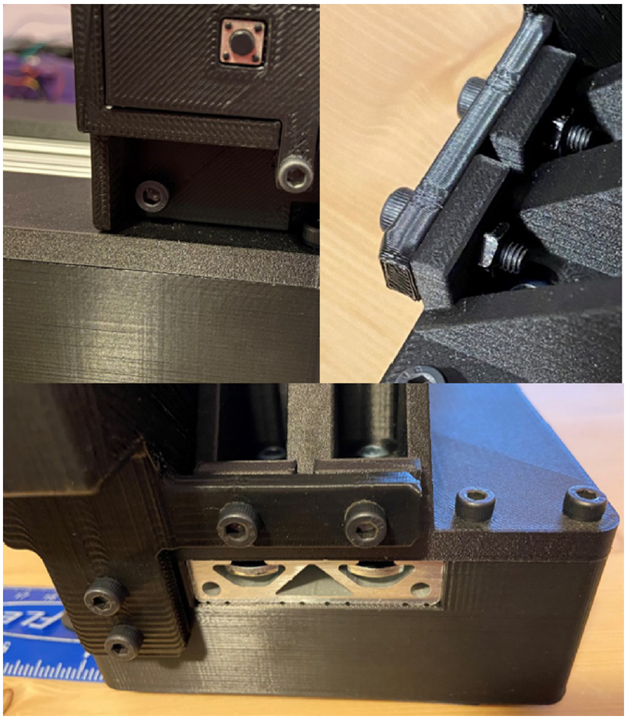


Fig. 30. TOP LEFT shows a single 4 M × 0.7 16 mm bolt connecting the D-Pad Housing to the Battery Cover. TOP RIGHT shows two 4 M × 0.7 16 mm bolts connecting the D-Pad Housing to the Battery Cover. The bolts are secured on the backside by two 4 M × 0.7 bolts. BOTTOM shows two 4 M × 0.7 10 mm bolts on the bottom left connecting the D-Pad Housing to the Battery Housing.

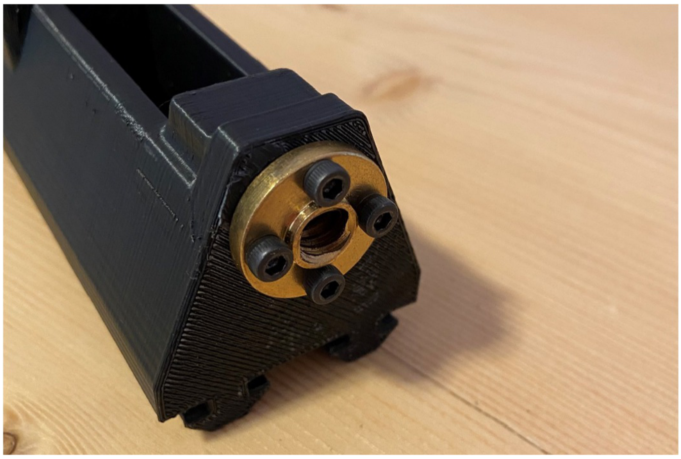


Fig. 31. The Lead Screw Nut has been secured to the Load Cell Carriage.

Now take the ADDR pin from the ADS1115 and the spare wire coming out of the wire nut in Fig. 42 and plug them both into the left ground line of the Breadboard as seen in Fig. 49.

Now plug the SCL pin of the Analog-to-Digital Converter into the Arduino pin 21 and plug the SDA pin of the Analog-to-Digital Converter into Arduino pin 20 as seen in Fig. 50.

Now the Load Cell Cable needs to be prepared. Cut the cable at roughly 24″ long and strip the four exposed wires (they are very small). Solder larger wires (or even male connector pins) to those four small wires (red, black, white, & green) so that they can be plugged into the Breadboard. The finished product is in Fig. 51.

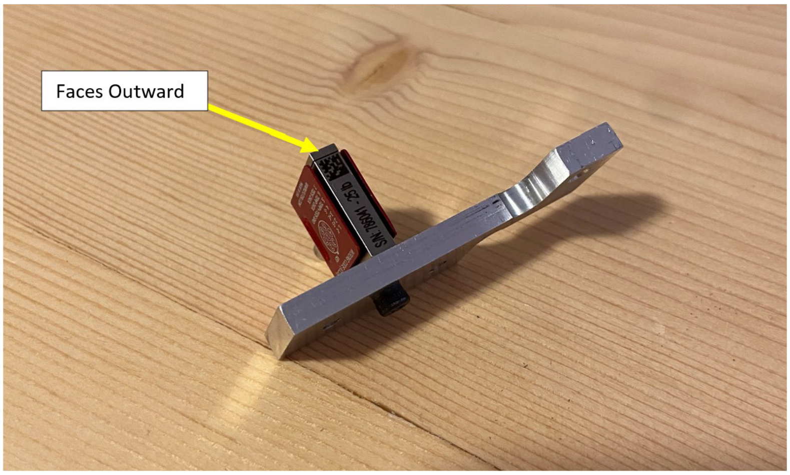


Fig. 32. The Load Cell has been attached to the Load Cell Backplate with the small gap at the bottom of the Load Cell facing outwards.

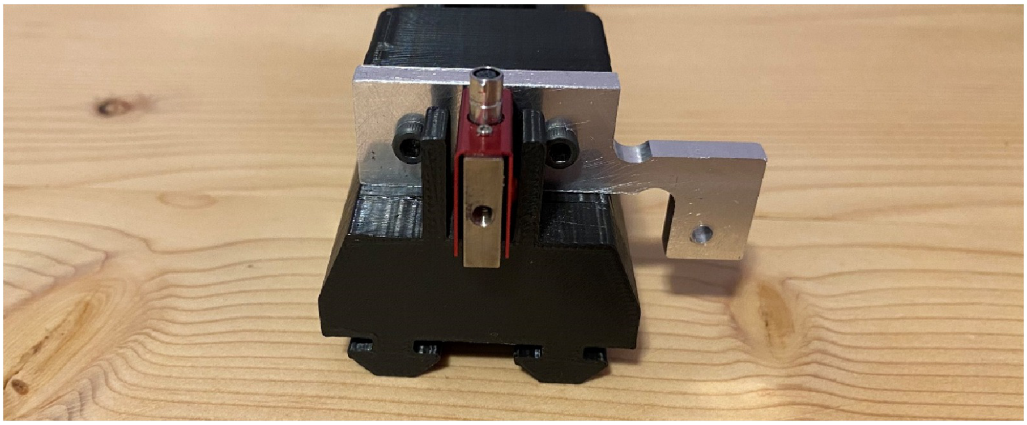


Fig. 33. The Load Cell Backplate secured to the Load Cell Carriage.

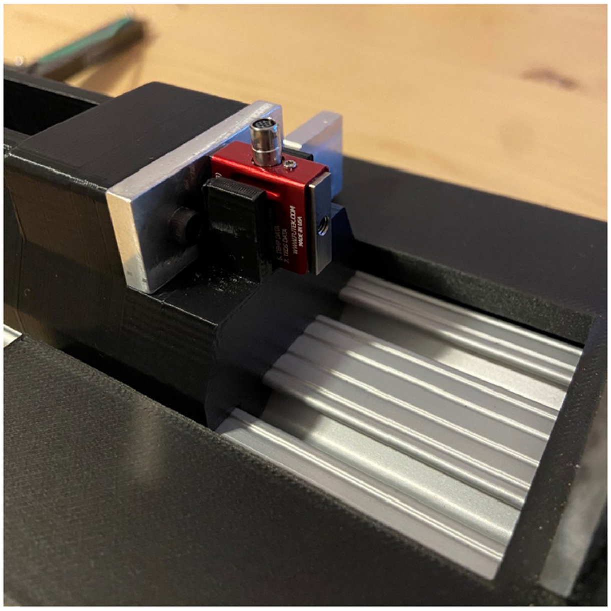


Fig. 34. The Load Cell Carriage attached to the 8020 Beam.

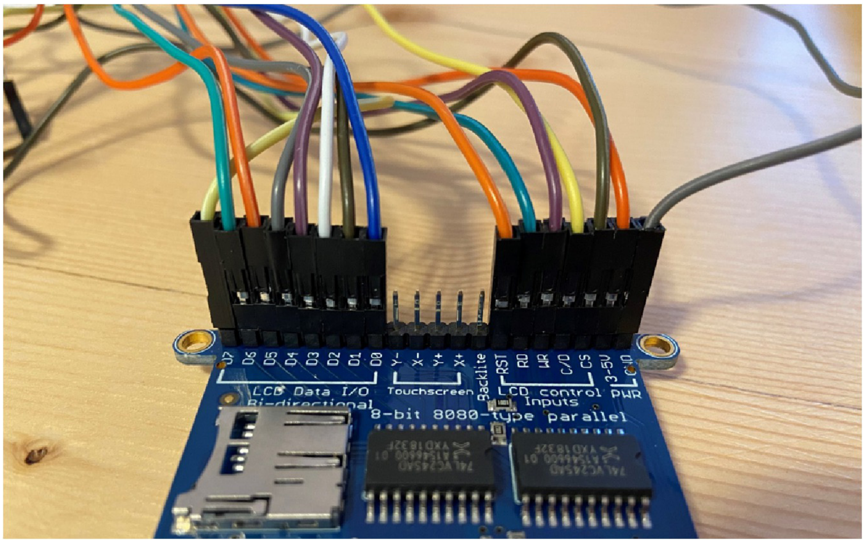


Fig. 35. The jumper wires connected to the screen. Note that the pinouts on the other side of the screen are different.

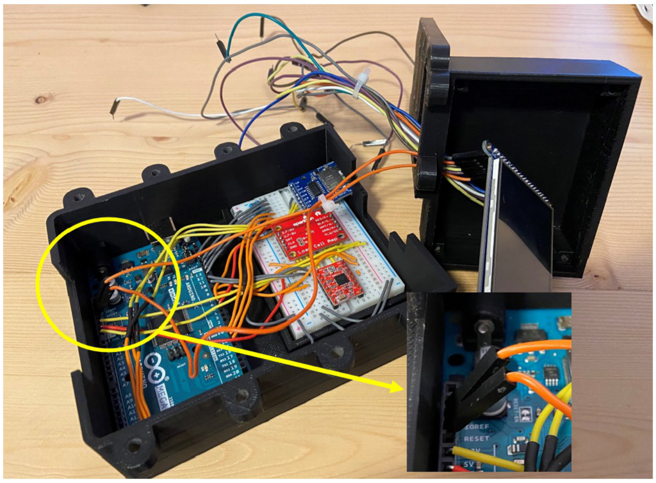


Fig. 36. The RST and 3–5 V lines of the screen have been connected to the Arduino's RESET and 3.3 V+ lines.

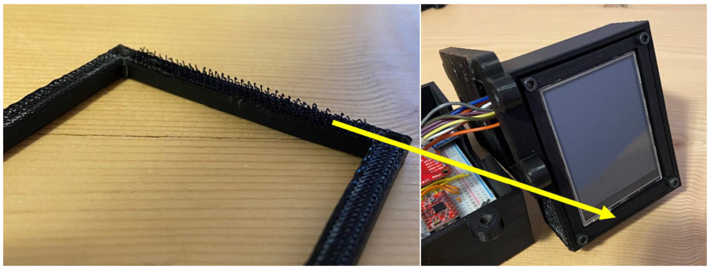


Fig. 37. LEFT shows the applied velcro and RIGHT shows that the attached Screen Border. The velcro is on the right side of the Screen Border (if applicable).

Now plug the four wires into the Breadboard. These wires connect to the same rows as the yellow jumper wires from Fig. 3. The red wire is connecting to the RED/E+ pin of the Load Cell Amplifier, the black to BLK/E-, the white to WHT/A-, and the green to GRN/A+. See Fig. 52 for reference.

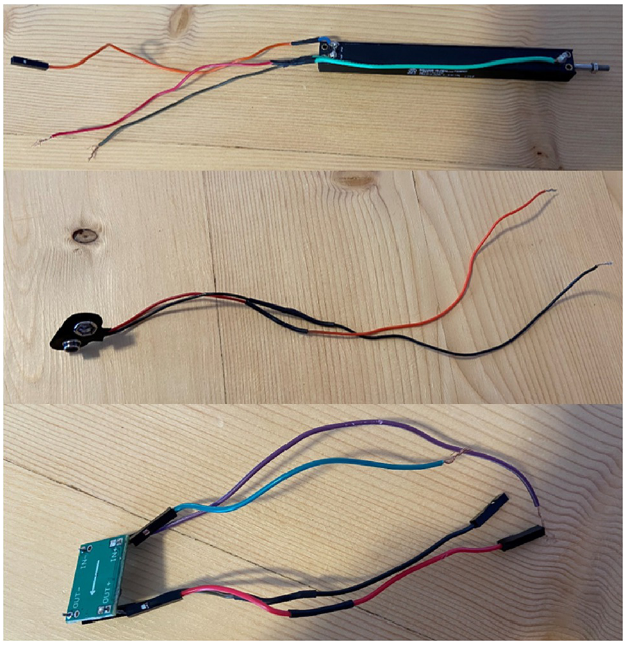


Fig. 38. (Top) The Linear Potentiometer with soldered leads. (Middle) The 9 V Battery Connector with leads. (Bottom) The Voltage Regulator with its leads attached.

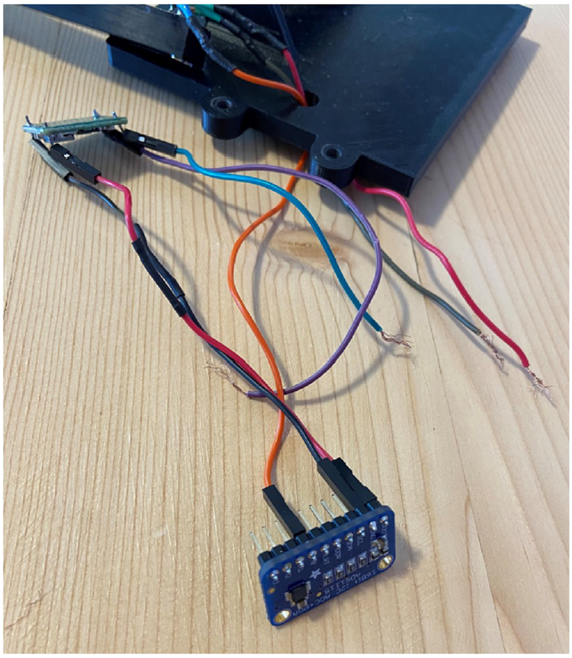


Fig. 39. The Voltage Regulator and Linear Potentiometer lines have been connected to the ADS1115.

Now pull the opposite end of the Load Cell Cable up through the hole in the Screen Housing. Pull the wires for the Stepper Motor down through the same hole in the Screen Housing. See Fig. 53 for reference.

Take the wires from the Stepper Motor and connect the blue lead to the Stepper Motor Driver' B1, the red lead to A1, black to A2, and green to B2. If these colors are different, consult the datasheet for both the Stepper Motor and Stepper Motor Driver to confirm the wire placements. See Fig. 54 for reference.

