**DHRUV SHARMA - 2581618**

**BACKGROUND**

**Title : Sensor based E-Textile for Safety**

**Part 1 : An Overview of the project space**

The project is aimed at creating an amalgamation of textile and the use of electronics to create intelligent fabric that provides information of state of fabric. The idea of the project for now leans towards Women Safety, with growing cases of sexual harassment/physical abuse and rape this can be a leap in the right direction, making environment aware clothing that measures the metrics of a fabric will/can help someone in despair. Having sensors like stretch sensor/shear sensor/force sensor can help with the data required in an unfortunate event, this data can in-turn set off a relay of several different mechanisms like GPS tracking, notifying the nearest police station and/or sending alerts to close ones. The initial aim of the project can be to demonstrate the first phase of electronics and textile integration where it can be studied if/how can the two elements be made to work together. Apart from women safety this can be applied to ensure an anti-bullying culture where physical bullying can be tracked and penalised. For this project integrating a conductive fabric with an e-circuit to demonstrate the concept is key, collecting data from the sensor whilst being a part of the fabric will be the challenge, actuating based on that data can go beyond the scope of the project.

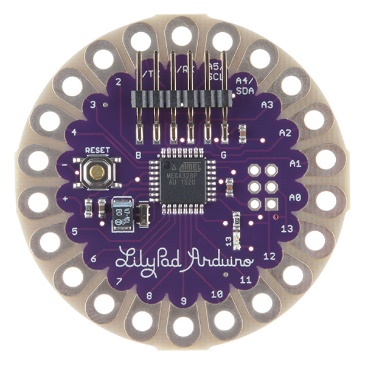
**Part 2 : Related Work**

Three Research papers were used as references that worked in the domain of sensor and fabric integration or creating a sensor based on fabrics. One paper works around Flex sensors, As seen from the idea exploration section one of the ideas chalked out was stretch sensor based which is really close to this research paper that gives details about a system that is meant to detect the lightest muscle activity, any small bend from natural position is measured and reported. This is one paper that holds resemblance in terms of directly embedding the sensor with the user using the fabric as a medium, here the sensor is attached to the upper arm and then deflections are measured, the aim of our project is to have sensors embedded in garments but to detect external factors influencing the sensor/user wearing the garment. The next paper titled ‘Textile pressure sensor made of flexible plastic optical fibres ‘ has a very interesting motive where the team came up with a use of optical fibre technology to measure pressure level measuring the amount of deflection in the path. This would enable us to totally negate any sensor circuits as the fabric itself would give information of state, For our motive this happens to be the perfect product as fabric is inter woven with optical fibres that would give not only pressure information but would also indicate heavy deflection on shearing the fabric, the commercial unavailability of the product makes it tough to acquire. Next we move on to one very relevant Research paper titled ‘A textile-based capacitive pressure sensor’, here the team has come up with a matrix of sensor/capacitor circuits that measure pressure on any point of the surface. The premise is very similar to pressure sensor used in this project. On seeing this one would feel they have the exact same goal but what differs between the two is that the afore-mentioned prototype works as a standalone board that measures vertical pressure from the palm. The aim of our project is integration into everyday garments where Vertical pressure isn’t the only/ideal pressure vector. For our project we take inspiration from one subset of the concept from this research paper.

Technical:

Making use of materials like :

1. Lilypad (arduino)
2. Conductive Fabric
3. Concealable wiring
4. Stretch/shear/force sensor

**IDEA EXPLORATION**

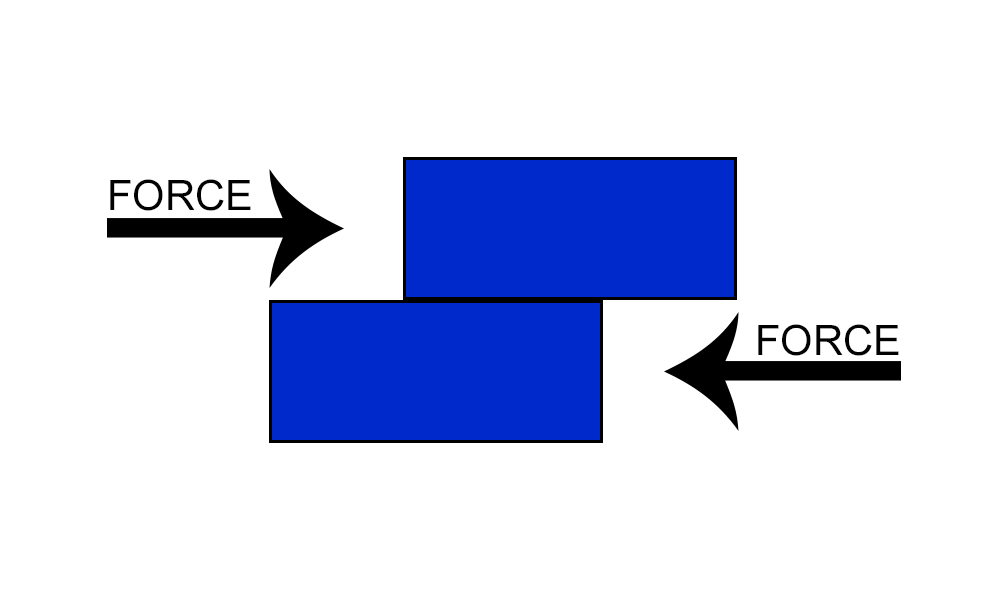
The background gives an idea of where the project is leaning, three ideas for one project emerge from the background description given above.

