Homebrew Fermentation Monitoring System

Kyle Broadway
Real Time Embedded Systems – ECE 4220

What and Why

- Homebrewing perils
    - Bacteria
    - Dead/Spoiled Yeast
    - Temperature changes
- Specific Gravity of the Wort
    - Usually measured using a hydrometer
    - Requires a sample of fermenting wort
        - Risk of infection
        - Loss of wort

What and Why

- Measures Temperature
  - Ensures vitality of yeast
  - Prevents unchecked temperature spikes resulting in off flavors
- Measures specific gravity
  - Monitors progress of yeast
- Much more sterile way to take measurements
- No loss of wort

What and Why

- How it pertains to Embedded Systems

    - Uses the arduino to collect data via sensors on the fermentation tank.

    - Wireless broadcast of data (Zigby protocol) to receiving program on computer

    - Data is broadcast in spurts – loss of data occurs if computer is not consistently watching for data.

    - Need a way always watch for data and store in a buffer where it can be parsed by the computer.

How it was done

- The computer implements two threads

    - One thread watches for incoming data and stores it to a circular buffer, marking that spot as full

    - The other thread watches the circular buffer, parses data when available, and stores into the appropriate text files for later graphing.

- Data is stored into the circular buffer as an object with a data message and a boolean "full" variable.

How it was done

- Because the data is broadcast in spurts, data could be easily lost due to the monitoring computer parsing instead of watching for data.

- The two threads with a circular buffer solves this problem.

- The same problem was encountered during our capstone project.

How it was done

- Program Overview
  - Upon starting the program, a name is required
  - This name is attached to the text files used to store data gathered.
  - Threads 0 and 1 are started. Thread 0 watches for incoming data and saves it to the circular buffer, and marks it as full.
  - Thread 1 parses the data in the circular buffer, marks it as empty, and stores the data in the corresponding text file.
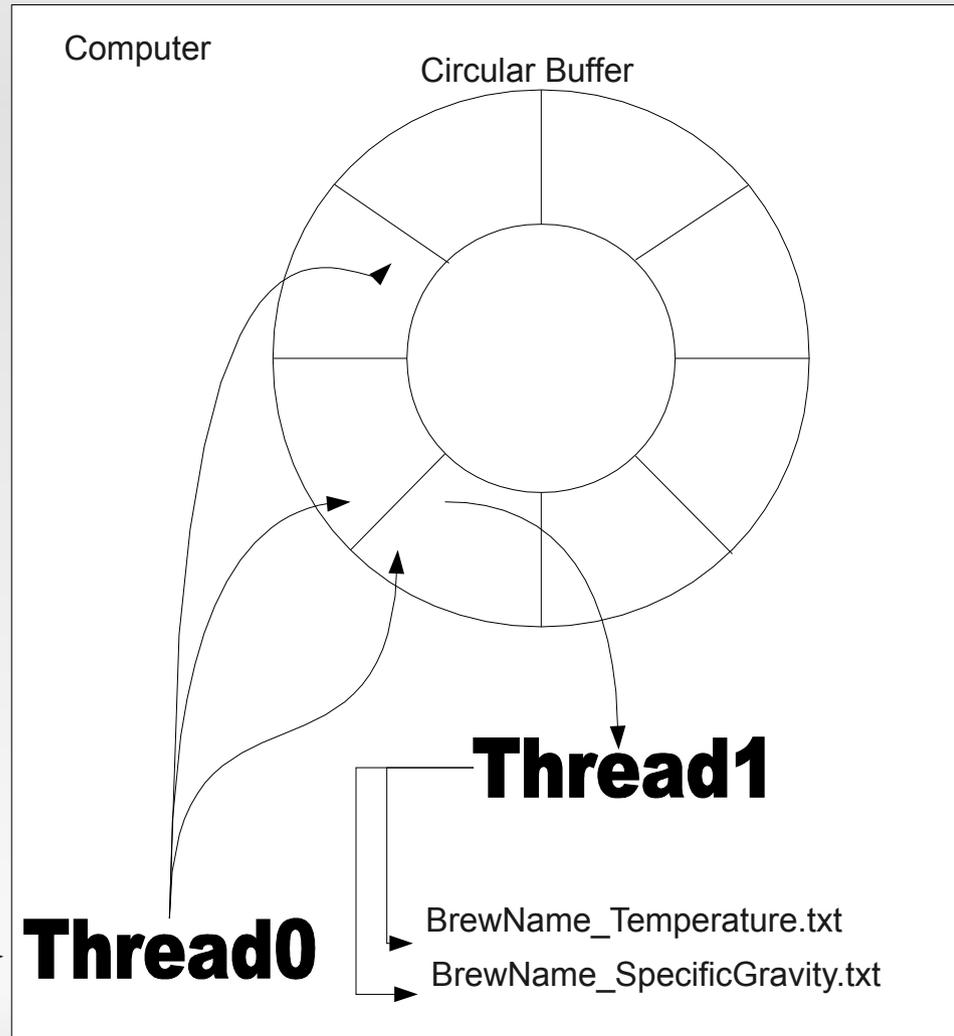
University of Missouri
Electrical and Computer Engineering
ECE 4220 – RT Embedded Systems II

How it was done

Primary Fermentation

Arduino

Computer

Circular Buffer

Thread1

Thread0

BrewName_Temperature.txt

BrewName_SpecificGravity.txt

How it was done

- Specific gravity is calculated in the second thread using the equation:

$$\left(\frac{(TankWeight_{kilograms})}{Volume_{Gallons}*0.00378541178_{(cubic\ meters\ per\ gallon)}}\right)*\left(\frac{1}{DensityOfWater_{(kilograms\ per\ cubic\ meter)}}\right)$$

Results

- Using a testing program generating data for the computer to process, the program is a success.