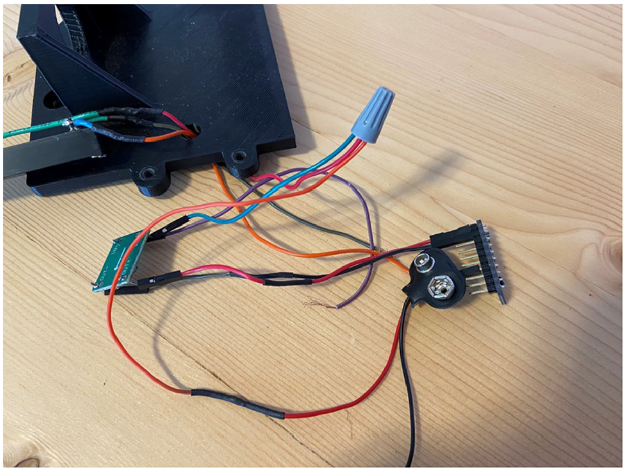


Fig. 40. The positive line of the 9 V Battery Connector, the positive lead of the Linear Potentiometer, and the IN+ line of the Voltage Regulator have been connected via the wire nut.

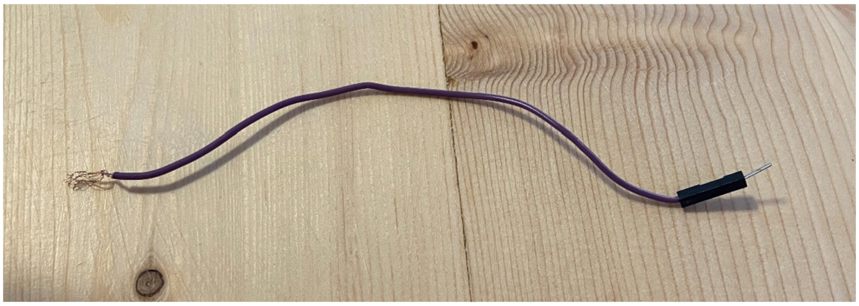


Fig. 41. Custom connector wire.

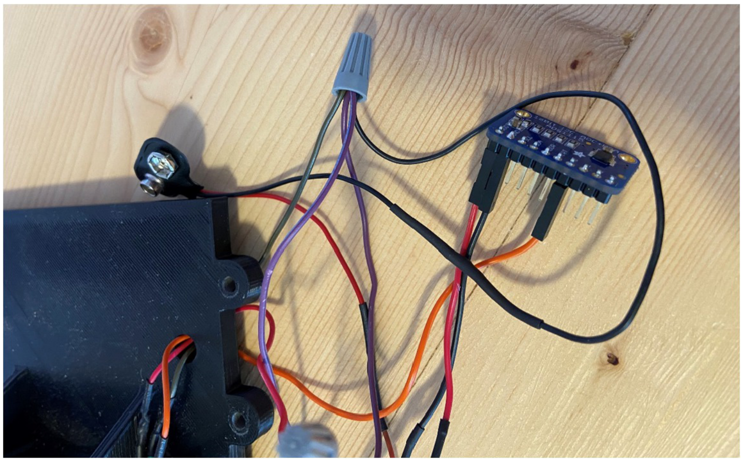


Fig. 42. The new wire, the IN- of the Voltage Regular, the GND line of the Linear Potentiometer, and the negative end of the 9 V Battery Connector have been wired together via the small wire nut.

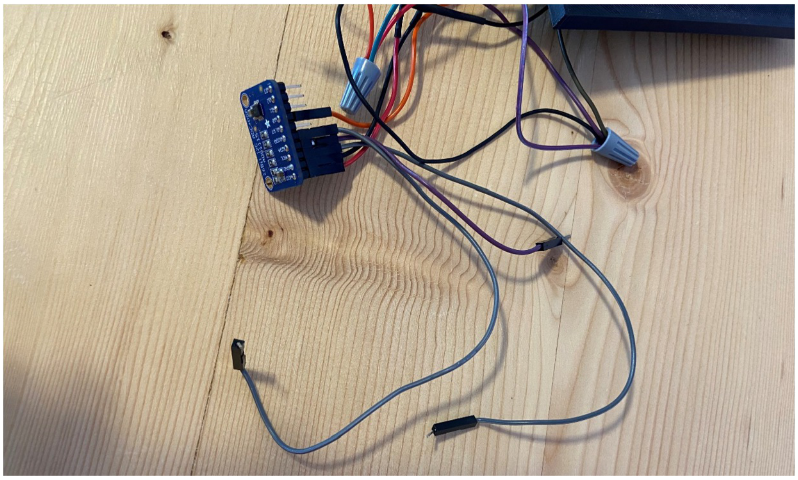


Fig. 43. Three short jumper wires have been attached to the ADS1115's SCL, SDA, and ADDR pins.

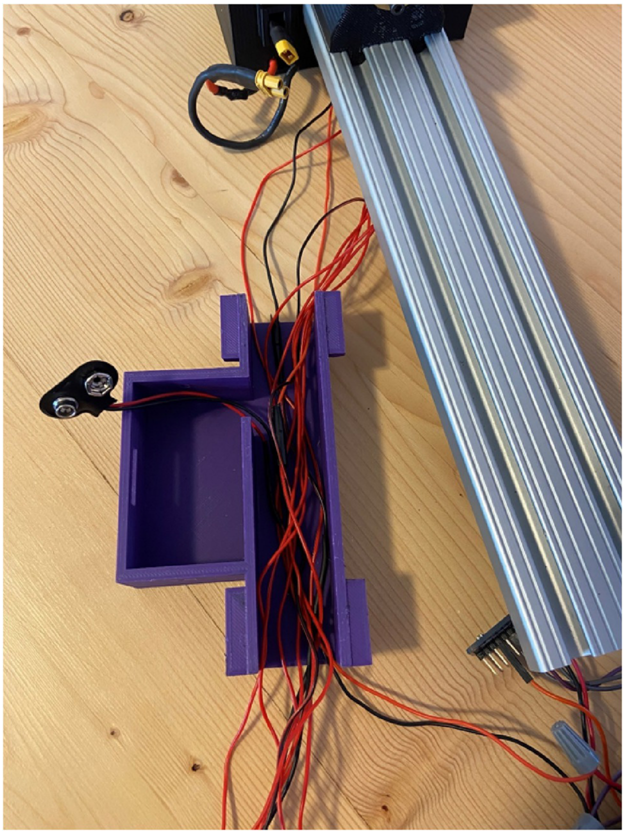


Fig. 44. The Wire Track in place with the D-Pad, XT60 Connector, and the 9 V Battery Connector wires running through the Wire Track.

At this point, it is wise to wrap both the ADS1115 and the Voltage Regulator in electrical tape to protect the two breakout boards as seen in Fig. 55.

Take two Single-T Nuts and slide them into the middle of the 8020 beam. They will be used to secure the electronics cover later. Also fit the 8020 Beam into the grooves in in the Electronics Housing. See both changes in Fig. 56.

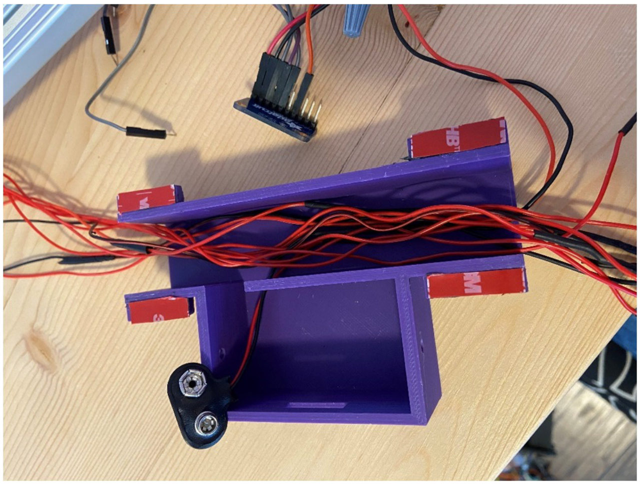


Fig. 45. The double-sided tape has been applied to the Wire Track.

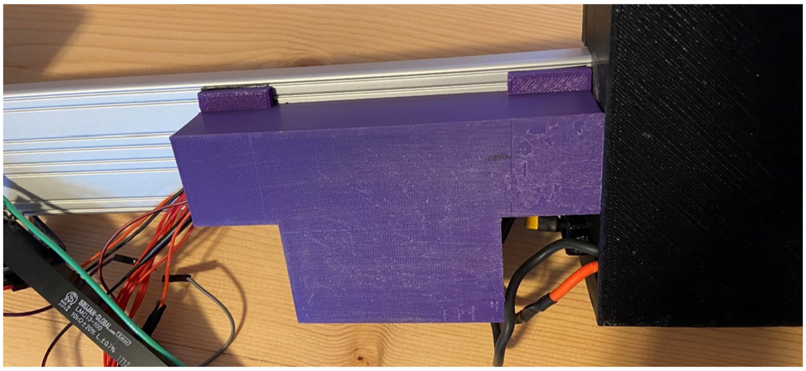


Fig. 46. The Wire Track has been attached to the bottom of the 8020 Beam.

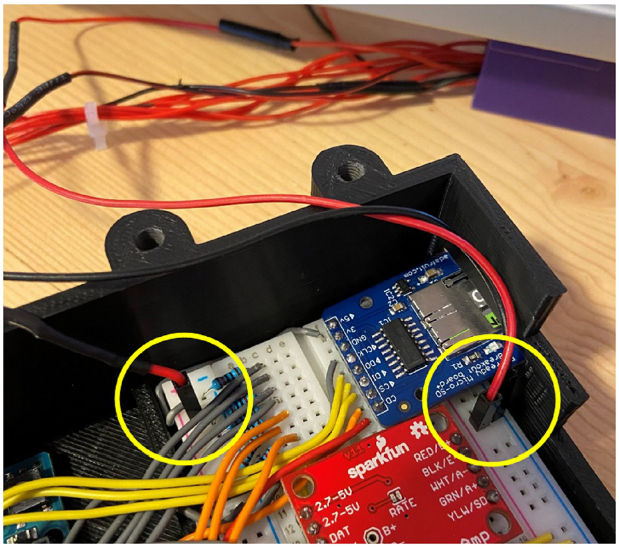


Fig. 47. On the left is the common 5 V+ line from the D-Pad, and on the right are the ends of the XT60 Connector.

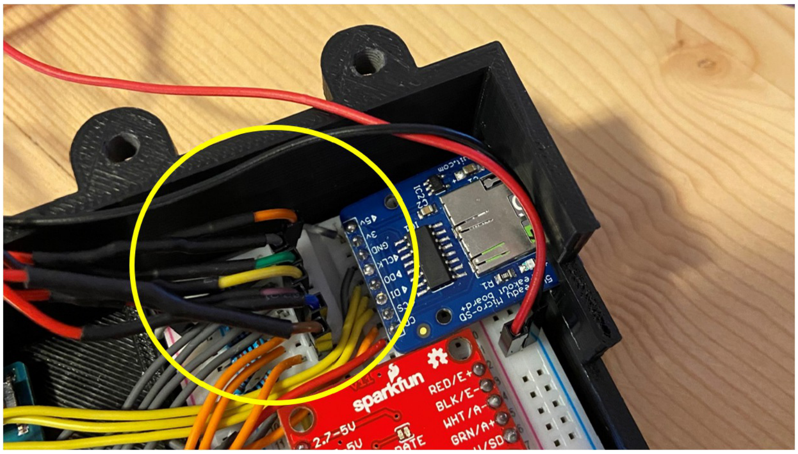


Fig. 48. The 7 unique ground lines from the D-Pad have been plugged into Breadboard rows 1–7.

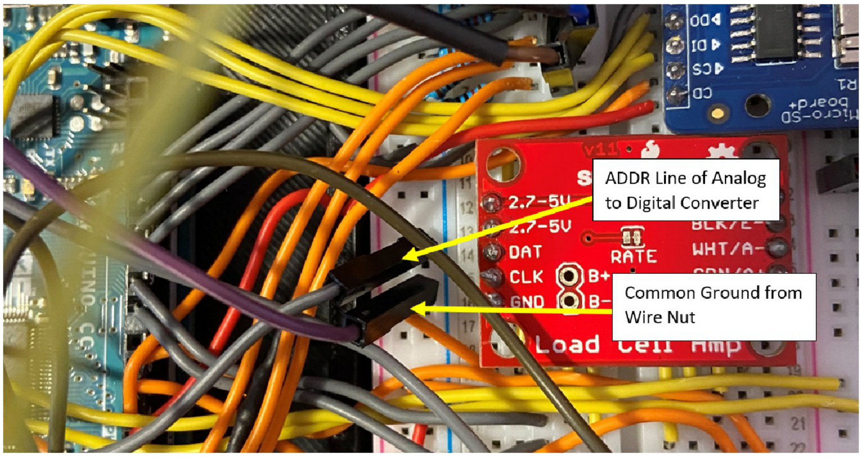

ADDR Line of Analog to Digital Converter
Common Ground from Wire Nut

Fig. 49. The ADS1115′s ADDR line and the custom wire from Fig. 41 have been plugged into the negative column on the left side of the Breadboard.

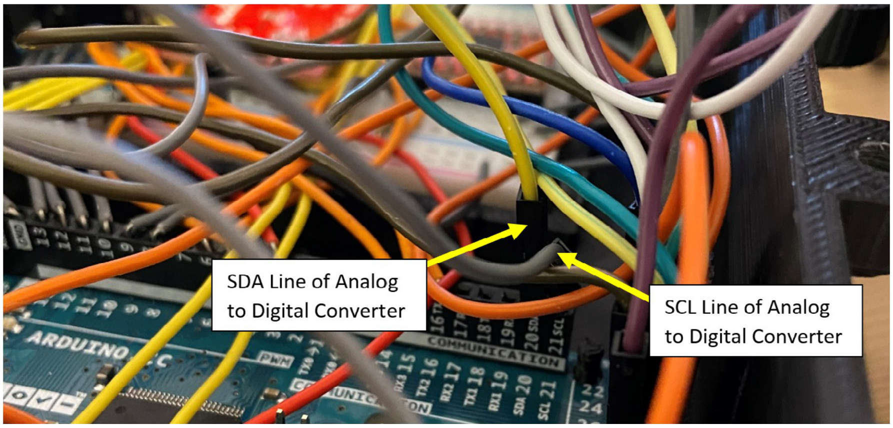

SDA Line of Analog to Digital Converter
SCL Line of Analog to Digital Converter

Fig. 50. The SDA and SDL lines of the ADS1115 plugged into the Arduino Mega.

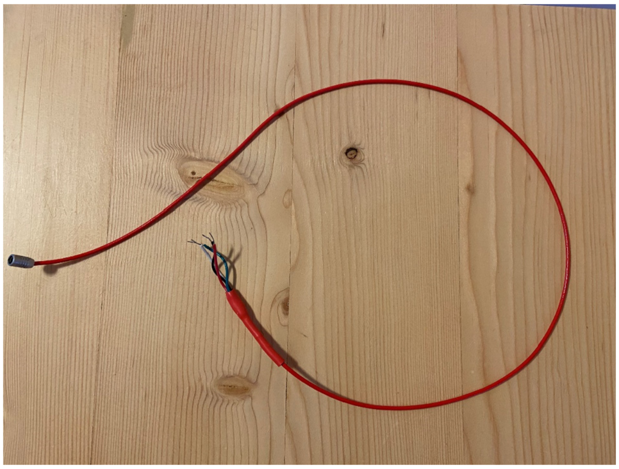
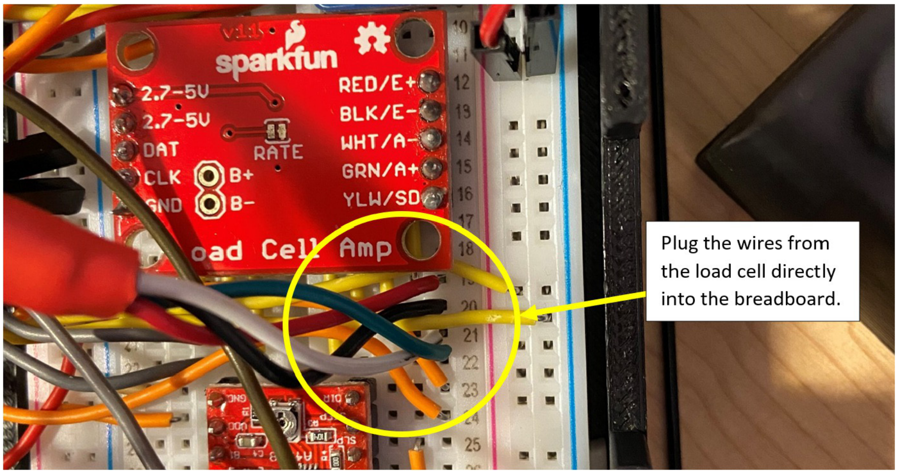


Fig. 51. The adjusted Load Cell Cable.