1. Stretch Sensor

Stretch sensor integrated with a fabric that gives the data about the amount of pull applied of the fabric, in-case of physical or sexual abuse where a piece of clothing might be tugged at forcefully in a bid to remove or initiate an altercation this aware piece of clothing might just provide the much needed last minute assistance.

1. Shear Sensor

Shear sensor too is applied within the same direction and can be used in tandem with the stretch sensor to not only measure pull but also the amount of shearing in a fabric to ensure that some misinterpreted data from the stretch sensor can be removed. The image below might give an idea about shearing.



1. Force sensor

Force sensor applied on conductive fabric whose sensitivity can be set can measure a minor force like 10mg right up until 100kgs of force applied on the sensor, this finds its application in Anti-bullying/Anti-sexual harassment causes where an inappropriate touch can be sensed and an output relay can be an active circuit that might shock the person who touched the surface.

1. Stroke Sensor

One very interesting application of a sensor inspired by the examples in the examples section is in the safety inspired theme just like the ones above, stroke sensor to sense strokes in inappropriate areas to detect touch without consent. This finds various applications in anti-sexual harassment, anti-pedophilistic acts etc.

COMMONALITY:

 All three projects make use of a conductive fabric, an Arduino lilypad and a sensor to measure.

All four project ideas root from one sense or one theme, the aim objective of all remains the same, the things they measure and the way they do it varies.

LINK TO WIKI : <https://hci-lecture.cs.uni-saarland.de/mod/wiki/view.php?pageid=200>

**SKETCHES & IMPLEMENTATION INFORMATION**

All three projects use the components stated above in the background and Idea-Exploration sections, the hardware requirements for the projects remain the same. As far as software goes, all projects use Arduino Lilypad that can be programmed using the ArduinoIDE.

Here are some links to locally source all afore-mentioned material:

1. <https://robu.in/product/force-sensor-5-08mm-circle/>
2. <https://robu.in/product/force-sensor-resistor-square-38-1mm-pressure-sensor/>
3. <https://robu.in/product/flex-sensor-2-2-bend-sensor-hand-gesture-recognition/>
4. <https://robu.in/product/lilypad-328-main-board-atmega328p-atmega328-16m-arduino/>
5. <https://robu.in/product/stainless-steel-conductive-thread-wire-for-wearable-lilypad/>
6. <https://www.fabtolab.com/ada-woven-conductive-fabric>

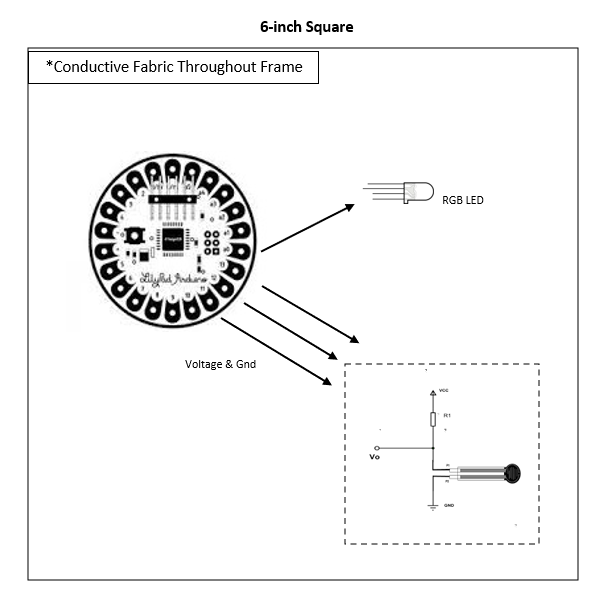
**Sketch Components:**

1. **The 6-inch box is the 15\*15 cm wooden frame.**
2. **The Bold arrow identifies the connections made from Lilypad to sensor.**
3. **The Dotted lines represent the conductive thread used to stitch components to Fabric.**
4. **The RGB LED is used as an output device to show state/information change from the sensor.**

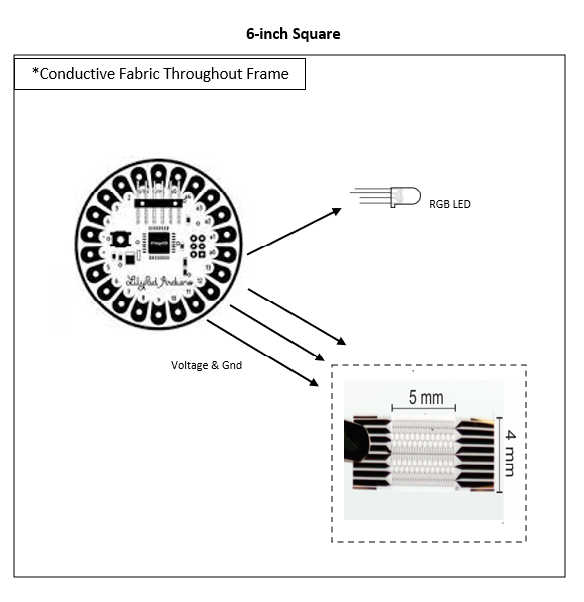
**Eg. Different Pressure levels will have different colors to indicate state change of sensor.**

