

Developing A Low-Cost, Arduino Based Datalogger

Kenyon Gentry¹; Westminster College

Research Mentor: Steven Burian², Youcan Feng³, Thomas Walsh⁴; University of Utah

Goals

1. To develop an inexpensive, functionally comparable, Arduino based data logging unit.

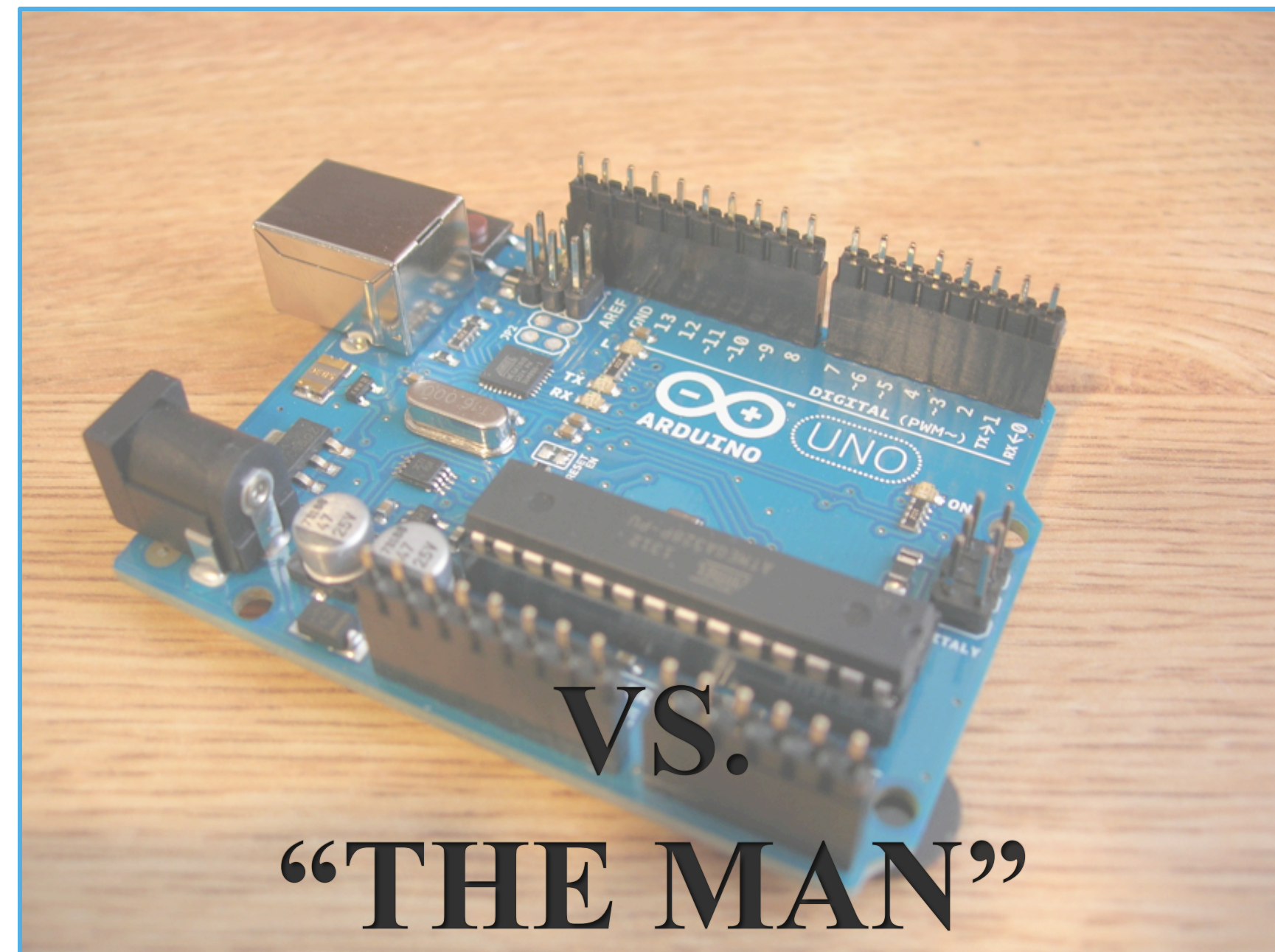
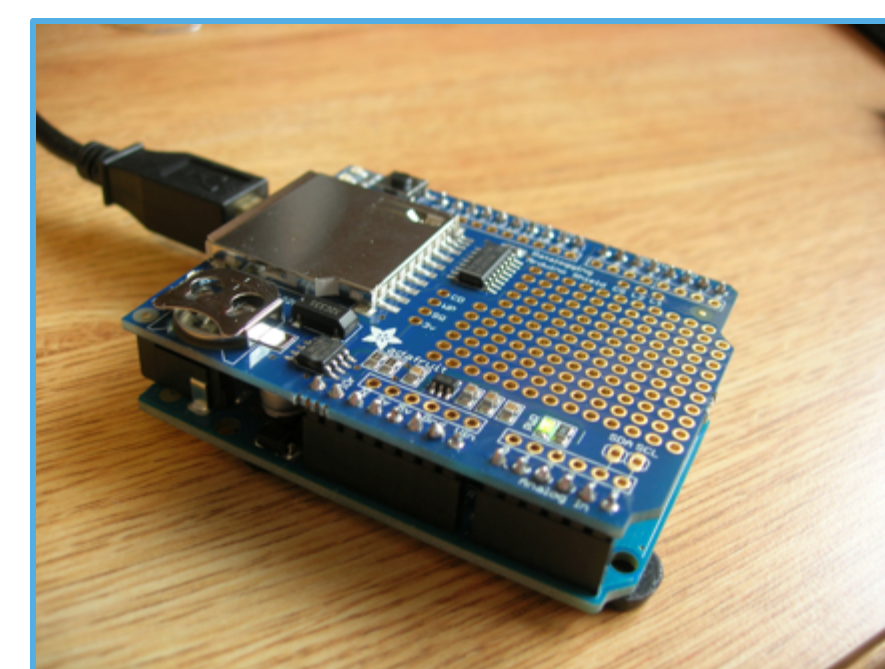