Plug the wires from the load cell directly into the breadboard.

Fig. 52. The four Load Cell Wires have been plugged into rows 19–22.

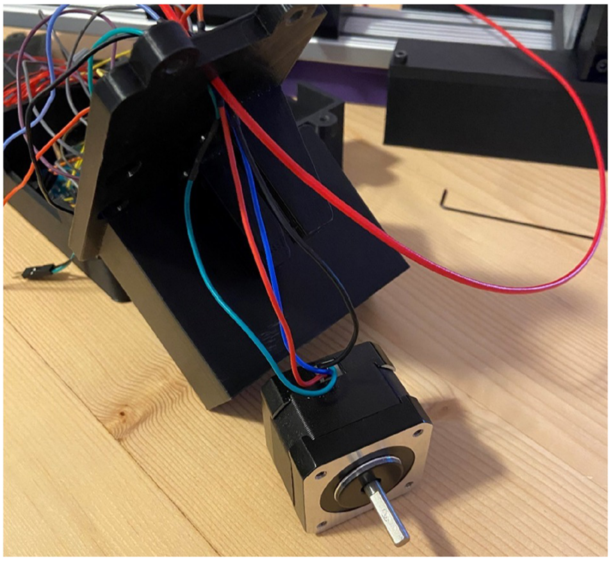


Fig. 53. The leads for the Stepper Motor and the Load Cell wire have been pulled through the top of the Screen Housing.

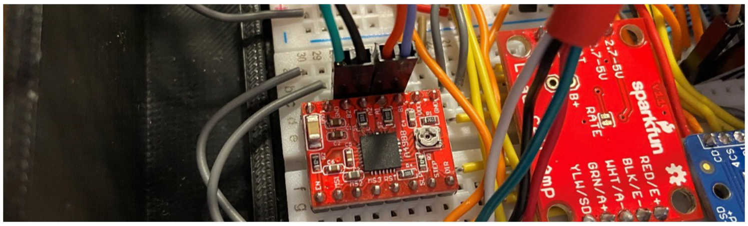


Fig. 54. The Stepper Motor has been plugged in.

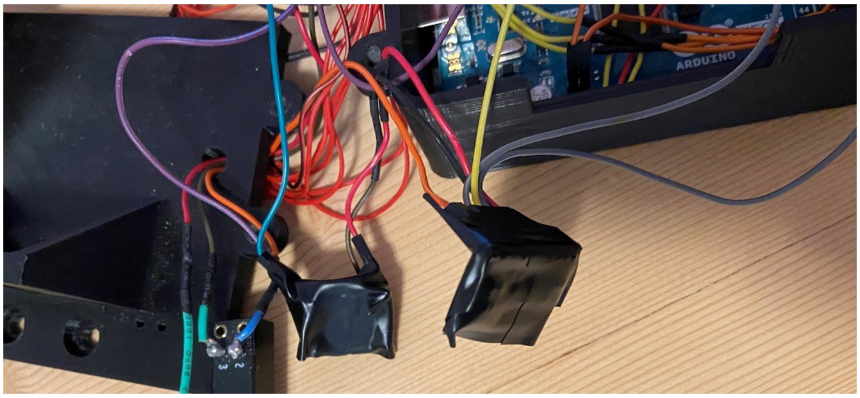


Fig. 55. The ADS1115 and the Voltage Regulator have been wrapped in electrical tape.

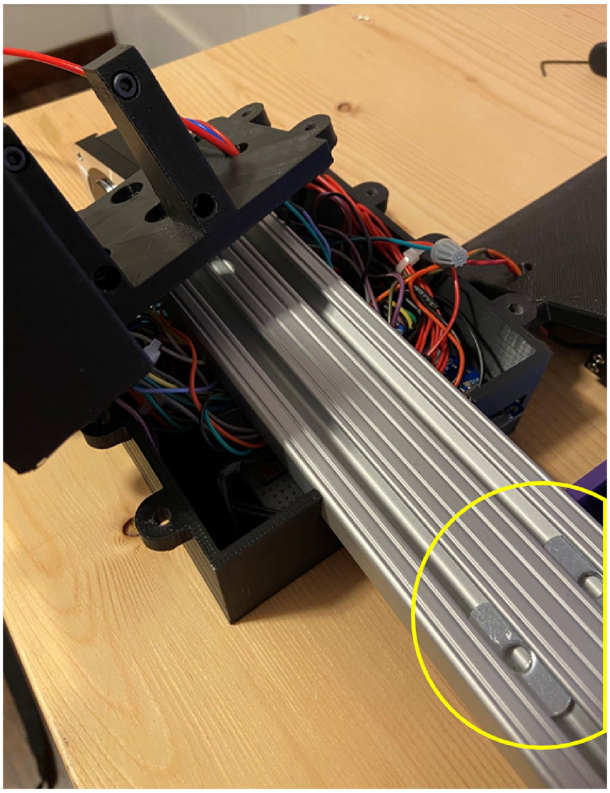


Fig. 56. The 8020 Beam has been set over the Electronics Housing and the two Single T-Nuts have been slid into the middle of the 8020 Beam.

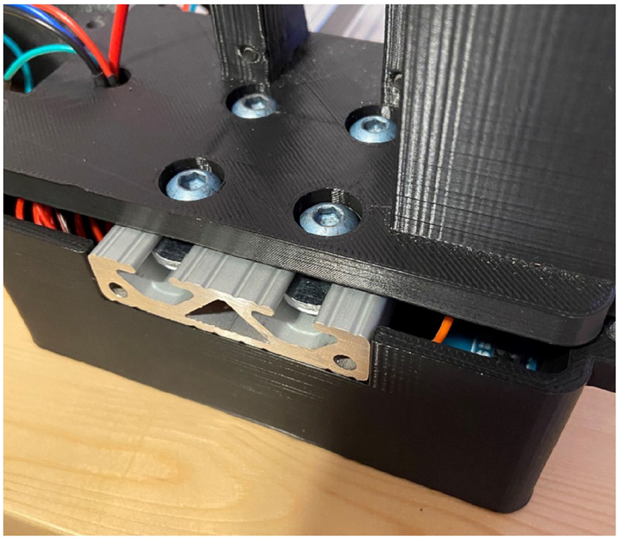


Fig. 57. The Screen Housing has been lightly attached to the 8020 beam (it can still slide).

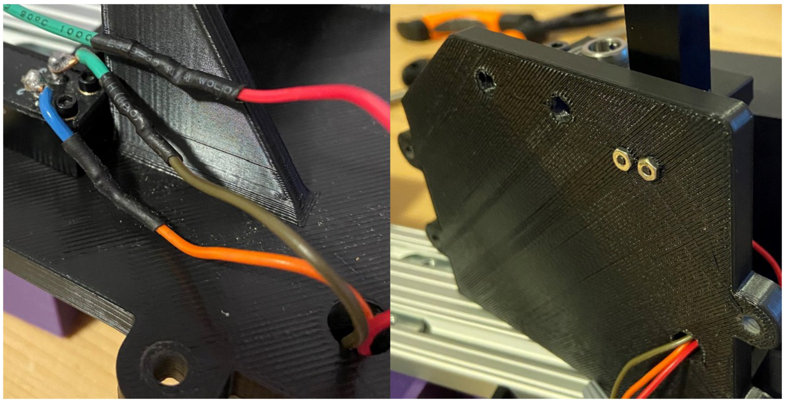


Fig. 58. LEFT shows the Linear Potentiometer connected to the Electronics Cover from the Top. RIGHT shows the location of the two 2 M × 0.4 nuts.

Attach the screen border to the 8020 beam using two Double-T Nuts as shown below. Do not completely tighted the four 6 M × 1.0 10 mm bolts so that the Screen Housing can still slide back and forth. The 8020 Beam should be flush with the outer edge of the Electronics Housing. See Fig. 57 for reference.

Now, connect the Linear Potentiometer to the Electronics Housing using two 2 M × 0.4 20 mm bolts and two 2 M × 0.4 nuts as seen in Fig. 58.

Connect the opposite side of the Linear Potentiometer to the Battery Cover. It will likely be necessary to place a washer underneath the Linear Potentiometer as a spacer in order to get the Linear Potentiometer rod in place to connect with the Load Cell Backplate. The Linear Potentiometer is secured with two 2 M × 0.4 20 mm bolts. See Fig. 59 for reference.

Now secure the rod of the Linear Potentiometer to the Load Cell Backplate using the appropriate nut as seen in Fig. 60. If the Linear Potentiometer's rod does not freely slide back and forth when bolted to the Load Cell Backplate, the hole in the Load Cell Backplate may need to be drilled out (made a little larger).

Now use eight M4 × 0.7 20 mm bolts and eight M4 × 0.7 nuts to secure both the Screen Housing and the Electronics Cover to the Electronics Housing. Also tighten down the four 6 M × 1.0 10 mm bolts connecting the screen housing to the 8020 Beam. See Fig. 61 for these changes.

Now use two M6 × 1.0 10 mm bolts to secure the Electronics Cover to the 8020 Beam using the Single T-Nuts from Fig. 56. See Fig. 62 for these updates.

Now put the Lead Screw (cut to 7.5 in) through the bearing block and into the Lead Screw Nut (from Fig. 31) on the Load Cell Carriage as seen in Fig. 63.

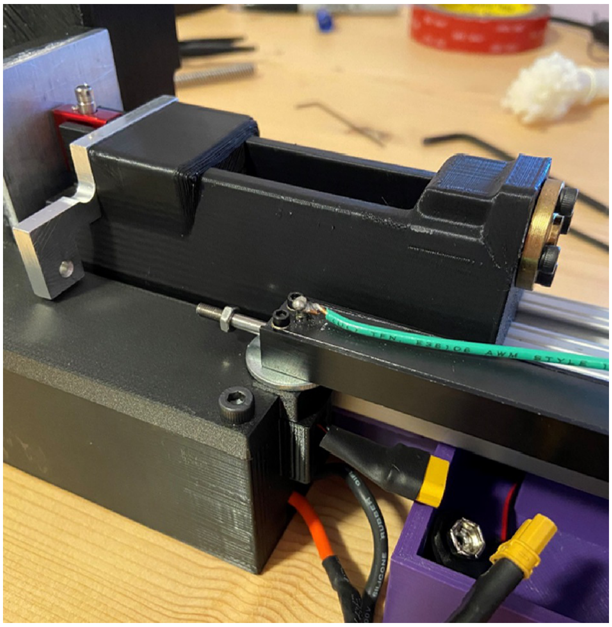


Fig. 59. The opposite side of the Linear Potentiometer connected to the Battery Cover.

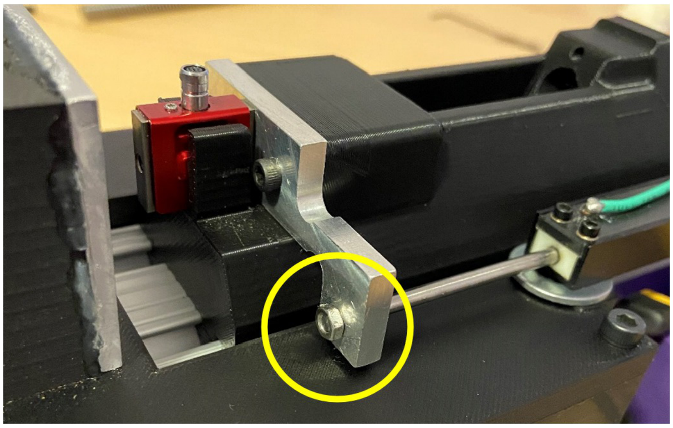


Fig. 60. The rod of the Linear Potentiometer has been connected to the Metal Backplate.

Now attach the Stepper Motor to the Screen Housing using four 3 M × 0.5 12 mm bolts. Also use the Shaft Coupler (blue) to connect the axle of the Stepper Motor to the Lead Screw as seen in Fig. 64. Be sure to tighten the set screws in both sides of the shaft collar.

Lastly, plug in the Load Cell as seen in Fig. 65.

The hardware for the Crop Clamp is now complete (Fig. 66).

Part H: Software Configuration and Calibration

Before the Crop Clamp is fully operational, the D-Pad Buttons need to be mapped and multiple calibrations need to be performed on the device. Using the Arduino IDE, upload the Crop_Clamp_Buttons.ino sketch to the Crop Clamp. While the sketch is running on the Crop Clamp, open the Serial Monitor within the Arduino IDE (nothing will be displayed on the Crop Clamp's Screen). When a button is pressed on the D-Pad, the corresponding Arduino pin will be printed to the Serial Monitor.

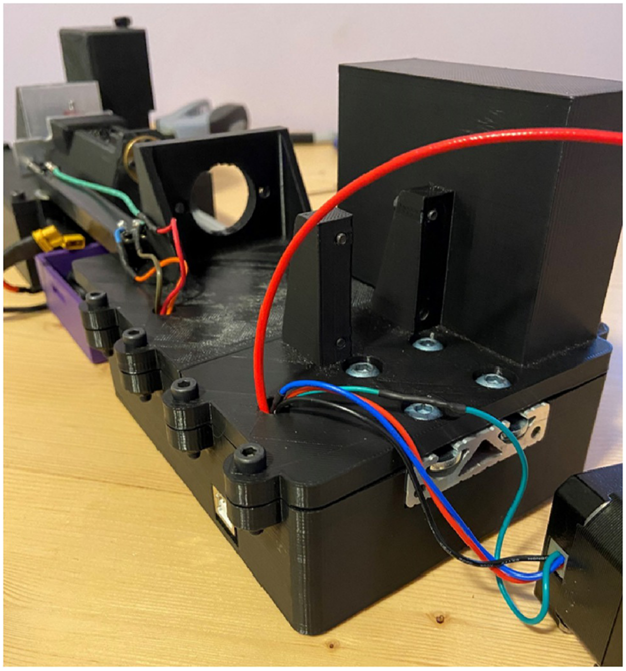


Fig. 61. Both the Screen Housing and the Electronics Cover have been secured.

Note which ports correspond to which buttons and open the CropClamp_OS.ino sketch and replace the pin numbers near the top of the sketch with the values found from the Crop_Clamp_Buttons.ino sketch. The section of the code that needs adjusted can be found in Fig. 67. Note that SELECT_UP is the OKAY button, and SELECT_DOWN is the BACK button.