1. **The Conductive Fabric is omnipresent throughout the frame.**

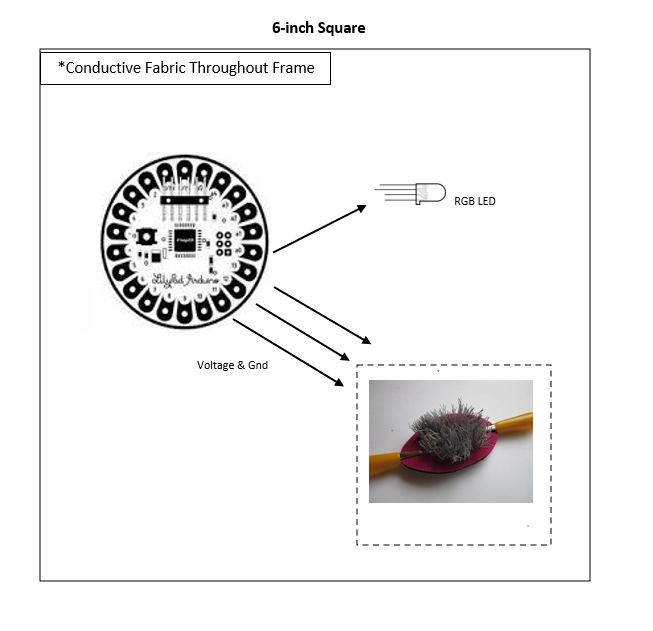
**SKETCH 1 : Force Sensor**



**SKETCH 2 : Stretch Sensor**



**SKETCH 3 : Stroke Sensor**

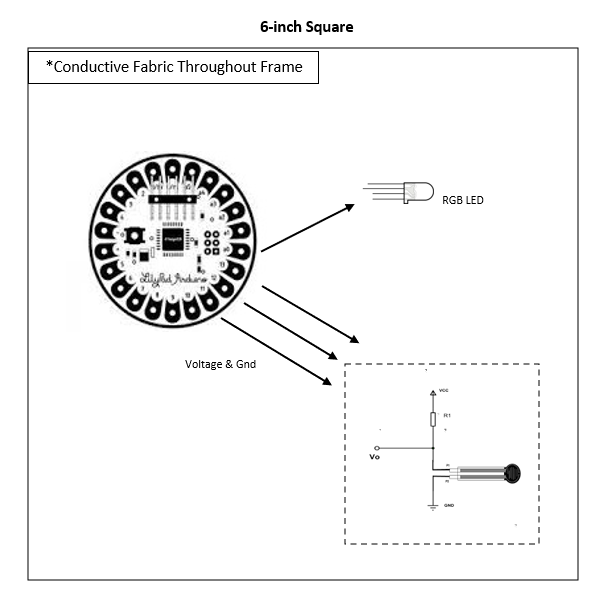


**PROJECT CONCEPT**

**Project Description**

The Concept of the project is to make environment-aware and safe textiles that can be used towards a more secure world. Integration of Fabric and Technology is the key factor in this project, here the aim is to Integrate a **Force/Pressure Sensor** with a Fabric that on sensing external force is able to vaguely indicate the situation the wearer might be in. The focus here is to tweak and play with sensitivity of the sensor and make it usable with Fabrics, this basically is a sensor based project rather than an actuator based project. All components and connections have been briefly described above in the sketches section of the Ideation, now we can look at the overall working and demonstration of the project. The sensor has been stitched to the fabric in a concealed manner, the sensors chosen are deliberately kept thin to promote concealability and to enable minimal interference into a users experience of garments. The connections from ground and Voltage are made to an Arduino Lilypad present within the 15 by 15 demo frame, we use conductive thread to make the connections possible. To indicate state change and change of pressure/force levels from the sensor we make use of either 3 LEDS or a single RGB LED where a specific colour will indicate a pressure level (Just as an example we can take Green-Denotes Light/Low, Blue – Mid range pressure on sensor, Red – High Pressure on Sensor). The actuation that has been left out of the scope of this project can include things like alert to close ones once pressure level enters Red-range, GPS live location sharing via smartphone to selected 3 contacts, In-case things are to be taken to a bit extreme a shock circuit that would act as a deterrent to the person (Apparently adverse).

**Visual Representation:**



(Layout of Gnd and Voltage followed to Sensor and LED – Arrows Used to simplify Representation)

**PROJECT IMPLEMENTATION**

LIST OF MATERIALS: (Self-Sourced due to Geographical Distance from University/Germany)

1. Arduino Lilypad – Wearble Technology – Commercial Component
2. Microcontroller to act as main unit.
3. Best known for Wearable Technology
4. Light Weight and Easy to integrate.
5. Conductive Fabric
6. To enable least resistance in electrical transmission in circuit
7. Directly embeddable with sensor
8. Concealable wiring /Conductive Thread – Locally Sourced
9. In-case of Resistance from Fabric , conductive thread can be used to connect.
10. Removes the use of hard cables to connect
11. Light weight and easy to work with
12. Stretch/shear/force sensor – Locally Sourced
13. Core component of project.
14. Used to detect pressure/force levels
15. RGB LED – Subject to change if better alternative is available
16. Only output mechanism of Project
17. Used to show change in pressure/force levels.
18. Removes the need of separate LEDs.