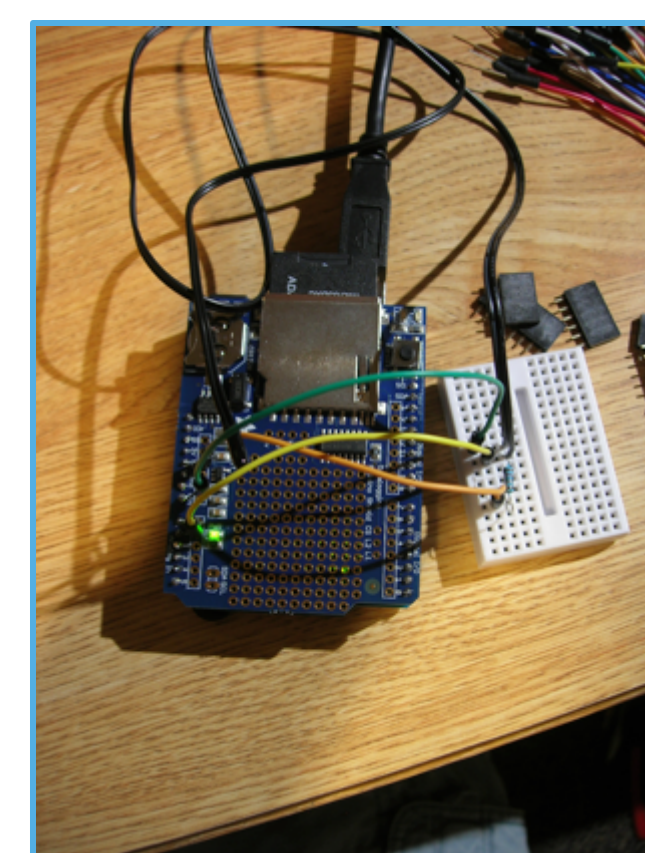
Image 1: Arduino Uno R3

2. To compare a traditional tipping-bucket method of measuring outflow against an eTape volume sensor and Arduino data logger.

Simple Process



- Install data logging shield for RTC and SD storage.



- Connect various sensors.

```
01307
// Save and time functions using a DS1307 RTC connected via I2C and Wire lib
#include <Wire.h>
#include "RTClib.h"
RTC_DS1307 rtc;

void setup() {
  Serial.begin(9600);
  pinMode(LED_BUILTIN, OUTPUT);
  Wire.begin(); // Shield I2C pins connect to alt I2C bus on Arduino Due
  rtc.begin();
}

void loop() {
  if (!rtc.isrunning()) {
    Serial.println("RTC is NOT running!");
    // following line sets the RTC to the date & time this sketch was compiled
    rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
    // This line sets the RTC to an explicit date & time, for example to set
    // January 21, 2014 at 0am you would call:
    // rtc.adjust(DateTime(2014, 1, 21, 0, 0, 0));
  }

  Serial.print(DateTime(rtc.now().year(), rtc.now().month(), rtc.now().day(),
    rtc.now().hour(), rtc.now().minute(), rtc.now().second(), true));
  Serial.print(" ");
  Serial.print(rtc.now().year());
  Serial.print(" ");
  Serial.print(rtc.now().month());
  Serial.print(" ");
  Serial.print(rtc.now().day());
  Serial.print(" ");
  Serial.print(rtc.now().hour());
  Serial.print(" ");
  Serial.print(rtc.now().minute());
  Serial.print(" ");
  Serial.print(rtc.now().second());
  Serial.print("\n");
}
```