When the pins are adjusted, the CropClamp_OS.ino sketch is ready to be uploaded to the Crop Clamp. It is likely that several Arduino libraries will need to be installed to the Arduino IDE before the CropClamp_OS.ino sketch can be compiled. These libraries can be found by searcing in the Arduino IDE's library manager (Sketch->Include Library->Manage Libraries). Some of the built-in libraries may need to be downgraded to previous versions. The full list of libraries is below:

- SD – Built-In by Arduino, Sparkfun V1.2.4
- TFT – Built-In by Arduino, Adafruit V1.0.6
- Adafruit ADS1X15 – by Adafruit V1.1.1
- Adafruit GFX Library – by Adafruit V1.10.1
- Adafruit FTFLCD Library – by Adafruit V1.0.3
- HX711 – by Rob Tillaart V0.2.0 (if not available from Arduino IDE, find at: https://github.com/RobTillaart/HX711)

An additional library has to be installed from gitHub:

- MemoryFree - https://github.com/maniacbug/MemoryFree

The CropClamp_OS.ino sketch should compile with these libraries installed and can now be uploaded to the Crop Clamp. Before the Crop Clamp is ready for normal use, the user *must* calculate both the natural offset of the device and the calibration factor for the Load Cell. The natural offset is the amount of flex the device experiences during a normal test. Instructions for calculating and accounting for the natural offset can be found in the Operating Instructions under the Calibration heading. The Load Cell calibration factor scales the output of the Load Cell so that it reflects the actual force readings. Instructions for performing this calibration can also be found under the Calibration heading in Operating Instructions.

Operation instructions

The Crop Clamp is designed for use on a tabletop. The device does not have to be level to operate correctly and can be operated with the user holding the device in their hands. However, the ergonomics of the device are optimized for table top use and not for handheld use.

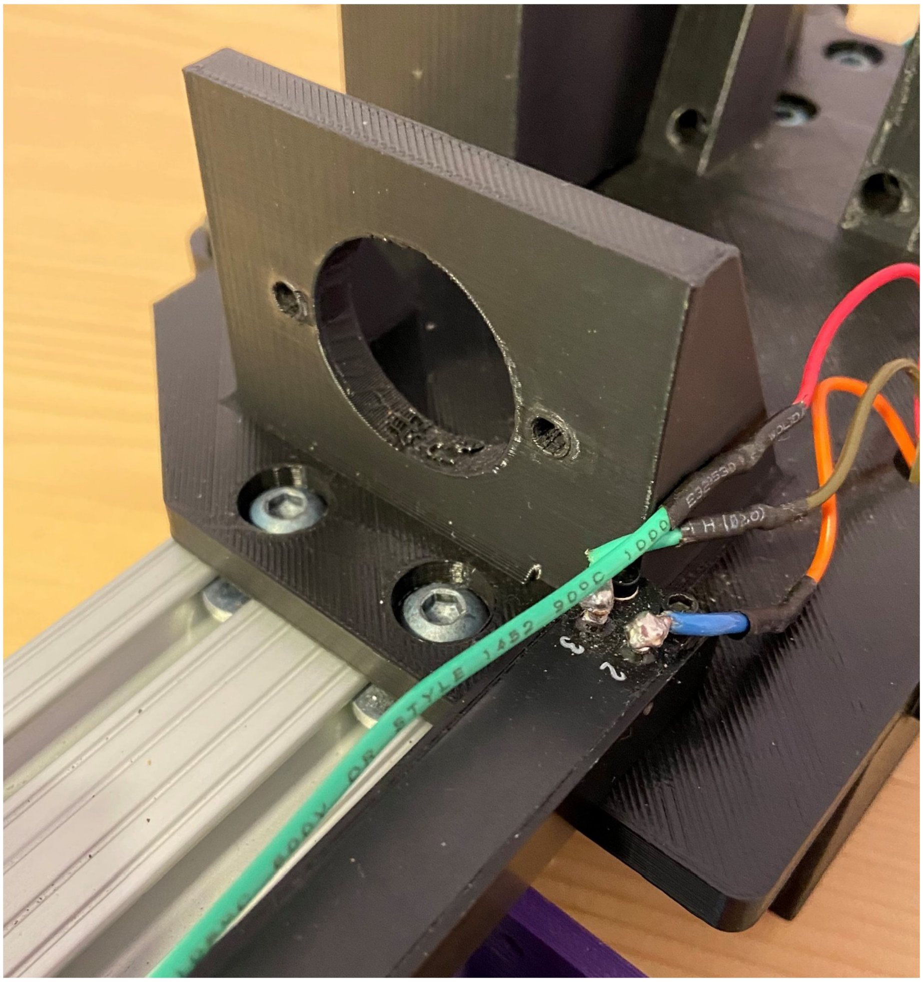


Fig. 62. The Electronics Cover has been secured to the 8020 Beam.

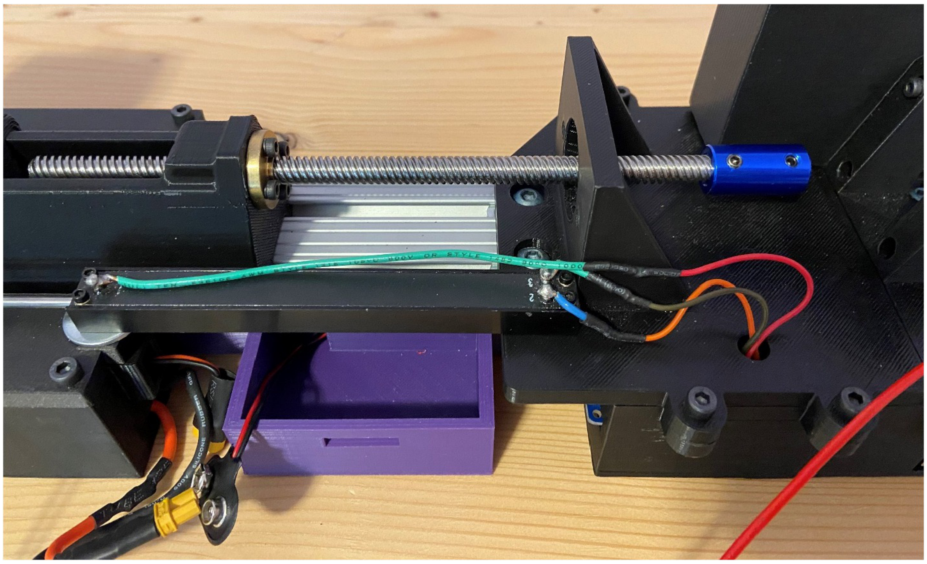


Fig. 63. The Lead Screw has been connected to the Load Cell Carriage.

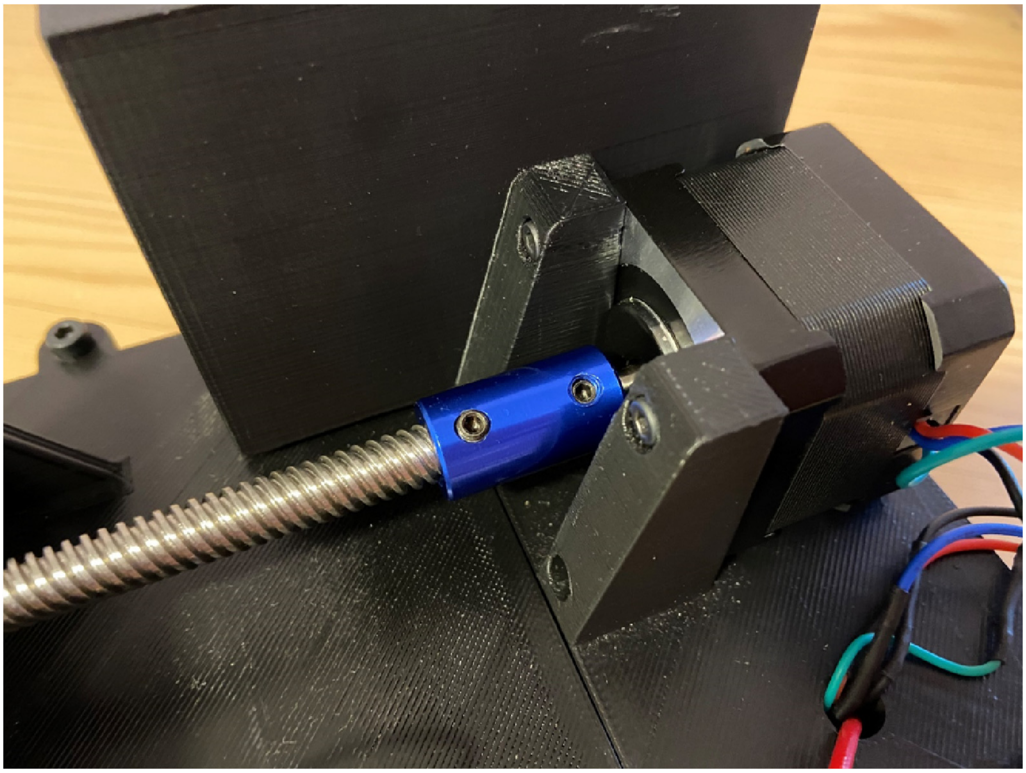


Fig. 64. The Stepper Motor has been secured and the motor shaft has been connected to the Lead Screw.

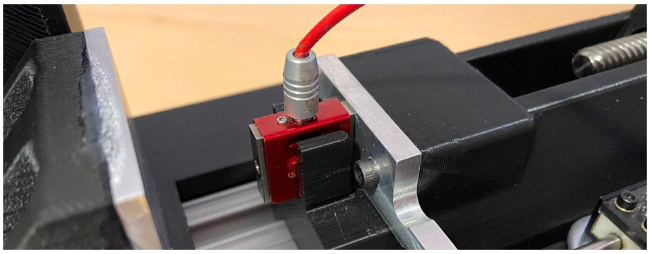


Fig. 65. The Load Cell has been plugged in.

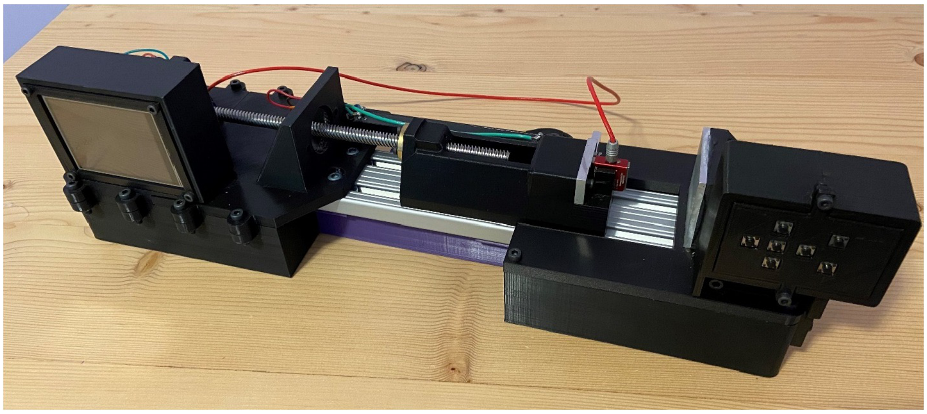
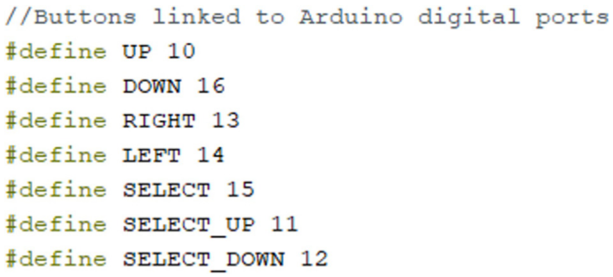


Fig. 66. The completed Crop Clamp.



```
//Buttons linked to Arduino digital ports
#define UP 10
#define DOWN 16
#define RIGHT 13
#define LEFT 14
#define SELECT 15
#define SELECT_UP 11
#define SELECT_DOWN 12
```

Fig. 67. These pin numbers must be changed to reflect the new wiring.

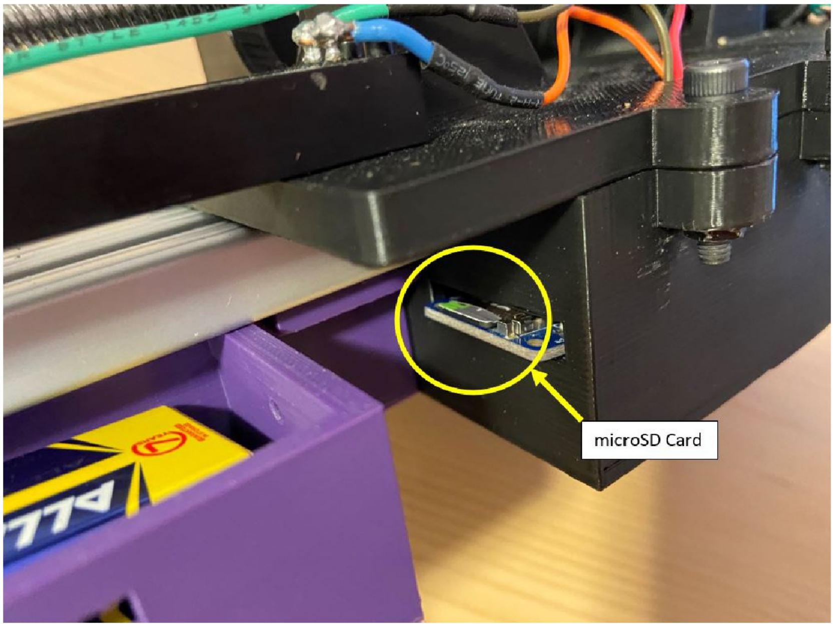


Fig. 68. The microSD card has been inserted into the microSD Card Reader.

Before powering the Crop Clamp, plug a microSD Card into the microSD Card Reader as seen in Fig. 68. The microSD Card must be formatted as a FAT32 card and it should be empty.

The Crop Clamp requires the internal battery to be charged and an external 9 V battery to operate. Begin by linking the XT60 connectors and plugging a 9 V into the 9 V Connector as seen in Fig. 69.

When the Crop Clamp is powered on, the Home Screen will appear as seen in Fig. 70. If there is a problem with the microSD card, an error page will appear prompting the user to insert the microSD card or continue without microSD functionality. If the device is used without microSD functionality, no test data will be saved. If the error page persists, there is likely a wiring error (Fig. 71).

If the device has *never* been powered on before, then the user *must* calculate the natural offset of the device as well as the Load Cell calibration factor. See the Calibration heading below for instructions on how to perform these operations.

There are six options from the Home Screen. Use the UP and DOWN buttons on the D-Pad to navigate through the options, and use SELECT to choose an option. Here is a summary of the available options:

- Raw Measure shows a live output of force from the Load Cell and displacement calculated from stepper motor rotation count.
- File Read takes a transverse stiffness measurement of a stalk and saves it to the microSD card. The device can measure between 1 and 10 internodes per stalk.
- Settings allows the user to change the device's operating and GUI settings.
- Reboot SD will reinitialize the microSD card. If the microSD card becomes disconnected, the messages at the bottom of the screen will display *SD Card: Not Inserted* and *SD Inactive*. The user must re-insert the microSD card and use the Reboot SD option before data can be collected again.