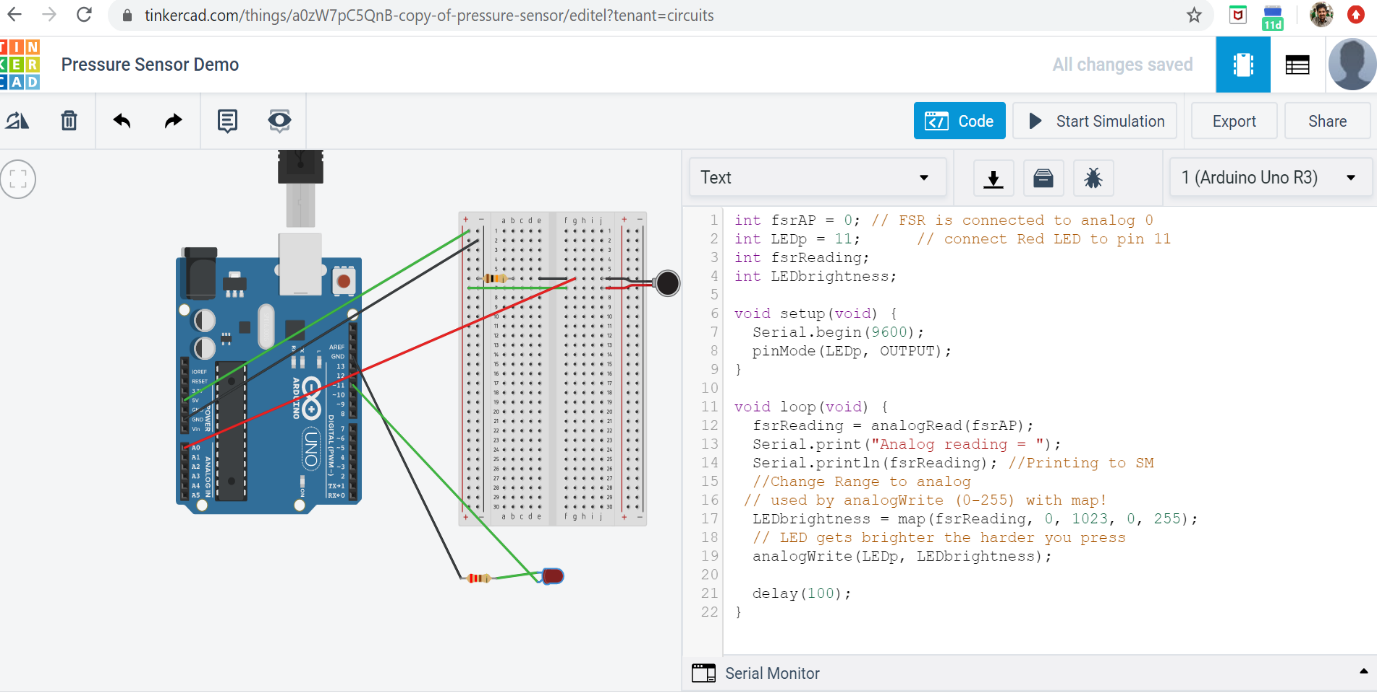
PURCHASE LINKS: (From Present location due to COVID)

1. <https://robu.in/product/force-sensor-5-08mm-circle/>
2. <https://robu.in/product/force-sensor-resistor-square-38-1mm-pressure-sensor/>
3. <https://robu.in/product/lilypad-328-main-board-atmega328p-atmega328-16m-arduino/>
4. <https://robu.in/product/stainless-steel-conductive-thread-wire-for-wearable-lilypad/>
5. <https://www.fabtolab.com/ada-woven-conductive-fabric>
6. <https://robu.in/product/dip-3-color-led-module/>

**SOFTWARE PROTOTYPING**

The first step in prototyping the project would be to get a solid understanding of the workspace digitally, checking the circuit on a CAD software would be the ideal first step to check what will work and how well do components work together. So here I have made an initial draft of the circuit digitally, this will give everyone a fair idea of what the project will do. Here I have used a vibrator motor to emulate a pressure sensor, when pressure on the component increases the LED glows brighter. From this we want to assure feasibility of software and use this as a proof of concept.

**CIRCUIT DESIGN : (TINKERCAD)**



Here I have simulated the intended circuit that works, the LED glows. The pressure part can be seen on the serial monitor as one cant physically apply pressure on a digital render. It was vital to test this virtually, then we move on to physical components and then we integrate with fabric to set things in perspective. Here Lilypad couldn’t be used as it isn’t provided by Tinkercad, the functionality of which will be similar to UNOR3.

CODE :

int fsrAP = 0; // FSR is connected to analog 0

int LEDp = 11; // connect Red LED to pin 11

int fsrReading;

int LEDbrightness;

void setup(void) {

Serial.begin(9600);

pinMode(LEDp, OUTPUT);

}

void loop(void) {

fsrReading = analogRead(fsrAP);

Serial.print("Analog reading = ");

Serial.println(fsrReading); //Printing to SM

//Change Range to analog

// used by analogWrite (0-255) with map!

