

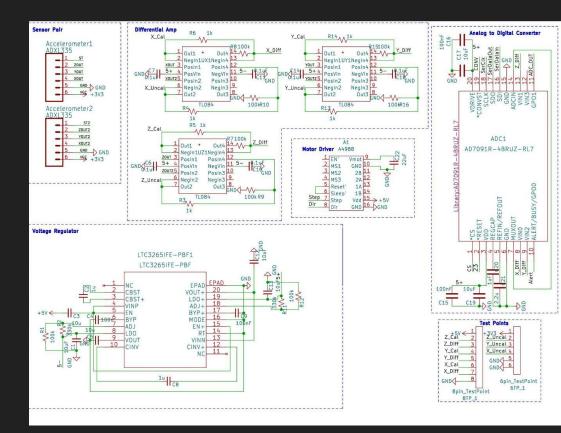
#### Security PUFfins Month Recap Presentation Client: Julie H. GTA: Jordan Beverly

Benjamin Assmann, Sharley Fabro, and Traigh Kirkeeng 14 October, 2022

### PCB

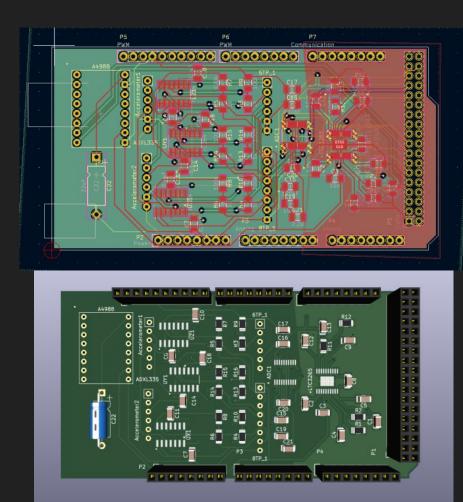
#### Modifications

- Added more differential amplifiers
- Added motor driver for testing
- Added voltage regulator to supply a reliable ±5∨
- Added test points to easily identify if there is data being read



#### PCB

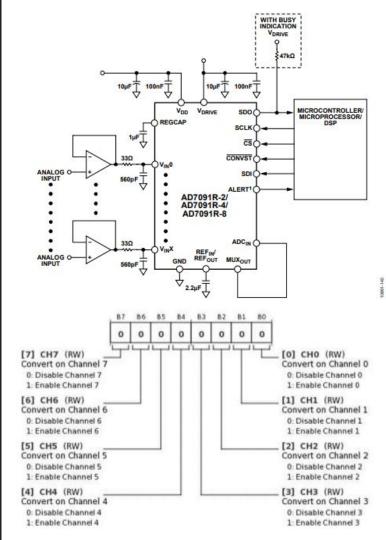
- Grounding planes are utilized to isolate signals
- Testing points are available for raw data values to be read
- Capacitors are utilized to stabilize input voltages to devices
- Resistor values were chosen according to datasheet specifications
- PCB will arrive by the end of next week



### Arduino and ADC

- Pseudo code for reading and measuring values has been developed
  - Configure registers on the ADC
  - $\circ$  Start the conversion
  - Read the digital value
  - Change channel and repeat for other 2 channels

- Potential Issues with timing
  - Arduino clock speed is 62.5 ns
     ADC can lose data after a few 100 ns



# Arduino and ADC

- Stepper Motor code has been completed
- Simple 40 step rotation, with stops to read values from accelerometers
- Currently reads values and develops PUFs, but not very precise

```
void loop()
{
    int steps;
    while (steps < 40) {
        if (steps == 0) {
            delay(3000);} //wait 3 seconds. It gives enough time to read again once it comes back to the original position.</pre>
```

// Set motor direction clockwise.
 digitalWrite(dirPin, HIGH);

```
for(int x = 0; x < stepsPerRevolution; x++)</pre>
```

digitalWrite(stepPin, HIGH); delayMicroseconds(2000); digitalWrite(stepPin, LOW); delayMicroseconds(2000);

```
delay(1000); // Wait 1 second
//reset the values before every read
float ZDiff = 0.0;
float XDiff = 0.0;
float XDiff = 0.0;
float ZCal = 0.0;
float XCal = 0.0;
float XCal = 0.0;
```

```
for(i=0; i< 10; i++) {</pre>
```

```
ZDiff = analogRead(A0);
ZCal = analogRead(A1);
YDiff = analogRead(A2);
YCal = analogRead(A3);
XDiff = analogRead(A4);
XCal = analogRead(A5);
```

```
//store 10 values for each axis
ZDiff_values[i]= ZDiff;
YDiff_values[i]= YDiff;
XDiff_values[i]= XDiff;
ZCal_values[i]= ZCal;
YCal_values[i]= YCal;
XCal_values[i]= XCal;
}
```