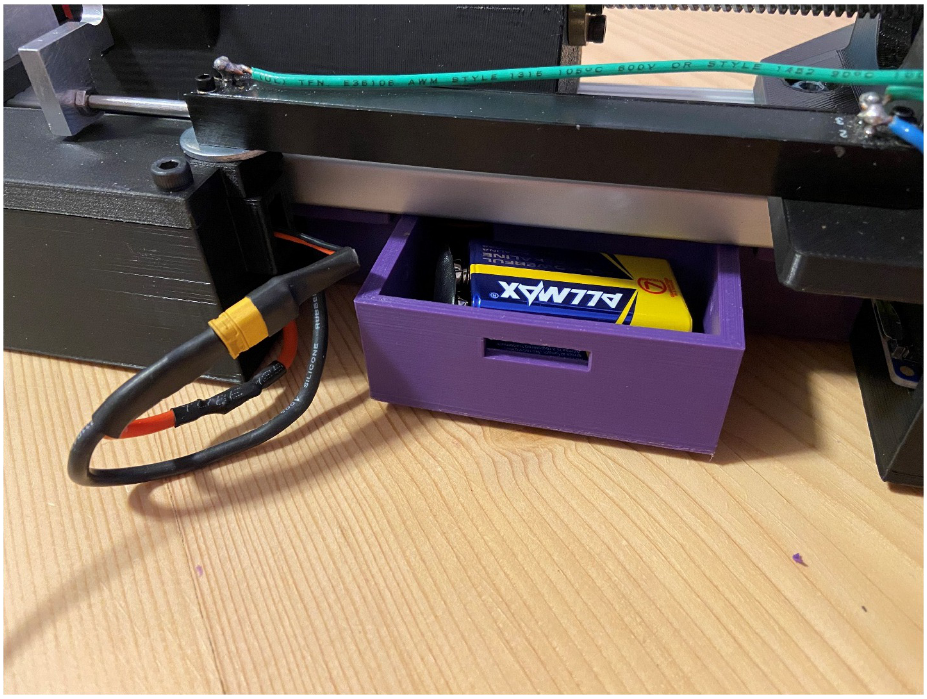
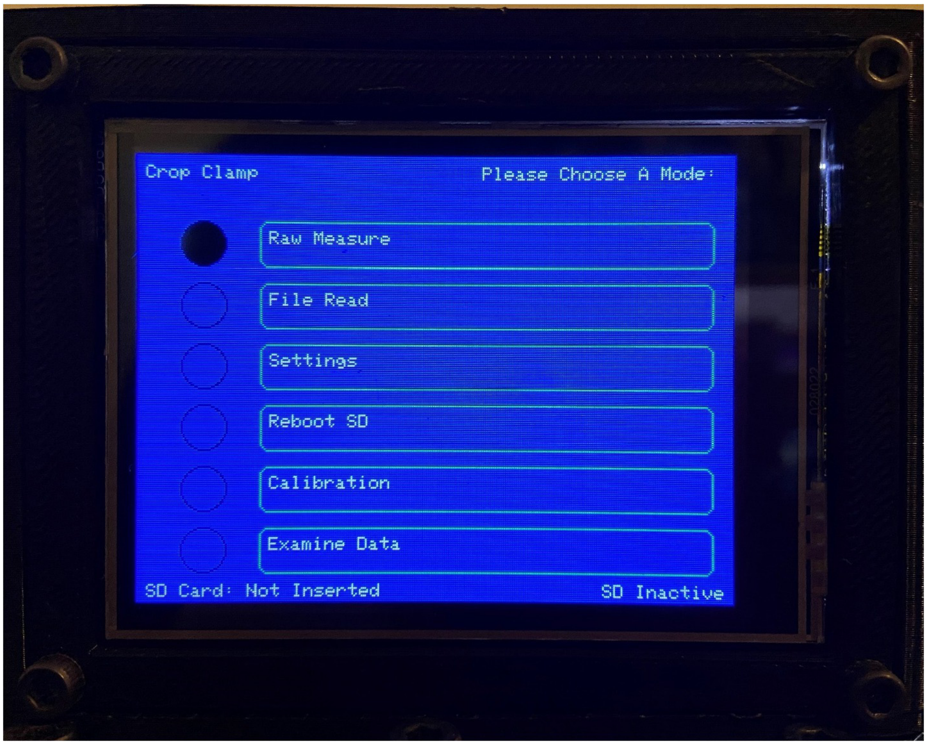


Fig. 69. The internal battery has been plugged in via the XT60 connectors, and the 9 V has been connected.

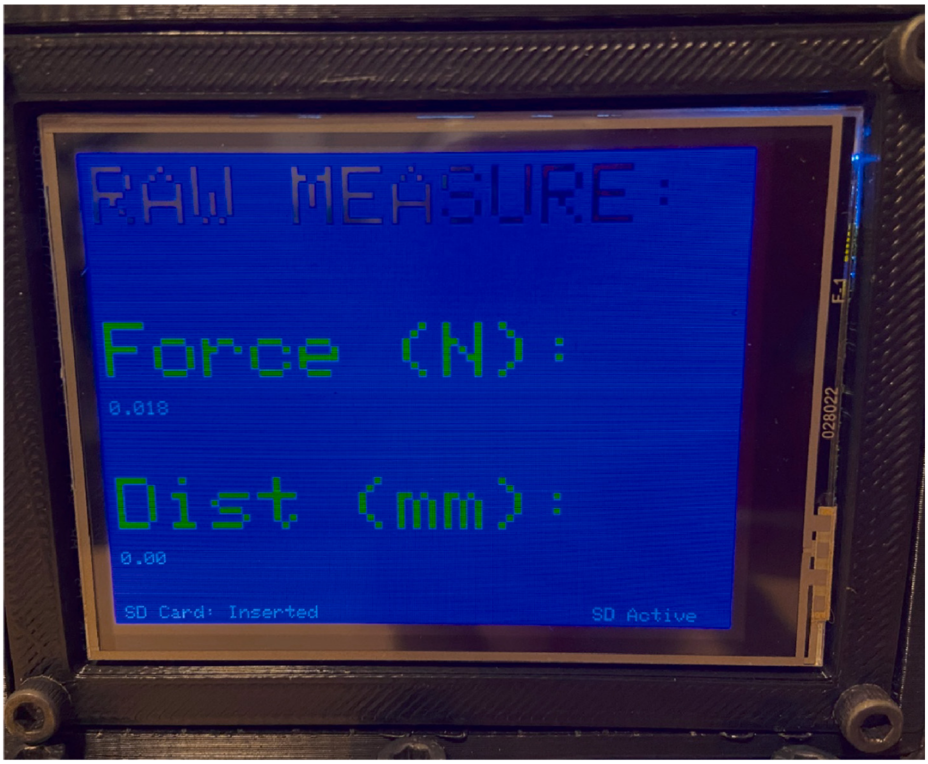


Please Choose A Mode:

SD Card: Not Inserted

SD Inactive

Fig. 70. This is the default Home Screen.


RAW MEASURE:

Force (N):

0.018

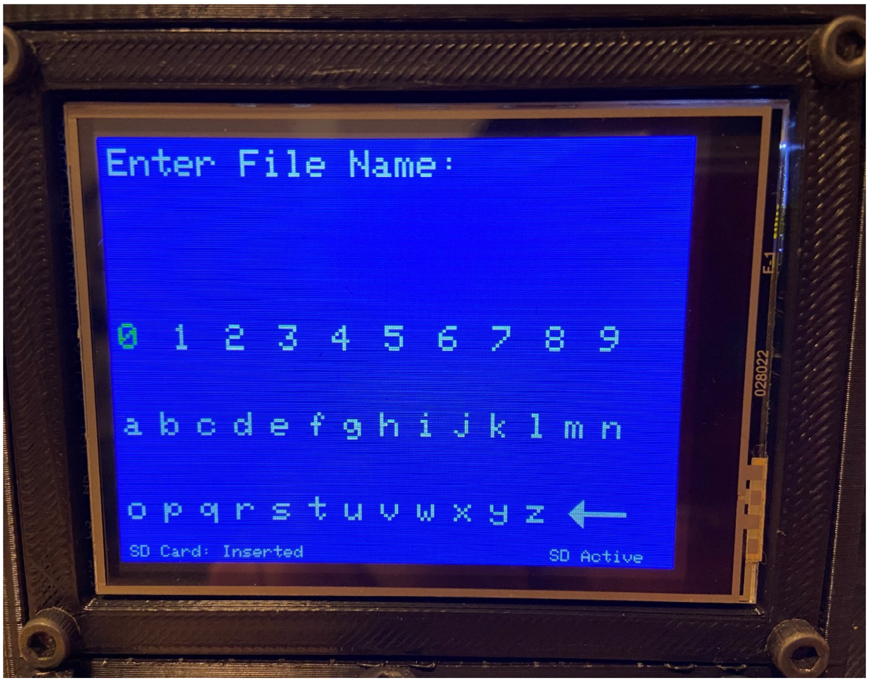
Dist (mm):

0.00

SD Card: Inserted

SD Active

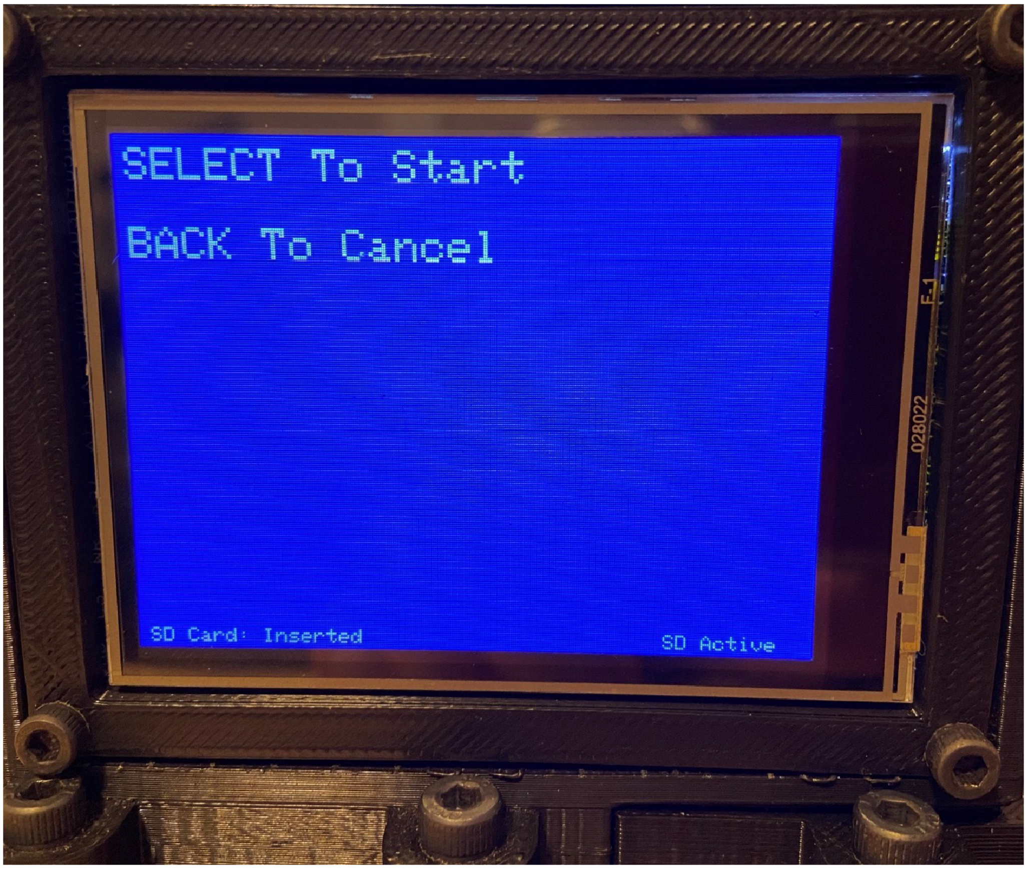
Fig. 71. The Raw Measure Screen (SI units).


Enter File Name:

SD Card: Inserted

SD Active

Fig. 72. The stalk name entry screen.



SELECT To Start

BACK To Cancel

SD Card: Inserted

SD Active

Fig. 73. The pre-test screen.

- Calibration will allow the user to calibrate the Linear Potentiometer, the natural offset of the device, and the Load Cell's calibration factor. The Linear Potentiometer must be calibrated every time the device is turned on, but natural offset calibration only has to occur on the first boot-up and every five hundred tests during normal operation. Similarly, the Load Cell calibration factor only needs to be calculated on the device's first boot-up.
- Examine Data allows the user to examine data graphs and transverse stiffness values for past tests.

Raw measure

While on this screen, the force and distance displayed on the screen are real-time measurements. The following buttons will perform the following actions:

UP: Moves the Stepper Motor right 20 steps (0.2 mm/0.00787in)
DOWN: Moves the Stepper Motor left 20 steps (0.2 mm/0.00787in)
RIGHT: Moves the Stepper Motor right 4 steps (0.04 mm/0.00157in)
LEFT: Moves the Stepper Motor left 4 steps (0.04 mm/0.00157in)
SELECT: Sets the distance to zero and tares (zeroes) the load cell
OK: Moves 100 steps in the direction of the last move (+-1.00 mm/0.03937in)
BACK: Returns to the Home Screen

In Raw Measure mode, the displayed distance is determined using the number of Stepper Motor rotations. This becomes inaccurate when substantial loads are applied to the device. In File Read mode, the distances are taken from the Linear Potentiometer and are accurate (+- 0.001 mm) regardless of applied load.

File Read

The File Read option can only be accessed from the Home Screen *after* the Linear Potentiometer has been calibrated and the microSD card has been properly initialized. If these requirements have been met, the Crop Clamp will prompt the user to

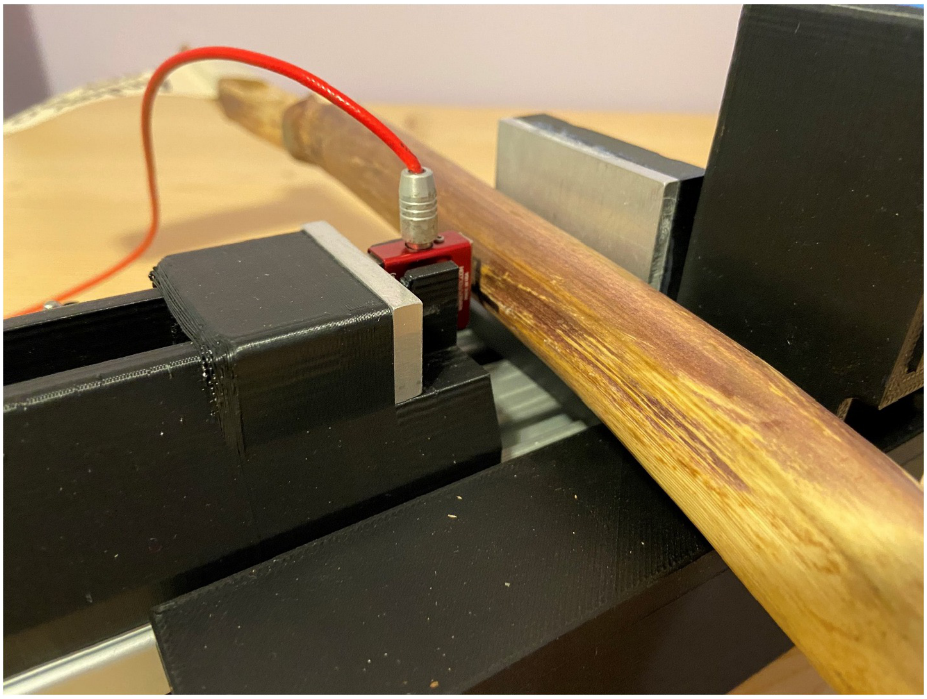


Fig. 74. A maize stalk positioned in the Crop Clamp for testing. The stalk is touching the Metal Backplate and is about half an inch from the Load Cell.