LEDbrightness = map(fsrReading, 0, 1023, 0, 255);

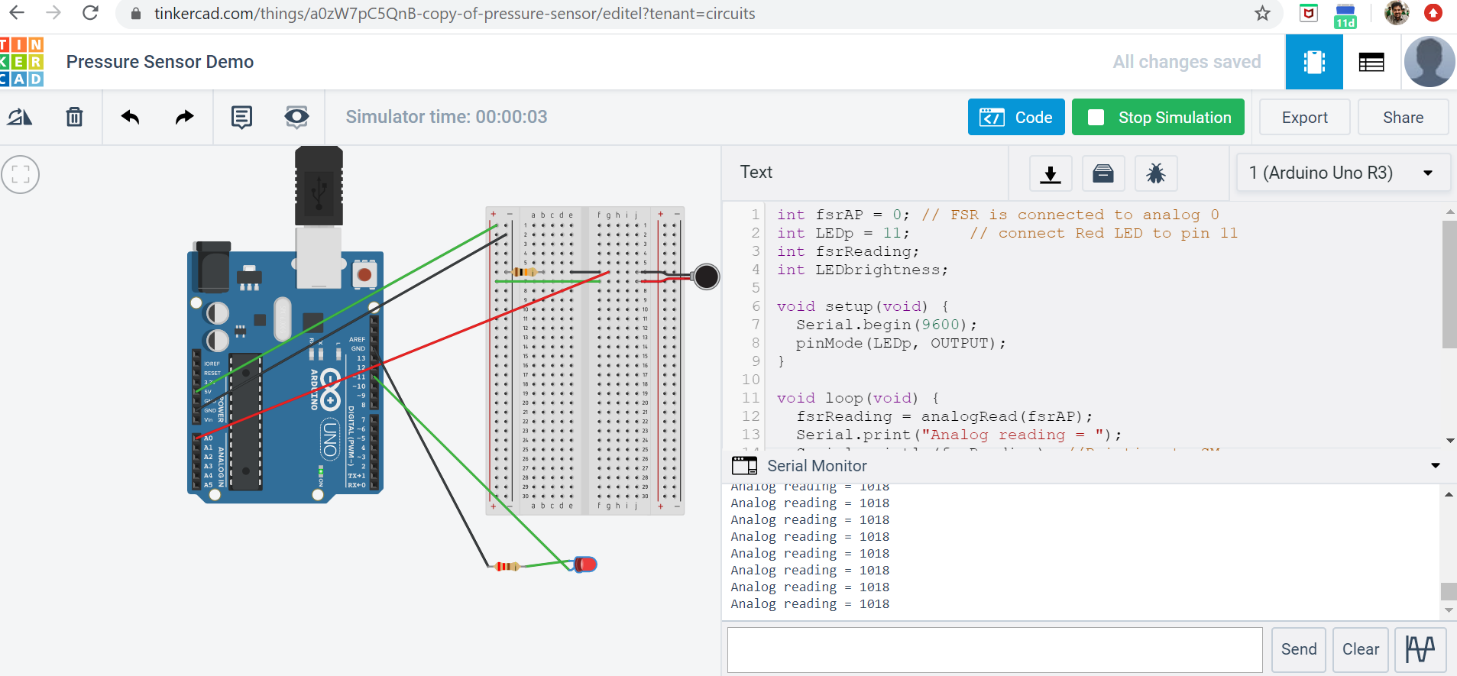
// LED gets brighter the harder you press

analogWrite(LEDp, LEDbrightness);

delay(100);

}

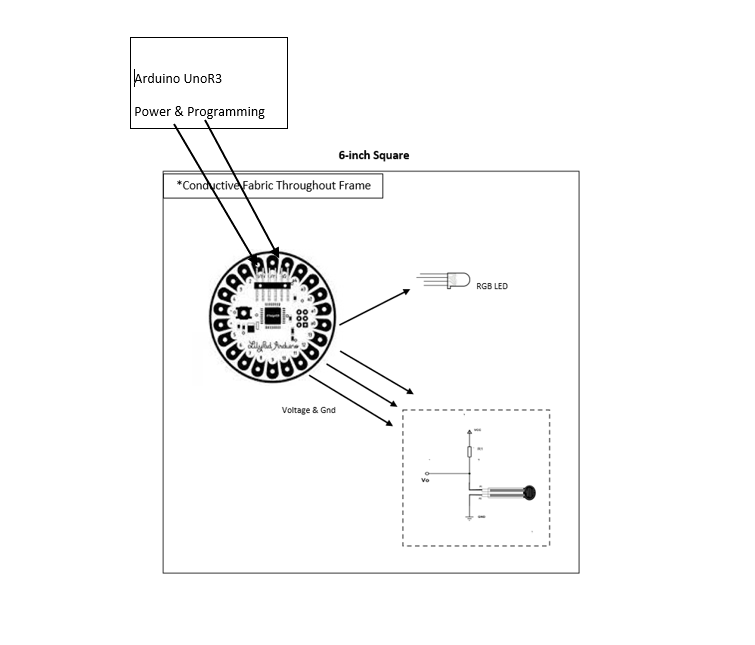
OUTPUT :



**HARDWARE FEASIBILITY**

In the Software prototyping section we saw how to implement the project on a CAD tool, the pressure sensor as a components was missing but we emulated its behaviour using the vibration motor, the aim within this week was to gather all the components needed to implement this project. Before going ahead and doing the initial connection its better to study the design along with its Merits and De-merits and figure out challenges before diving into making the actual connections. While going through the above mentioned designs we realised there was an issue of powering the lilypad in its raw form and we would need regulated supply to the board in order to safely implement all components. We are considering using a Arduino UnoR3 to power the system and relay the program to the lilypad. The aim now on will be to look for a suitable solution to this, we can either keep lilypad or we can consider discarding it as a component and use UnoR3 only to power and program the project. We aim to adopt the most suitable design that enables us to showcase the project within a 15 by 15 frame.

Updated Hardware Sketch :



What we see above is the feasibility of the design keeping in mind the constraints of certain components. The next challenge we are faced with is the voltage limits of the components we use such as the pressure sensor, we want to avoid passing extra voltage to the sensor as it might blow up the sensor, the RGB LED can be negated from the project too as an on board LED can achieve the same effect as the RGB LED.