- Configure Board: upload example code from GitHub.

```
millis,stamp,datetime,light,temp,vcc
2000, 1273007391, "2010/5/4 21:9:51", 233, 80.97, 3.54
3000, 1273007392, "2010/5/4 21:9:52", 235, 80.97, 3.99
4000, 1273007393, "2010/5/4 21:9:53", 236, 80.97, 4.04
5000, 1273007394, "2010/5/4 21:9:54", 235, 80.97, 4.04
6000, 1273007395, "2010/5/4 21:9:55", 236, 80.97, 4.05
7000, 1273007396, "2010/5/4 21:9:56", 242, 80.97, 4.05
8000, 1273007397, "2010/5/4 21:9:57", 328, 80.97, 4.05
9000, 1273007398, "2010/5/4 21:9:58", 330, 80.97, 4.04
10000, 1273007399, "2010/5/4 21:9:59", 343, 80.97, 3.99
11000, 1273007400, "2010/5/4 21:10:0", 368, 80.97, 4.08
12000, 1273007401, "2010/5/4 21:10:1", 567, 80.97, 4.05
13000, 1273007402, "2010/5/4 21:10:2", 521, 80.97, 4.05
14000, 1273007403, "2010/5/4 21:10:3", 520, 80.97, 4.05
15000, 1273007404, "2010/5/4 21:10:4", 520, 80.97, 4.05
```

- Modify existing codes to save sensor data as a CSV file.

Comparison

Tipping-bucket and volume sensor methods will be compared to manual flow rate determination.

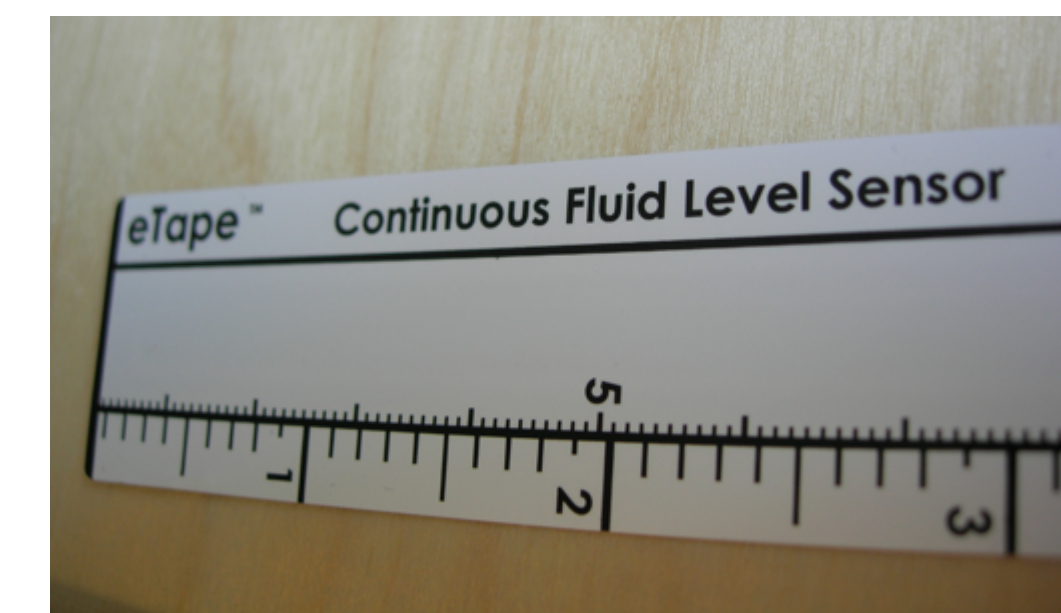


Image 2: Volume sensor used to determine flow rate.

Percent errors will be compared in addition to overall practicability.

Uses



Image 3: Youcan Feng's Lysimeter at the University of Utah Marriott Library.

This unit will be tested for use in a green roof ET rate experiment.

Useful when resources are limited.

IFELLOWS UNDERGRADUATE RESEARCH PROGRAM

¹kg0325@westminstercollege.edu
²steve.burian@utah.edu
³youcan.feng@gmail.com
⁴u0655790@utah.edu



<http://iutahepscor.org>