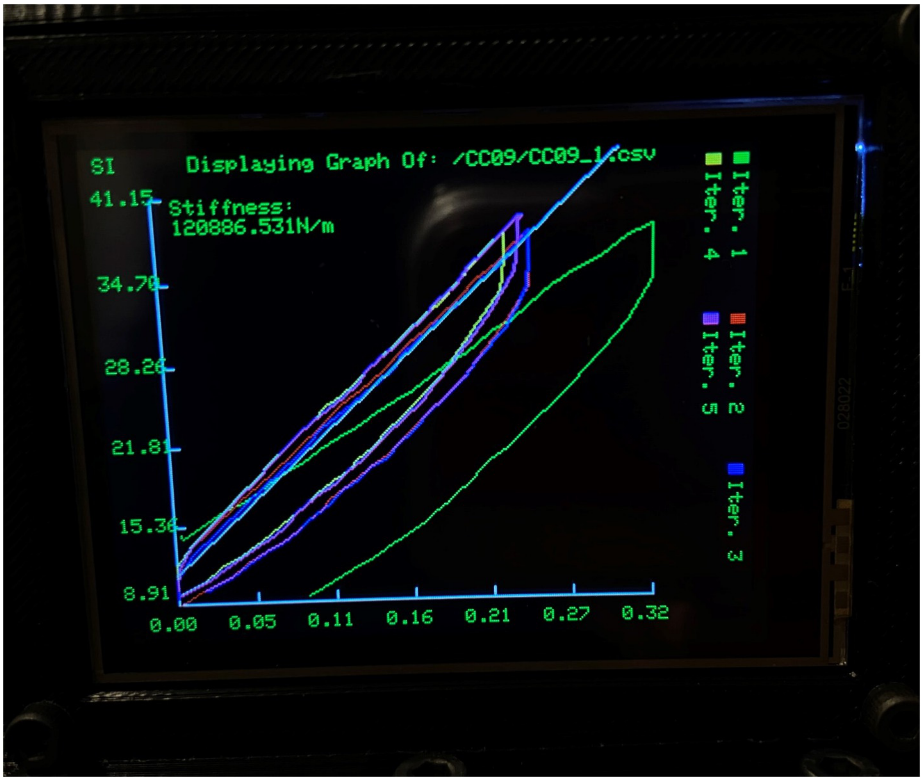


Fig. 75. An example of a post-test data graph.

enter a stalk name as seen in Fig. 72. If the Linear Potentiometer is uncalibrated or the microSD Card Reader is not initialized, the Crop Clamp will not allow the user to access the File Read option.

Due to Arduino constraints, the file name is limited to 5 characters. Use the UP, DOWN, RIGHT, LEFT, and SELECT buttons to choose a name. The backspace is the arrow on the bottom right.

Press the OKAY button to move on with the test. If the entered stalk name already exists on the microSD card the user will be prompted to re-enter the name.

When a valid name is entered, the user will be met with the screen in Fig. 73.

This screen will be displayed just before the Crop Clamp takes a transverse stiffness measurement. A maize internode should be placed between the Load Cell and the Metal Backplate, touching the Metal Backplate. Use the LEFT and RIGHT buttons to move the Load Cell to within half an inch of the stalk. When everything is positioned as in Fig. 74, press the SELECT button to begin the test.

When a test has begun, it cannot be stopped via the software. If the user needs to stop the test, then the device will have to be unplugged (this will not save the data from the current test, but it will not harm the device in any way). The Crop Clamp is designed to cycle between 10 and 40 N of force while recording the force and displacement data to the microSD card. The number of cycles can be adjusted on the Settings page.

When the device has completed a test, a graph of the data will appear as seen in Fig. 75.

In the graph, the tests are color-coded by cycle, and both the calculated transverse stiffness and file name are displayed. When the user has examined the data, they can press BACK to move on.

The user will then be prompted to Measure Another Internode, End Internode Collection, or Remeasure Internode. The user can measure up to 10 internodes for a single stalk. These data files are saved as [StalkName]_[InternodeNumber].csv files on the microSD card in the StalkName folder.

The remeasure internode option should be used if the user is dissatisfied with the data graph. This will delete the previous test and return the user to the pre-test screen. This is only way to delete unsatisfactory tests directly from the Crop Clamp.

When all internodes have been measured, the End Internode Collection option will return the user to the Home Screen (or the user will automatically be returned to the Home Screen after 10 internodes have been tested).

Device safety

User safety should always be considered when operating the Crop Clamp. The Stepper Motor used to apply compressive loads can become hot during operation. Although the Stepper Motor is not hot enough to cause permanent harm the user, contact with the stepper motor can be painful and should be avoided whenever the device is turned on. A label can be added to the device to indicate this.

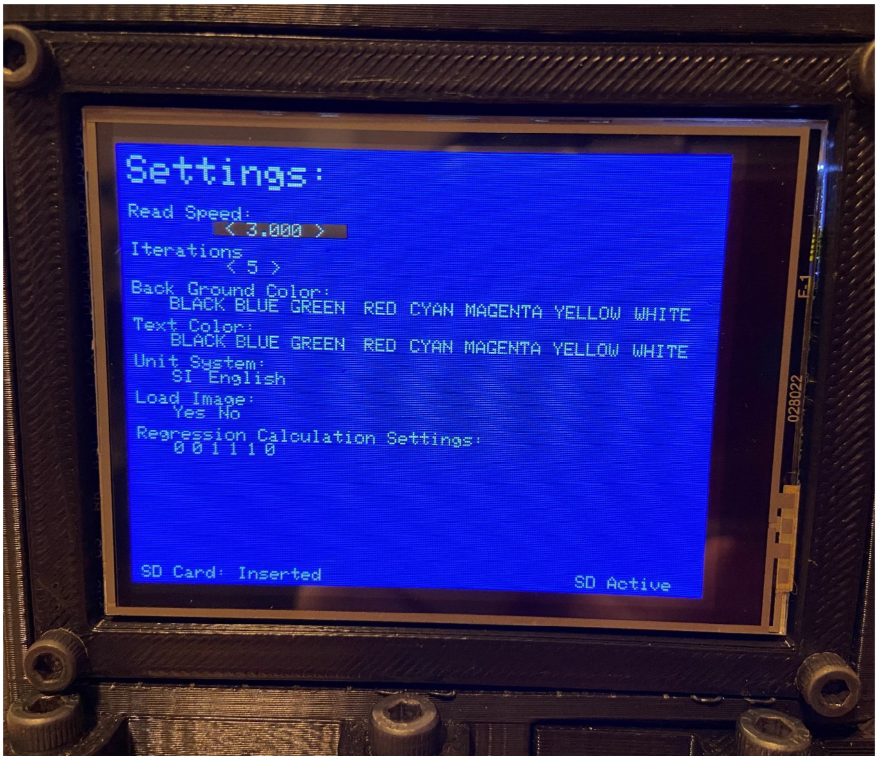
During compressive testing, it is not possible to stop a test using the software. Should a user pinch their finger or hand during a test, proper procedure is to unplug the device. The device will not continue to apply compressive forces after being unplugged, and the lead screw can *easily* be turned by hand to remove a trapped hand or finger. Unplugging the device will result in the loss of the current test data but cause no lasting damage to the device. It should also be noted that the crop clamp was intentionally designed not to apply a force that could seriously injure a user. Consequently, the device is not physically capable of applying more than 30 lbs of pressure. This is not enough to cause serious injury to an adult and the authors have pinched their fingers with the device to ensure this. However, users should always be careful to avoid pinching their hands or fingers during operation.

Settings

When Settings is selected from the Home Screen, the screen from Fig. 76 will appear.

Use the UP, DOWN, LEFT, and RIGHT buttons to navigate through the Settings page. For all the settings except Regression Calculation Settings, the SELECT button does not have to be used when changing settings.

- Read Speed: This is the speed at which the Load Cell moves in and out during pre and post-test movements (in mm/s). The default of 3.00 mm/s works well, but it can be adjusted in increments of 0.125 using the RIGHT and LEFT buttons.
- Iterations: This is how many times the Load Cell will go in and out during a test (i.e. how many times the stalk is compressed). The range for this setting is 1–6. Using 3 or more iterations is recommended [31].
- Back Ground Color: This is an aesthetic setting. The background color can be changed using the RIGHT and LEFT buttons. The device will not allow the same background and text colors.
- Text Color: Identical to background color but for the text.
- Unit System: This will allow the user to change the unit system of the device. The Crop Clamp can use SI or English units.
- Load Image: If the logo.bmp file is loaded onto the Crop Clamp's microSD Card, this will allow the user to display the AgMeq logo for 2 s upon bootup. Any 240x320 bmp image with the name logo.bmp loaded onto the microSD Card will be displayed.



```
Settings:
Read Speed:
      < 3.000 >
Iterations
      < 5 >
Back Ground Color:
   BLACK BLUE GREEN  RED CYAN MAGENTA YELLOW WHITE
Text Color:
   BLACK BLUE GREEN  RED CYAN MAGENTA YELLOW WHITE
Unit System:
   SI  English
Load Image:
   Yes  No
Regression Calculation Settings:
   0 0 1 1 1 0

SD Card: Inserted                      SD Active
```

Fig. 76. The Settings screen.

- Regression Calculation Settings: On this row of settings there are 6 numbers (of 1 s or 0 s). The numbers represent whether a test iteration is included in the linear regression calculations for transverse stiffness. Navigate through the numbers using the LEFT and RIGHT buttons. If the SELECT button is pressed while a number is highlighted, that number will change from a 0 to a 1, or from a 1 to a 0. A 1 means that iteration will be included in the calculation. A 0 means that the iteration will not be included in the calculation.
  - For example, if the numbers are 0 1 1 1 1 1, then iterations 2–6 will be used when calculating transverse stiffness.
  - Another example, if the numbers are 0 1 0 1 0 1, then iterations 2, 4, and 6 will be used when calculating transverse stiffness.
  - An important note on the regression settings: If the number of iterations is set to 5, then only the first 5 numbers of the Regression Calculation Setting are used when calculating transverse stiffness. If only 3 iterations are used, only the first 3 numbers will be used... etc.
  - It is not recommended to use the 1st iteration when calculating transverse stiffness [31].

The settings are saved to the microSD Card when the user presses the BACK button to return to the Home Page. Each time the device is powered up, the settings are read in from the inserted microSD card. If no microSD card is inserted or there is no settings.txt file stored on the SD card, the device will revert to its default settings.

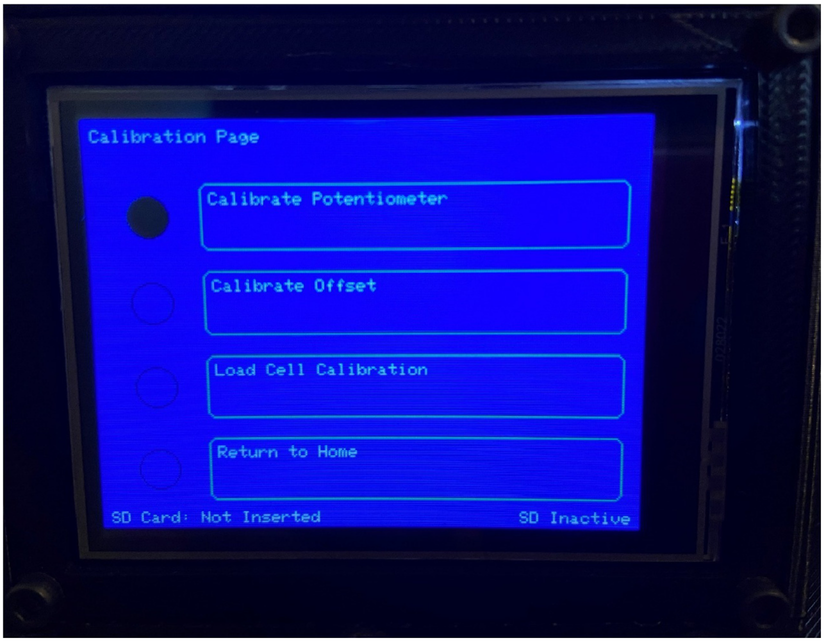
Reboot SD

When this is selected from the Home Screen, the device will reinitialize the microSD Card. If there is a problem initializing the microSD Card, an error page will appear prompting the user to investigate further.

Calibrate

When the Calibrate option is selected from the Home Screen, the user will be met with the screen in Fig. 77.

From here, the user can choose to Calibrate Potentiometer, Calibrate Offset, perform Load Cell Calibration, or return to the Home Screen. Calibrate Potentiometer calibrates the Linear Potentiometer and accounts for voltage differences in the 9 V battery. Calibrate Offset accounts for the device's natural offset and must be done when the device is turned on for the first time. Load Cell Calibration allows the user to calculate the calibration value for the Load Cell; this must also be performed when the device is first powered on. More details on all the options are below.


Calibration Page

SD Card: Not Inserted SD Inactive

Fig. 77. The Calibration Screen.

1. **Calibrate Potentiometer**

When Calibrate Potentiometer is pressed, the screen will tell the user to retract device and use SELECT to calibrate the Linear Potentiometer. While on this screen, the user can move the Load Cell forwards and backwards using the RIGHT and LEFT buttons. Before pressing SELECT, the edge of the Load Cell must be at least 3 cm from the Metal Backplate. When SELECT is pressed, the device will move forward approximately 2.9 cm and then display the calibrated Linear Potentiometer value. If the number is between 70 and 120, the user does not need to do anything. If the number if lower or higher than these bounds, double-check the 9 V battery connection. The Linear Potentiometer needs to be calibrated every time the device is powered on, so it may be useful to mark where the 3 cm point is on the 8020 Beam for future reference.

2. Calibrate offset

When Calibrate Offset is pressed, the screen will tell the user to move the device forward and press SELECT to calibrate the Crop Clamp's natural offset. Before doing this, find a small, solid metal rod and place it between the edge of the Load Cell and the Metal Backplate (touching the Metal Backplate) as seen in Fig. 78. Then use the RIGHT and LEFT buttons to move the load cell to within a couple millimeters of the metal rod and press the SELECT button. The device will then compress the metal rod 5 times and display the device's natural offset. It will take nearly a minute to calculate the natural offset, and the metal rod will have to be held in place throughout the duration of the calibration. Note the natural offset (it should be between 0 and 50) and open the CropClamp_OS.ino Arduino sketch and find the "STEP_OFFSET" variable on line 60. Replace the value there with the calculated natural offset. This number represents how much the device flexes during a transverse stiffness test, and it is crucial that this calibration be run before using the Crop Clamp for scientific tests. When the variable value is updated, reupload the CropClamp_OS.ino sketch to the Crop Clamp. The point of this calibration is to find out how much the device flexes when a maize stalk is compressed. The calculated natural offset value will be subtracted from the measured distance during a transverse stiffness test, thereby increasing the device's accuracy. It is recommended to perform this calibration every five-hundred tests performed on the Crop Clamp.

3. **Load Cell Calibration**

When Load Cell Calibration is selected from the Calibration Screen, the user will be taken to the screen shown in Fig. 79.