Test Code for Circuit : (Link : <https://pimylifeup.com/arduino-force-sensing-resistor/>)

int pressureAnalogPin = 0; //pin where our FSR is located.

int pressureReading; //variable for storing our reading

//Will Adjust these if required.

int noPressure = 5; //Threshold for no pressure on the pad

int lightPressure = 100; //Threshold for light pressure on the pad

int mediumPressure = 200; //Threshold for medium pressure on the pad

void setup(void) {

Serial.begin(9600);

}

void loop(void) {

pressureReading = analogRead(pressureAnalogPin);

Serial.print("Pressure Pad Reading = ");

Serial.println(pressureReading);

//Will replace Serial Code for Onboard LED of Arduino

if (pressureReading < noPressure) {

Serial.println(" - No pressure");

} else if (pressureReading < lightPressure) {

Serial.println(" - Light Pressure");

} else if (pressureReading < mediumPressure) {

Serial.println(" - Medium Pressure");

} else{

Serial.println(" - High Pressure");

}

delay(1000);

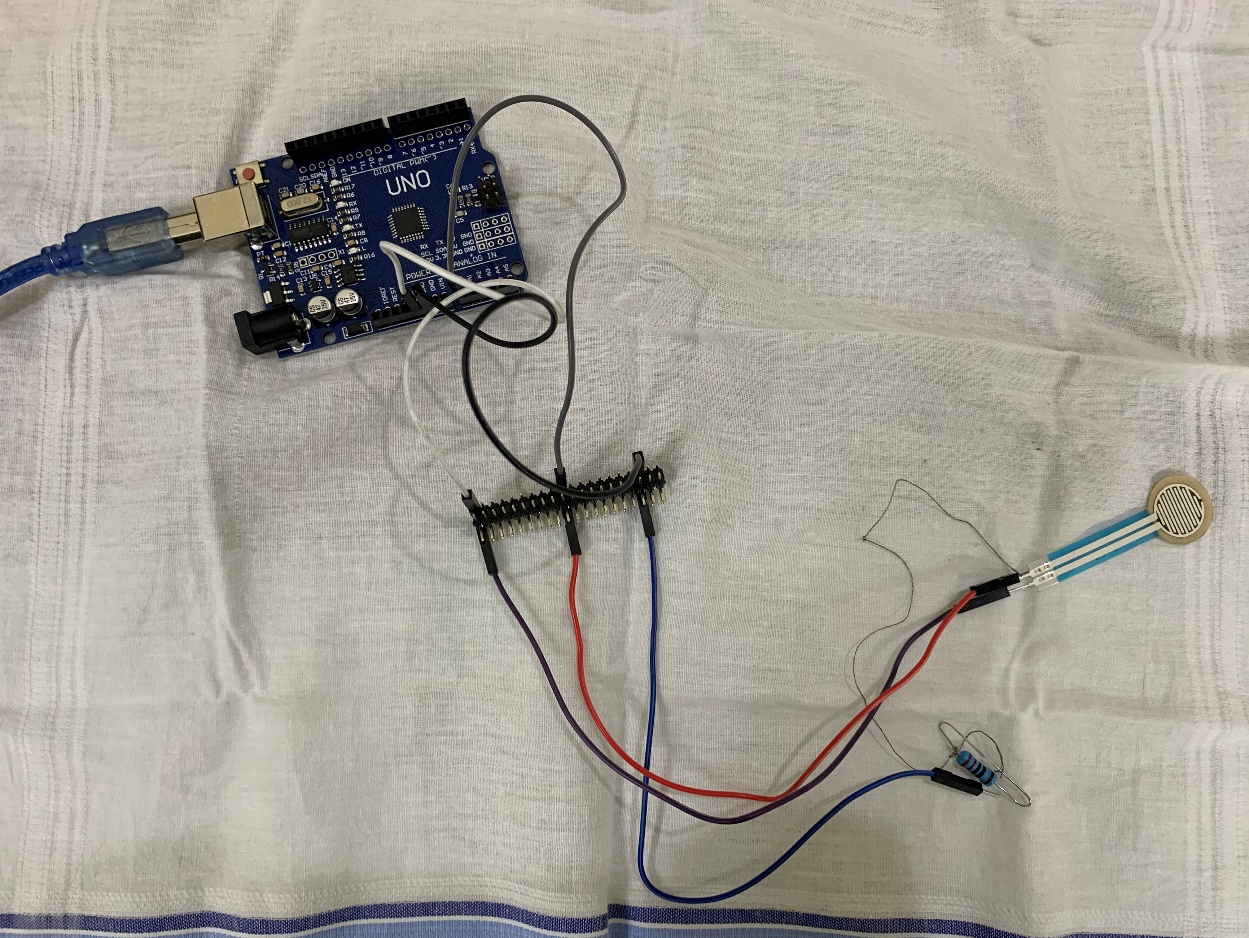
}

**This Code is just to test the circuit once I receive the components.**

**HARDWARE IMPLEMENTATION**

In this section we will implement the hardware components to form a practical working circuit and check if what we have theoretically sought after is physically implementable. In the sections above we have been able to sketch out the concept and build a code to test the circuit now we would like to implement the circuit with actual components and check if it actually works. In the initial list we called for 220 Ohm resistors, on receiving the components we did a pilot study of how much resistance should be required to ensure proper functioning of the FSR sensor or the force sensor we are using, as it turns out we should use 10K Ohm or 100K Ohm resistors for the FSR sensor, even though at the moment we have 220 Ohm resistors we will build the circuit but will refrain from powering it on so that we don’t blowup the FSR, the assorted resistors kit has been ordered to fix this, so for the next edition we will only require to swap the 220 ohm resistor with the actual resistor and then we can power on the circuit. We have already coded the Arduino Uno r3 so that as soon as the resistors arrive we can test out the project. Practically everything has been studied and we are confident the results will match our expectation.