- Text files are created with unique filenames

- Circular buffer and two thread implementation resulted in 0% lost messages during testing.
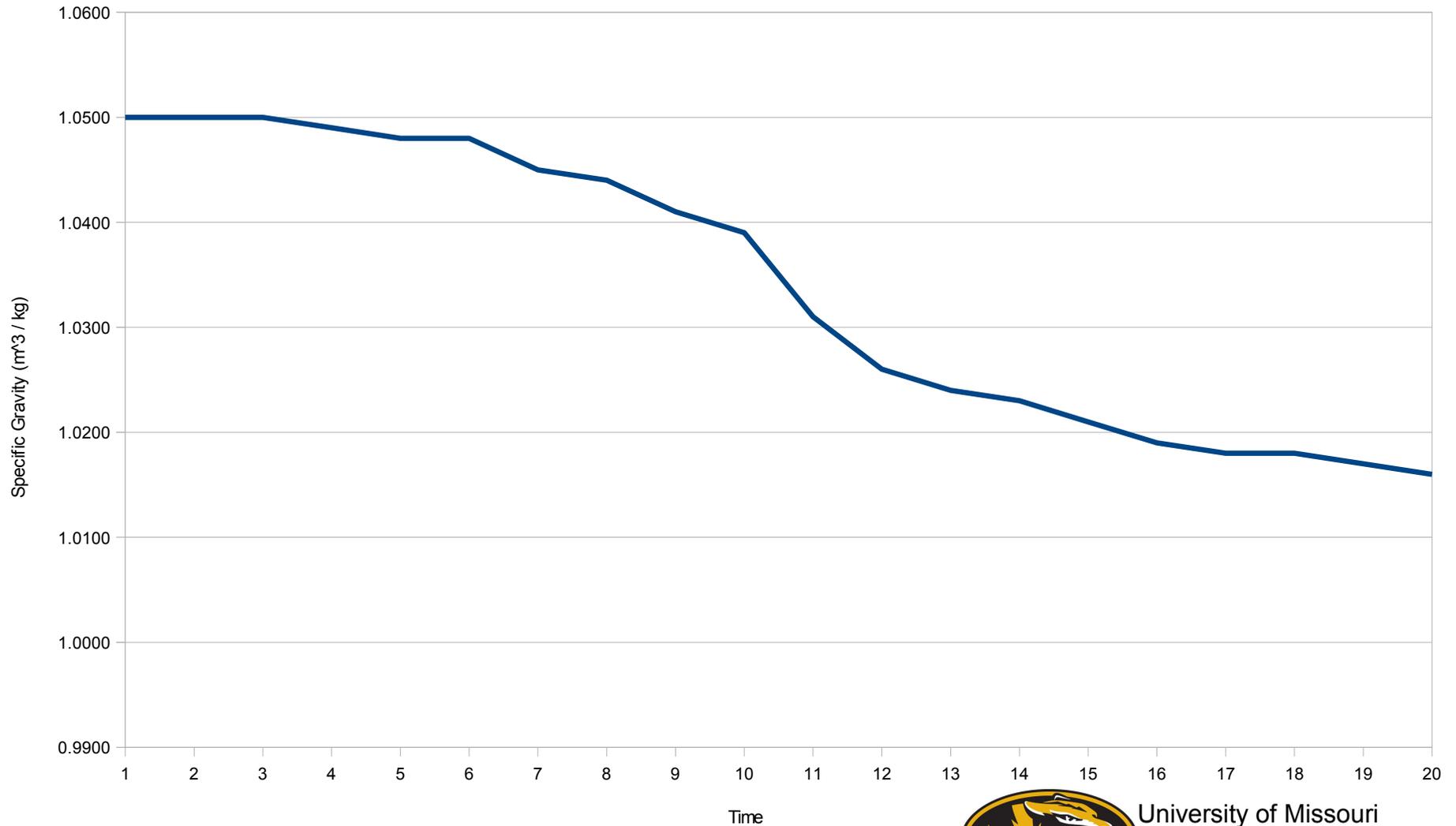
Results

- Unfortunately, the brewing process takes about two weeks in primary fermentation, and another 2 – 4 weeks in secondary fermentation.

- Because of this time constraint, it was unrealistic to test this project on a real batch of homebrew.

- Judging by the handling of generated samples however, this project is a success.

- Generally speaking, temperature should remain constant.
  - 15-24C (55-75F) for Ales
  - 0-10C (32-50F) for Lagers

(Projected) Results

Specific Gravity over Time

Fermemtation monitor

Results

Questions or Comments?