From this screen, the user can determine the calibration factor for the Load Cell. The basic idea is to apply a known force to the Load Cell and manually adjust the calibration factor until the device is displaying the force's true value. The user can adjust the calibration factor using the D-Pad buttons. Pressing the LEFT, DOWN, SELECT, UP, and RIGHT buttons will change the calibration factor by 1, 10, 100, 1000, and 10000, respectively. The OKAY button changes the sign of the additions to the calibration factor, and the current sign of the modifiers is specified after the word 'Direction' (it is positive in Fig. 79). When

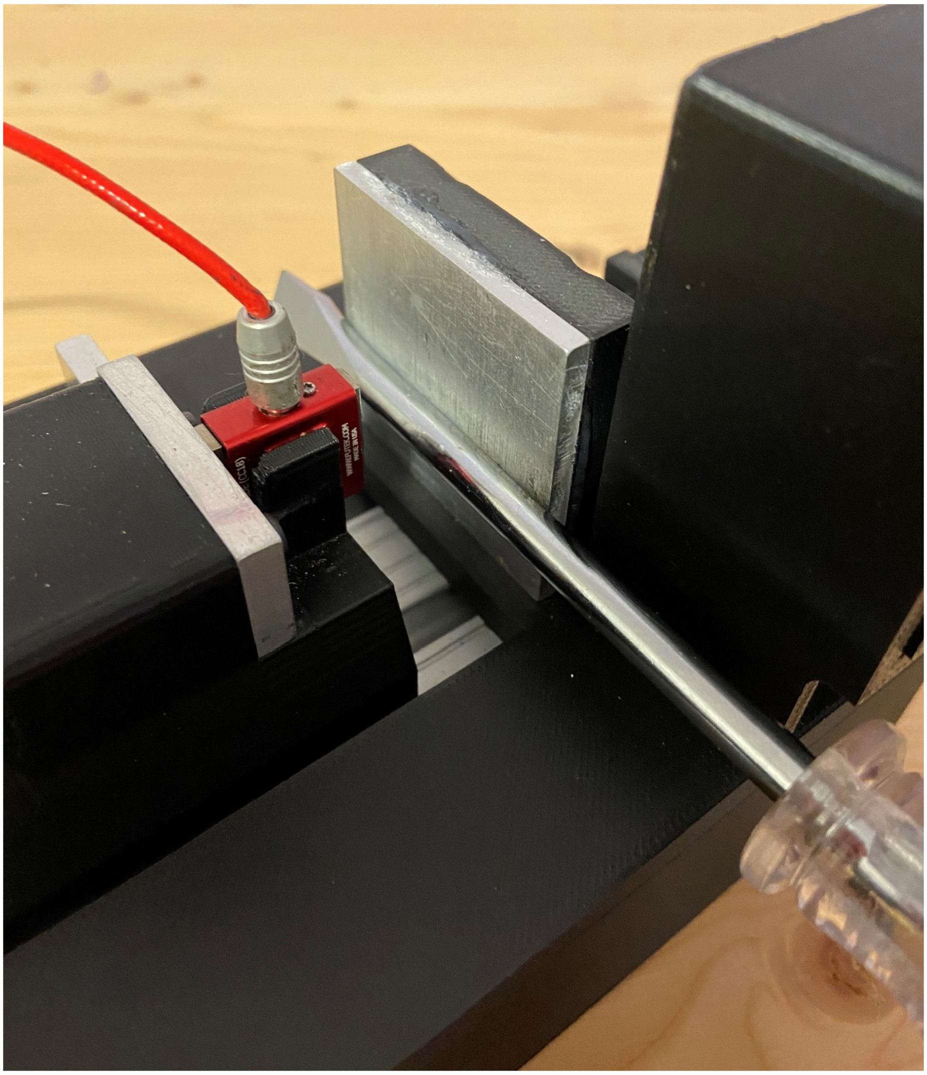


Fig. 78. A solid metal rod (in this case, a screwdriver) position for the Device Offset calibration.

the user has adjusted the calibration factor until the Force reading matches the applied force (the reading is in N), write down the calibration factor. Then, the CropClamp_OS.ino sketch must be opened in the Arduino IDE, and the variable "calibration_factor" on line 65 must be updated to this new value.

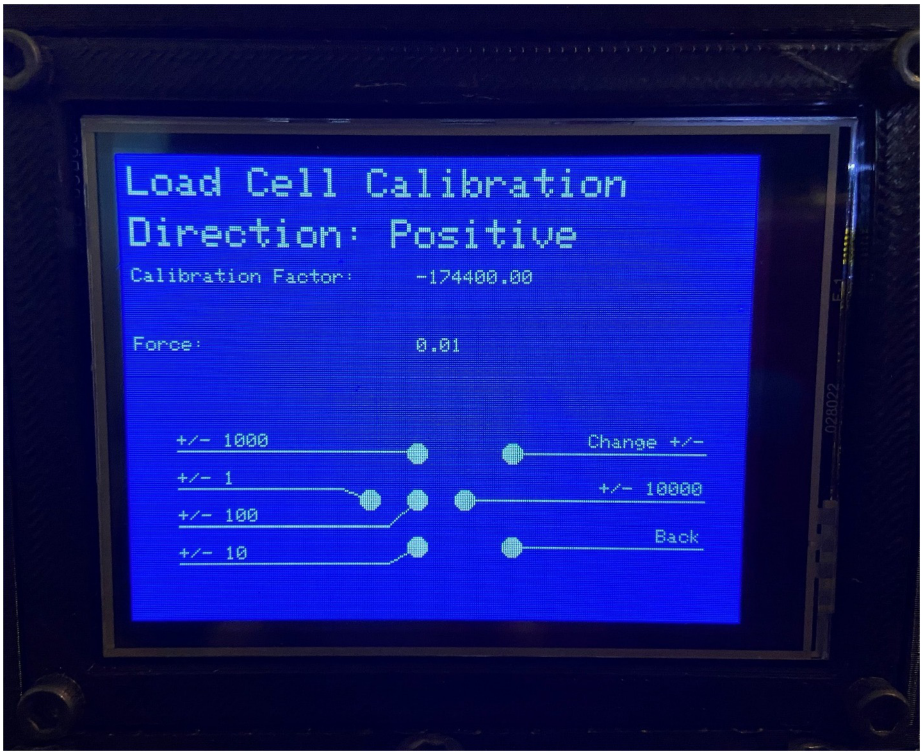
The most efficient way to apply a known force to the Load Cell is to remove the Load Cell Backplate (MILL01) from the Load Cell Carriage and hang a known mass from the Load Cell using the Calibration Piece (CAD11). The Load Cell Backplate (with the Load Cell still attached) should be clamped to the edge of a table facing downward as seen in Fig. 80.

At this point, when the user goes to the Load Cell Calibration page, the Force should read zero. If the device is reading a non-zero value, the user can tare (zero) the Load Cell in the Raw Measure page accessible from the Home Screen. When the Force reading is zero, take the Calibration Piece, attach it to the Load Cell using a 3 M × 0.5 8 mm bolt and a 3 M × 0.5 nut (the nut acts as a space, as seen in Fig. 81). Then, a known mass should be hung from the Calibration Piece as seen in Fig. 81.

Both the calibration piece and the mass should be weighed on an accurate scale before attaching them to the Load Cell. At this point the user may use the D-Pad scheme specified in Fig. 79 to change the calibration factor until the Force displayed on the device matches the known weight of the mass plus the calibration piece. The Force displayed on the device is in Newtons.

Examine Data

When Examine Data is selected from the Home Screen, the user can scroll through the stalk measurements that are on the microSD card. Six folders will be displayed at once (as seen in Fig. 82) and the RIGHT and LEFT buttons can be used to access more tests. Choose a test to examine using the SELECT button.



Load Cell Calibration
Direction: Positive
Calibration Factor: -174400.00

Force: 0.01

Fig. 79. The Load Cell Calibration Screen.

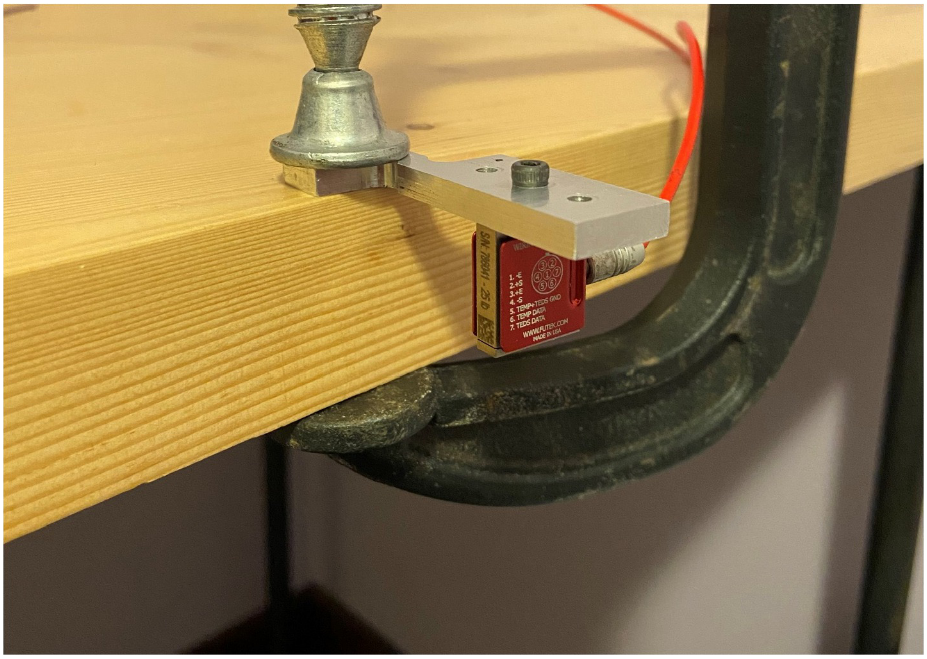


Fig. 80. The Load Cell Backplate has been clamped to the edge of a table for calibration.

When a folder is selected the user must choose which internode measurement to examine. Use the RIGHT and LEFT buttons to scroll through the available internodes and press SELECT to choose an internode.

When an internode is chosen, the data will be graphed and the transverse stiffness will be calculated based on the device's current regression settings (which may be different than the settings when the measurement was taken).

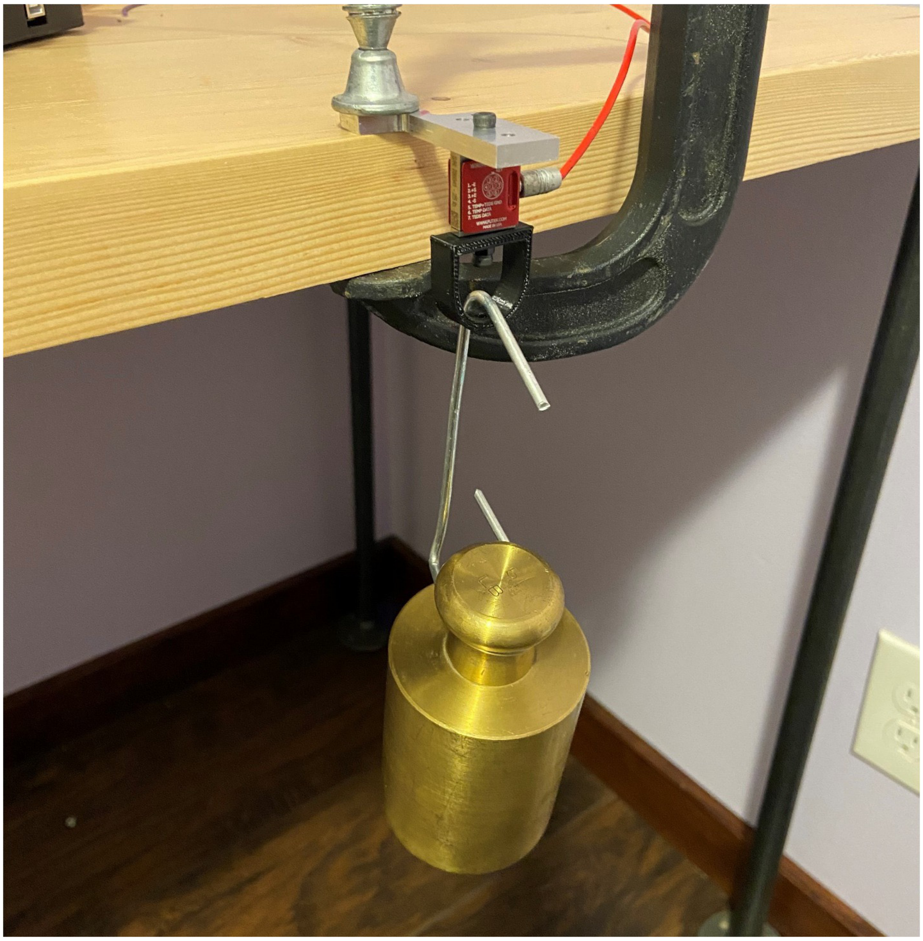


Fig. 81. A mass hanging from the Calibration Piece.

This page behaves exactly like a post-test data page, except that the transverse stiffness can be manually appended to the data file if the UP button is pressed. This is useful if the user wishes to calculate and store several transverse stiffness using different regression settings.

The internal battery of the Crop Clamp can be charged with the Rev Robotics Battery Charger.

Validation and characterization

Validation of the Crop Clamp was performed by comparing transverse stiffness data taken on the Crop Clamp to data on the same stalks gathered on an Instron Universal Testing Machine. The full validation method is outlined below.

Validation specimens

Fully mature and dry maize stalks were used for this validation study. The stalks were removed from the field at harvest time by cutting them just above the ground and just below the ear. The stalks were then placed in a forced-air dryers to reduce moisture content to a stable level to enable storage of stalk samples without molding. Stalks were stored at 20 bC and 15% humidity until testing. All stalks used in this study were free of disease.

Universal Testing Machine equipment and procedure

Compression testing was performed on the maize internodes using a Universal Testing Machine (Instron 5944, Instron Corp., Norwood, MA, USA). Loadings were measured using a 2 kN Instron Load Cell. The contact surface was a custom-

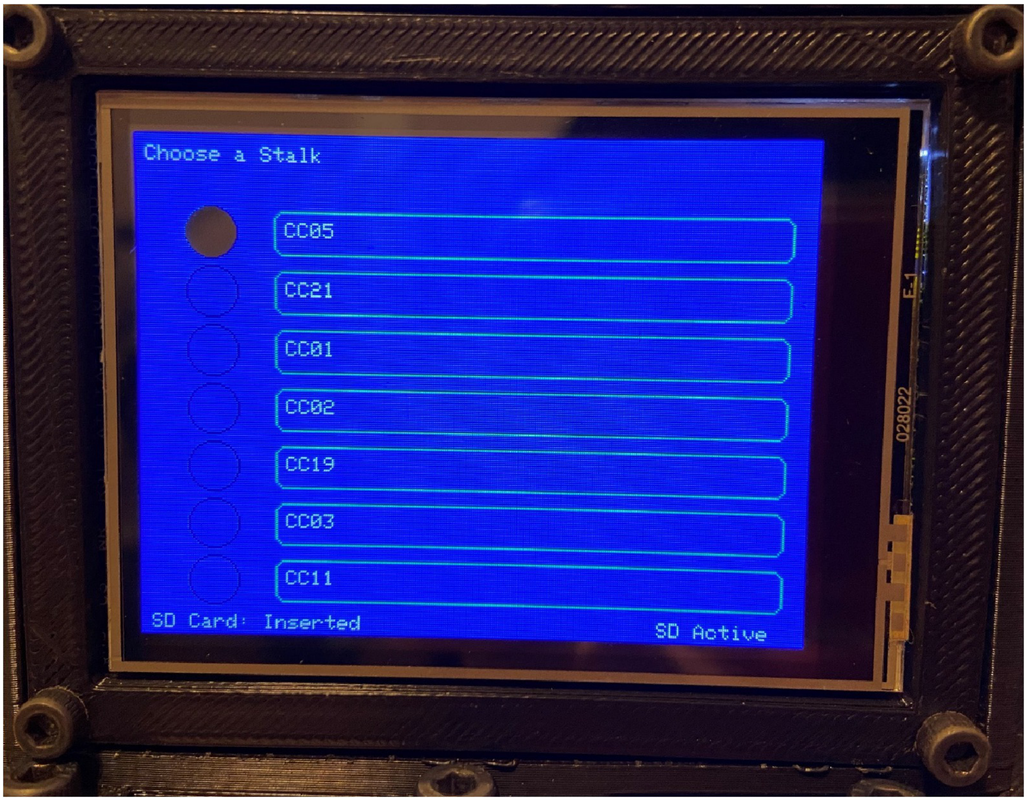


Fig. 82. Sample data folders.

machine aluminum plate matching the geometry of the FUTEK Miniature S-Beam Jr. Load Cell 2.0 (the contact surface used on the Crop Clamp). Test controls and data acquisition were handled by the Instron Bluehill 3.0 software.