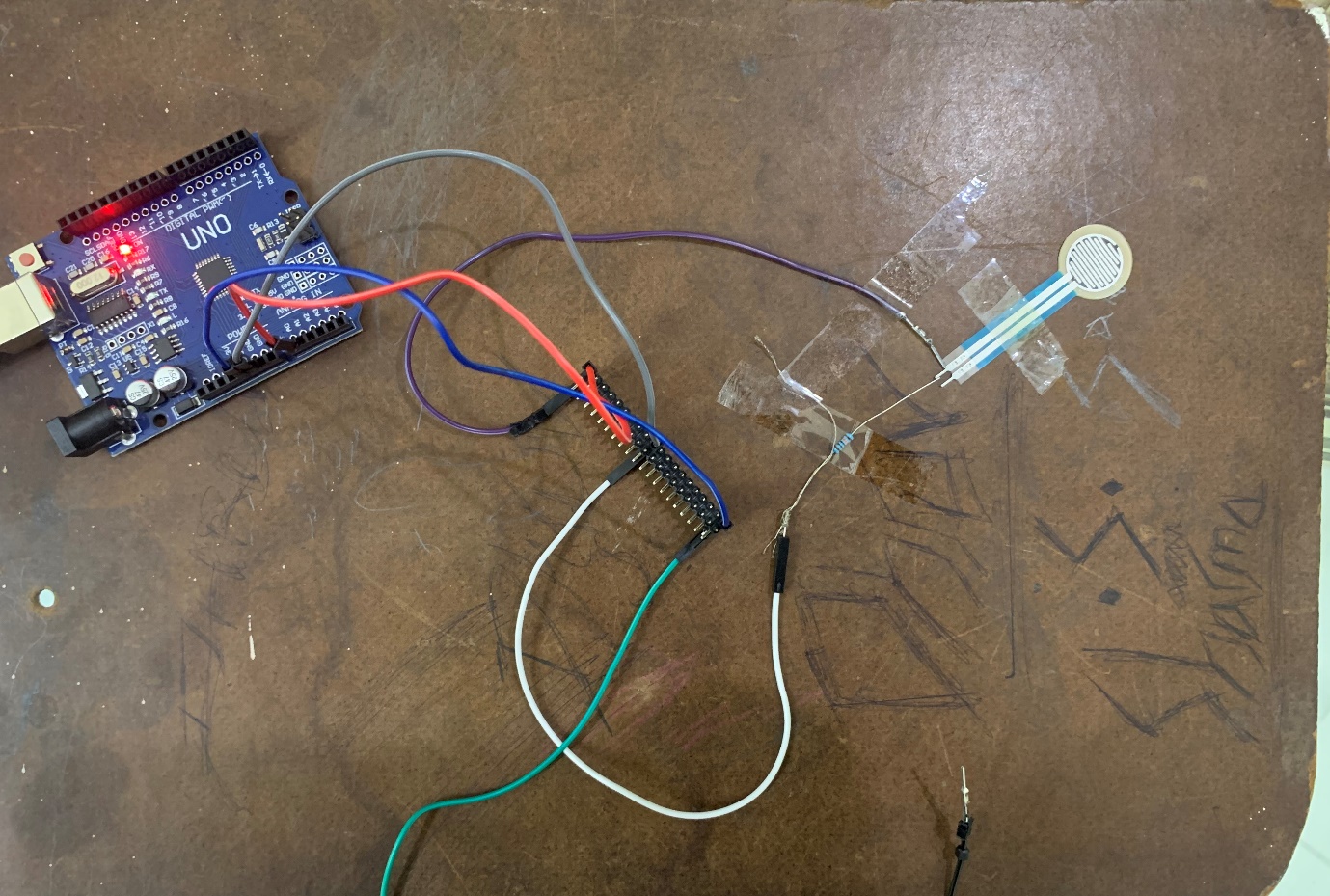
VISUAL :



As seen from the image we have used jumper/header cables to test out the circuit, the resistor has been connected to the sensor via the conductive thread. The resistor is connected to the left pin of the resistor which is connected to **ground** from the Arduino board, the right pin of the FSR is connected to the **5V power supply pin**, the left pin is also connected to the **A0 pin** on the board. We have used a bridge which we will use in the final version too, where the connections from the FSR will be relayed to the board using the jumper cables. Here we can see a **220 Ohm resistor** due to a problem with the shipment, new assorted kit of resistors have been ordered to replace that with a 10K/100K ohm resistor which will complete our circuit and we can power it on. We haven’t done this yet because it might result in the sensor blowing and we do not want to do that given the covid status here. On board LED will suffice.