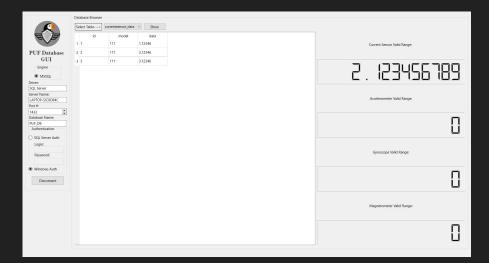
# Database

- Schema has been redone a few times to work with Microsoft SQL Server 2019
- Two ways of data insertion
  - One through built in Excel functionality and SQL Server Management Studio (SSMS)
  - Other through python script written that reads file and creates query file
- Working on writing documentation to ensure usability for everyone

ect -     **     **     **     **     **       LAPTOP-SIC8D84C (SQL Server 15.02000       Databases       *     System Databases       *     Database Snapshots       *     Database Diagrams       *     Database Diagrams       *     Tables       *     Tables       *     File Tables       *     File Tables       *     Graph Tables       *     #       <
■ System Databases         ■ System Databases         ■ Database Snapshots         ■ PUF_DB         ■ Tables         ■ Tables         ■ Tables         ■ System Tables         ■ FileTables         ■ External Tables         ■ Graph Tables         ■ dbo.accelerometer_data         ■ dbo.currenteensc_data         ■ dbo.currenteensc_data         ■ dbo.gyroscope_data         ■ dbo.gyroscope_data         ■ dbo.wagnetometer_data         ■ dbo.wagnetometer_data
Databases     System Databases     Oatabase Snapshots     Database Diagrams     Tables     System Tables     FileTables     Graph Tables     Graph Tables     dot.accelerometer_data     dot.acclibrated_data     dot.acrenteens_data     dot.gyroscope_data     db.gyroscope_data     db.magnetometer_data     db.magnetometer_data
<ul> <li>System Databases</li> <li>Database Snapshots</li> <li>PUE_DB</li> <li>Database Diagrams</li> <li>Tables</li> <li>System Tables</li> <li>FileTables</li> <li>FileTables</li> <li>Graph Tables</li> <li>Endoaccelerometer_data</li> <li>Endob.ccrelerometer_data</li> <li>Endob.core.dibrated_data</li> <li>Endob.core.dibrated_data</li> <li>Endob.core.dibrated_data</li> <li>Endob.core.dibrated_data</li> <li>Endob.core.dibrated_data</li> <li>Endob.core.dibrated_data</li> <li>Endob.core.data</li> <li>Endob.core.data</li> <li>Endob.core.data</li> <li>Endob.core.data</li> <li>Endob.core.data</li> <li>Endob.core.data</li> <li>Endob.core.data</li> </ul>
<ul> <li>Database Snapshots</li> <li>PUF_DB</li> <li>Database Diagrams</li> <li>Tables</li> <li>System Tables</li> <li>FileTables</li> <li>External Tables</li> <li>Graph Tables</li> <li>Graph Tables</li> <li>If doo.accelerometer_data</li> <li>If doo.culibrated_data</li> <li>If doo.currentsensor_data</li> <li>If doo.gyroscope_data</li> <li>If doo.magnetometer_data</li> <li>Views</li> </ul>
<ul> <li>Gatabase Diagrams</li> <li>Tables</li> <li>System Tables</li> <li>FileTables</li> <li>FileTables</li> <li>FileTables</li> <li>Graph Tables</li> <li>Gathabase</li> <li>Machine data</li> <li>Machael data</li> </ul>
System Tables     System Tables     System Tables     Fiel Tables     Graph Tables
Synonyms     Programmability     Service Broker     Storage     Security     SQL Server Import and Export \     Review Data Type Mappin     Select a table to review how its
conversion issues.
Table:
Table:
Table:
Table: Source Source 'Sheet 15'
Table: Source Sheet 15'
Table: Source Sheet 1\$' C Data type mapping: Source Column Source Typ D VarChar
Table: Source Source 'Sheet 15'
Table: Source Sheet 15' < Data type mapping: Source Column Source Typ

# Graphic User Interface(GUI)

- Ability to scale window and display as many data points as in table
- Cleaned up code and commented out sections for readability
- Ability to find statistic was added
- Ready for testing of bigger data sets and real data from our board



#### What needs to be Done

#### • PCB

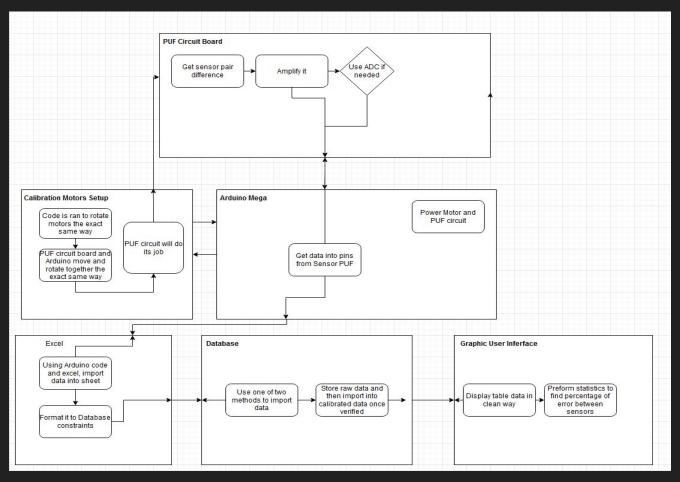
- Populate the board
- Test the board
  - Voltage regulator supplying ±5V
  - Components are working as intended
  - Motor driver is working
- Make revisions
- Test Arduino Code with fully populated board

• Integrate all components and test the overall product together

#### • Database and GUI

- Transfer compiled GUI and database code files from personal machine to a lab laptop
- Ensure the system as a whole works with real data from our circuit
- Fix math function. Make sure math function is accurate with small data set before large sets are used
- Help others get familiar with how to use software and import data for calibration

### **Testing Process**



# Thank you for your time Any Questions?