* Materials to purchase

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Component**  **Name** | **Unit Price ($)** | **Quantity** | **Unit** | **Description** | **Photo** |
| [Arduino Uno R3](http://www.adafruit.com/products/50) | 24.95 | 1 | piece | Function as ADC |  |
| [TMP36](http://www.adafruit.com/products/165) | 1.50 | 6 | piece | Analog Temperature Sensors |  |
| [USB Cable](http://www.adafruit.com/products/62) | 3.95 | 1 | piece | Standard A-B, 1m |  |
| [Breadboard](http://www.adafruit.com/products/64) | 5.00 | 3 | piece | half-size 5.5 cm x 8.5 cm |  |
| [Breadboarding wire](http://www.adafruit.com/products/153) | 6.00 | 1 | bundle | Approximately 65 wires in a mix of colors, 120mm - 250mm long |  |

Total price: $58.9

* Other Materials

A laptop with Windows 7, Matlab, Visual C++ IDE and Arduino IDE installed.

* Description

We use six analog sensors (TMP36) for measuring temperature in same spot and converting them into voltage values.

Then we connect TMP36s with analog inputs on an Arduino Uno R3 board, which is used as an analog-to-digital converter (ADC).

Next, the R3 board translate those six voltage signal into digital data, and a periodic collecting program inside the R3 board transmits them into a laptop through USB connection and virtual COM port.

In the laptop, a serial port access program reads the voltage data coming from the virtual COM port, and converts them into temperature values as well as saves those values to a text file with timestamp.

Finally, we use Matlab for analyzing those temperature variation data.

* Schematic Diagram



* Programs to develop

1. Front-end data acquisition program

For an Arduino Uno R3 board to function like an ADC as well as transmit the data to a laptop, we need to write a data acquisition program for R3 boards. This program would drive a R3 board to read its six analog inputs periodically (in second interval), form data frames, and send those data frames to a laptop through USB and virtual COM port.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Field(bytes) | STX(2) | CMD(1) | T1(2) | T2(2) | T3(2) | T4(2) | T5(2) | T6(2) | XOR(1) | ETX(2) |
| content | **FF 02** |  | data1 | data2 | data3 | data4 | data5 | data6 |  | **FF 03** |

Data frame format for virtual COM port

The R&D tool for this acquisition program is Arduino IDE.

1. Back-end receiving program

We also need to write a back-end receiving program running in a laptop in order to receive and interpret data frames coming from the virtual COM port. This program inquires the virtual COM port periodically, gets every voltage value from those six sensors, translates them into temperature values, tags them with timestamp, and saves them into a text file for later Matlab analysis.

The R&D tools for this data receiving program are Visual C++ IDE plus serial port SDK.