**FINAL WORKING PROTOTYPE**

In this section we will add the details of the final implementation we performed, taking over from the last section, we did not have the required resistor to actually carry out the implementation, we have acquired the resistor and tested out our circuit which now works as expected. The circuit has been maintained as seen above in the picture. We just swapped the 220 Ohm resistor with a resistor of higher resistance value. First we used a 100K Ohm resistor just as a test to see how does our circuit work, what we found was that pressure sensor logged only High Pressure and No Pressure and completely neglected the light and medium pressures as we can see in the code presented above. On researching further we realised that this limited the current more than we wanted to the FSR sensor, on going through the literature behind the FSR we realised that pressure changes resistance values and sends them to analogue A0, then we connected a lower resistance resistor to see if this problem is mitigated. We used a 10K Ohm resistor and this problem was finally solved. We were successfully able to see the output on the serial monitor and the on-board LED. If followed, this section will enable a reader to re-create the entire project up until this point.



**In the code we have tested and adjusted the code values to correspond to a higher visible output, the default values we set were low and the output wasn’t extremely distinguishable hence we adjusted the values to enable verification of output.**

**Updated Code for Implementation : (We used the on-board LED for output)**

int pressureAnalogPin = 0; //pin where our FSR is located.

int pressureReading; //variable for storing our reading

//Will Adjust these if required.

int noPressure = 5; //Threshold for no pressure on the pad

int lightPressure = 200; //Threshold for light pressure on the pad

int mediumPressure = 500; //Threshold for medium pressure on the pad

void setup(void) {

Serial.begin(9600);

pinMode(LED\_BUILTIN, OUTPUT);

}

void loop(void) {

pressureReading = analogRead(pressureAnalogPin);

Serial.print("Pressure Pad Reading = ");

Serial.println(pressureReading);

if (pressureReading < noPressure) {

Serial.println(" - No pressure");

} else if (pressureReading < lightPressure) {

Serial.println(" - Light Pressure");

digitalWrite(LED\_BUILTIN, HIGH);

delay(1000);

digitalWrite(LED\_BUILTIN, LOW);

delay(1000);

} else if (pressureReading < mediumPressure) {

Serial.println(" - Medium Pressure");

digitalWrite(LED\_BUILTIN, HIGH);

delay(500);

digitalWrite(LED\_BUILTIN, LOW);

delay(500);

} else{

Serial.println(" - High Pressure");

digitalWrite(LED\_BUILTIN, HIGH);

delay(200);

digitalWrite(LED\_BUILTIN, LOW);

delay(200);

}

delay(1000);

}

**EXPLANATION & WORKING : (Using Information provided here you can re-create the circuit)**

For ease of explanation we have color coded the wires and using the colors one can understand and make the circuit on their own. We have used the Arduino Uno as the control unit for the system. The **Blue wire** is the **5V power supply** that is relayed to the connecting bridge that can be seen, similarly the **grey wire** is **GND** and the **Red wire** is **A0.** From the bridge we connect the second side to the FSR Sensor, we used the connector pin bridge as we didn’t have any male to male header cables and this seemed the optimal solution. The **purple cable** extends the **red cable** and is connected to the **left pin** of the **FSR**, The **white cable** extends the ground cable which is colored **grey** which is connected to a **10K resistor** , this too is connected to the **left pin** of the **FSR sensor**, this cable carries the **analog input from the FSR** back to the microcontroller which we can process using our code and display the relevant output. The **green wire** you see is connected to a **black cable** that we can see at the edge of the picture, this was done just to increase the length of the wire and help us move the pin without moving the bridge. The black cable is just the **5V supply** from the board, we manually connect the pin to the FSR **right pin** and press the FSR on **the circular part**, this is when we must have the serial monitor on the Arduino software on so that we can **see the values logged**.

**LESSONS LEARNT**

The first lesson we learnt was regarding design, we were keen on implementing the project with an Arduino lilypad but we received an older board which didn’t have power-supply and would require a controller to limit the supply to the board, we sought after buying the lilypad USB plus which sadly isn’t available at the moment, hence we will be using the UNO R3 at our disposal. The proposed design above can be implemented but we questioned our-self if that is good design. Second big lesson is to always do your math before implementing, the good thing that happened was that we were able to foresee the situation with the help from an online community, a simple physics formula of V=IR to decide the resistance, this is the reason we are waiting for the 10K/100K resistors to test out performance. Lastly, we realised that the conductive thread isnt useful till final implementation. The thread needs to be soldered to the FSR and the resistors to maintain contact hence we used the header cables for the pilot testing.

While creating the working prototype we learnt the affect of value of resistance on values we log, having optimal resistance is vital for the accurate values to be logged, we connected 100K Ohm in the first run and realised that the values weren’t accurate, we did some background study and decided to go in for a lower resistance value. 10K Ohm turned out to be the right resistance for our circuit, we were able to get close to accurate values and realised we need to lower the delay time to take inputs faster.

**PROJECT REFERENCES**

All Materials referenced below hold relevance to the on-going project in some capacity. These aren’t directly observed within the scope of the project. (ACM DL Format followed)

1. J. Meyer, P. Lukowicz, and G. Troster: Textile Pressure Sensor for Muscle Activity and Motion Detection, 10th IEEE International Symposium on Wearable Computers, 69--72, 2006
2. M. Rothmaier, P.M. Luong, and F. Clemens: Textile pressure sensor made of flexible plastic optical fibers. Sensors, 8(7):4318--4329, 2008.
3. M. Sergio, N. Manaresi, M. Nicolini, D. Gennaretti, M. Tartagni, and R. Guerrieri: A textile-based capacitive pressure sensor. Sensor Letters, 2(2):153--160, 2004