For testing, a 10 N preload was applied to the maize internodes before cycling back and forth between the 10 N preload and a maximum value of 40 N. This preload increases the accuracy of measurements in biological tissues [31]. Three cycles were performed, with the first two being conditional cycles. Only the third cycle's data was used to calculate the transverse stiffness. For all tests, the displacement rate of the load cell was 10.0 mm/min and the data collection rate was 50 Hz. A total of 40 maize stalks were tested in this fashion comprising 202 internodes.

Crop Clamp equipment and procedure

After the tests were performed on the Universal Testing Machine, the same stalks were tested on the Crop Clamp. Loading was measured using the Crop Clamp's FUTEK Miniature S-Beam Jr. Load Cell 2.0. The bare end of the Load Cell was the contact surface. Test controls and data acquisition were handled by the custom operating software on the Crop Clamp's Arduino Mega. The testing procedure is the same as the process outlined in the Operating Instructions section, with the first cycle used as a preconditioning cycle and the last four cycles being used to find an average transverse stiffness.

Validation results

A linear regression was performed on the data from the Instron and Crop Clamp. For the regression, the Instron's data was treated as the independent variable while the Crop Clamp's data was the dependent variable. The slope of the regression was 0.954 and the R^2 value for the regression was 0.827 (p-value < 1e-16). The statistically significant correlation between data from the Instron and the Crop Clamp can be seen in Fig. 83, where the regression line can be seen in red with a blue 1:1 reference line.

Limitations

- The Crop Clamp was measured to be accurate to within +/- ~6% of Instron-calculated values. The accuracy could likely be improved by calibrating the Crop Clamp's natural offset and the Load Cell calibration factor more often.

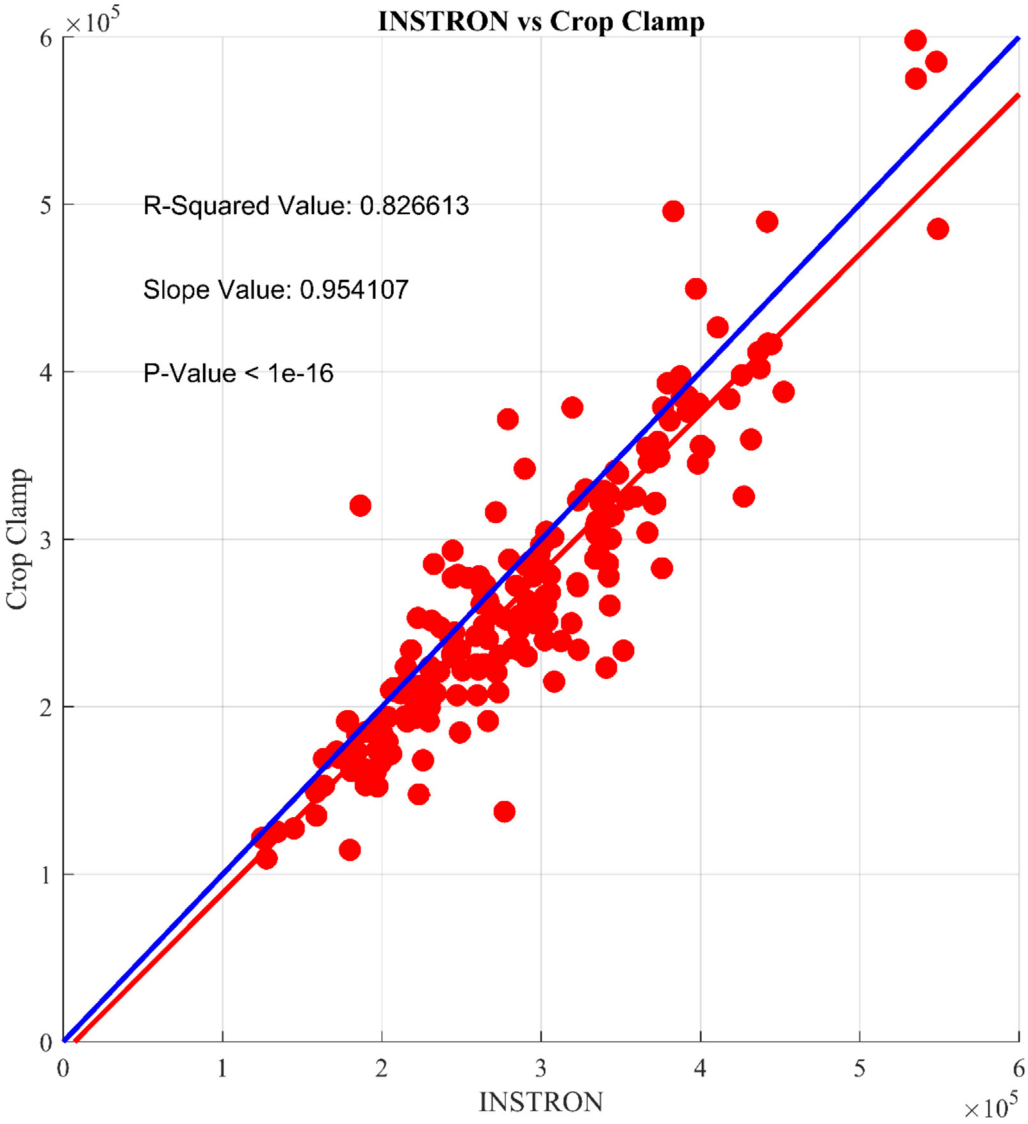


Fig. 83. Measurements for the Instron Universal Testing System are on the x-axis. The measurements for the Crop Clamp are on the y-axis. All units are Newtons per meter.

- In this study, the Crop Clamp was only validated against dry maize stalks. In the future there is also potential for the device to be validated on other crops like sorghum and bamboo.
- The Crop Clamp's battery capacity drops during normal operation. After a year of heavy use, battery capacity will drop from 3 h to roughly 1 h. Replacing the battery every six months to a year will improve the operating life of the Crop Clamp.

Future improvements

There are two notable improvements that the developers of the Crop Clamp would like to pursue in future device iterations. First, the developers would like to refine the device's ergonomics for handheld use. While the Crop Clamp could technically be used as a handheld device, it is much easier to operate when placed on a tabletop. The developers would like to add a shoulder strap and handles, change the position of the screen, and improve wire management to encourage handheld operation.

The second major improvement the developer would like to pursue is the transition from a breadboard-based electronics system to a custom printed circuit board (PCB) setup. For initial device development, it was not feasible to pursue PCB development due to lack of resources. However, switching to a PCB system for future device iterations would simplify device assembly improve device reliability. Any future improvements to the device will be posted to the data repository which accompanies this paper.

CRediT authorship contribution statement

Andrew M. Stucker: Methodology, Software, Validation, Formal analysis, Writing – original draft. **Ethan Morris:** Software, Validation, Writing - review & editing. **Christopher J. Stubbs:** Writing - review & editing, Supervision. **Daniel J. Robertson:** Conceptualization, Funding acquisition, Supervision, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The author would like to thank Taylor Spence and Ankit Gupta for their technical advice throughout the development of the Crop Clamp; Dr. Brent Stucker for his technical writing input and Andrew Pilchard for his feasibility review. Funding for this work was provided by the National Science Foundation EPSCOR Award #1826715. Any opinions, findings, conclusions, or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the view of the National Science Foundation.

References

[1] S.A. Flint-Garcia, C. Jampatong, L.L. Darrah, M.D. McMullen, Quantitative trait locus analysis of stalk strength in four maize populations, Crop Sci. 43 (2003) 13–22, https://doi.org/10.2135/cropsci2003.0013.

[2] J.M. Arnold, L.M. Josephson, Inheritance of stalk quality characteristics in maize 1, Crop Sci. 15 (1975) 338–340.

[3] R.S. Sekhon, C.N. Joyner, A.J. Ackerman, C.S. McMahan, D.D. Cook, D.J. Robertson, Stalk bending strength is strongly associated with maize stalk lodging incidence across multiple environments, Field Crops Res. 249 (2020) 107737.

[4] J. Abadassi, Maize agronomic traits needed in tropical zone, Int. J. Sci. Environ. Technol. 4 (2015) 371–392.

[5] R. Kumar, A. Gyawali, G.D. Morrison, C.A. Saski, D.J. Robertson, D.D. Cook, N. Tharayil, R.J. Schaefer, T.M. Beissinger, R.S. Sekhon, Genetic architecture of maize rind strength revealed by the analysis of divergently selected populations, Plant Cell Physiol. (2021), https://doi.org/10.1093/pcp/pcab059.

[6] F.D. Cloninger, Methods of evaluating stalk quality in corn, Phytopathology. 60 (1970) 295, https://doi.org/10.1094/Phyto-60-295.

[7] L.A. Al-Zube, D.J. Robertson, J.N. Edwards, W.H. Sun, D.D. Cook, Measuring the compressive modulus of elasticity of pith-filled plant stems, Plant Methods. 13 (2017). ARTN 99 10.1186/s13007-017-0250-y.

[8] C.J. Stubbs, C. McMahan, W. Seegmiller, D.D. Cook, D.J. Robertson, Integrated Puncture Score: Force-Displacement Weighted Rind Penetration Tests Improve Stalk Lodging Resistance Estimations in Maize, (2020).

[9] D.J. Robertson, S.Y. Lee, M. Julias, D.D. Cook, Maize stalk lodging: flexural stiffness predicts strength, Crop Sci. 56 (2016) 1711, https://doi.org/10.2135/cropsci2015.11.0665.

[10] D.J. Robertson, M. Julias, B.W. Gardunia, T. Barten, D.D. Cook, Corn stalk lodging: a forensic engineering approach provides insights into failure patterns and mechanisms, Crop Sci. 55 (2015) 2833, https://doi.org/10.2135/cropsci2015.01.0010.

[11] C.J. Stubbs, Y.A. Oduntan, T.R. Keep, S.D. Noble, D.J. Robertson, The effect of plant weight on estimations of stalk lodging resistance, Plant Methods 16 (1) (2020), https://doi.org/10.1186/s13007-020-00670-w.

[12] C.J. Stubbs, K. Seegmiller, C. McMahan, R.S. Sekhon, D.J. Robertson, Diverse maize hybrids are structurally inefficient at resisting wind induced bending forces that cause stalk lodging, Plant Methods 16 (2020) 1–15.

[13] D. Robertson, J. Cornwall, C. Stubbs, McMahan Christopher, The overlooked biomechanical role of the clasping leaf sheath in wheat stalk lodging, Front. Plant Sci. (n.d.). doi: 10.3389/fpls.2021.617880.

[14] D.J. Robertson, M. Julias, S.Y. Lee, D.D. Cook, Maize stalk lodging: morphological determinants of stalk strength, Crop Sci. 57 (2017) 926, https://doi.org/10.2135/cropsci2016.07.0569.

[15] G. Von Forell, D. Robertson, S.Y. Lee, D.D. Cook, Preventing lodging in bioenergy crops: a biomechanical analysis of maize stalks suggests a new approach, J. Exp. Bot. 66 (2015) 4367–4371, https://doi.org/10.1093/jxb/erv108.

[16] D.J. Robertson, S.L. Smith, D.D. Cook, On measuring the bending strength of septate grass stems, Am. J. Bot. 102 (1) (2015) 5–11, https://doi.org/10.3732/ajb.1400183.

[17] D. Robertson, S. Smith, B. Gardunia, D. Cook, An improved method for accurate phenotyping of corn stalk strength, Crop Sci. 54 (2014) 2038, https://doi.org/10.2135/cropsci2013.11.0794.

[18] D.J. Robertson, D.D. Cook, K. Meehan, L. Asatiani, The Effect of Probe Geometry on Rind Puncture Resistance Testing of Maize Stalks, (2020).

[19] D. Jo Heuschele, J. Wiersma, L. Reynolds, A. Mangin, Y. Lawley, P. Marchetto, The Stalker: An open source force meter for rapid stalk strength phenotyping, HardwareX 6 (2019) e00067, https://doi.org/10.1016/j.ohx.2019.e00067.

[20] W. Seegmiller, J. Graves, D. Robertson, A Novel Rind Puncture Technique to Measure Rind Thickness and Diameter in Plant Stalks, Preprint. (2020). 10.21203/rs.2.19825/v1.

[21] D.D. Cook, W. de la Chapelle, T.-C. Lin, S.Y. Lee, W. Sun, D.J. Robertson, DARLING: a device for assessing resistance to lodging in grain crops, Plant Methods 15 (2019) 102.

[22] A. Bebee, C.J. Stubbs, D.J. Robertson, Large deflection model for multiple, inline, interacting cantilever beams, J. Appl. Mech. 88 (2021), https://doi.org/10.1115/1.4049072 041005.

[23] L. Erndwein, D.D. Cook, D.J. Robertson, E.E. Sparks, Field-based mechanical phenotyping of cereal crops to assess lodging resistance, Appl. Plant Sci. 8 (8) (2020), https://doi.org/10.1002/aps3.v8.810.1002/aps3.11382.

[24] C.J. Stubbs, W. Sun, D.D. Cook, Measuring the transverse Young's modulus of maize rind and pith tissues, J. Biomech. 84 (2019) 113–120.

[25] T.A. Jackson-Ziems, J.M. Rees, R.M. Harveson, Common stalk rot diseases of corn, (2014).

[26] Agronomy Update, Beck's Hybrids. (n.d.). http://www.beckshybrids.com/Blog/ArtMID/841/ArticleID/162/Agronomy-EIndiana-and-Ohio-Check-for-Stalk-Lodging (accessed February 4, 2021).

[27] How To: Push & Pinch Test, (n.d.). https://www.channel.com/en-us/about/digital-on-point/how-to-push-pinch-test.html (accessed February 4, 2021).

[28] N.S. Inc, Pinch and Push Test in Corn, (n.d.). https://www.n2seed.com/n2-seed-blog/pinch-and-push-test-in-corn (accessed February 4, 2021).

[29] Pinch or Push Your Corn: Scouting for Lodging Potential: University of Illinois Extension, (n.d.). https://extension.illinois.edu/blogs/hill-and-furrow/2016-09-12-pinch-or-push-your-corn-scouting-lodging-potential (accessed February 4, 2021).

[30] What Is the Price of a Universal Test Machine?, (n.d.). https://www.testresources.net/blog/what-is-the-price-of-a-universal-test-machine/ (accessed February 8, 2021).

[31] S. Cheng, E.C. Clarke, L.E. Bilston, The effects of preconditioning strain on measured tissue properties, J. Biomech. 42 (2009) 1360–1362, https://doi.org/10.1016/j.jbiomech.2009.03.023.

Daniel Robertson is an assistant professor in Mechanical Engineering and the director of the National Academy of Engineering Grand Challenge Scholars Program for the University of Idaho. He acquired his doctorate in mechanical engineering from Brigham Young University in 2013. After receiving his doctorate, Robertson spent four years working overseas as a senior research scientist in the emerging area of crop biomechanics. His unique interdisciplinary research enabled him to work across numerous industries which led to his appointment as a United States Department of Agriculture NIFA-AFRI Fellow. His research eventually led Robertson to return to Idaho where he is working with local agronomists and industry partners to develop solutions to the long standing problem of stalk lodging (i.e., corn, wheat, rice, barley and canola plants breaking in wind and rain storms). In general, Robertson's teaching and research efforts are focused on using engineering principles to accomplish what the White House has called the great challenge of the 21st century: sustainably providing food, fuel and fiber for the world's growing population.