Project in Embedded Systems 15hp 1TE721

A FERMENTATION TEMPERATURE MONITORING SYSTEM
using Atmel AVR and Raspberry Pi

August Forsman
aufo8456@student.uu.se

Background

In recent years, the number of small scale breweries has skyrocketed in numbers. This has led to an increased demand of advanced brewing systems operating on much smaller volumes compared to the macro-scale brewing industry.

For homebrewers food grade steel fermentation tanks with inline glycol chilling systems.

In contrast to said methods, a counter-culture renaissance of uncontrolled brewing often referred to as *farmhouse brewing* by letting a mixed flora of yeasts and bacteria (commonly Brettanomyces, Pediococcus, Lactobacillus) ferment without controlled setting, other than moving the vessel to a suitable space. In order to achieve reproducible results without directly controlling the environment. The concern is mostly not letting the temperature reach certain extremes, high or low. Since there are no control parameters, it is of interest to log things such as the internal and ambient temperature in order to investigate what type of beer styles are good to brew during each season of the year. This is also a question of money since a mixed fermentation beer is sometimes barrel aged for up to three years.

Objectives

The objective of this project is develop a small and portable system prototype that allows an AVR MCU to communicate with a Raspberry Pi (RPi) through serial communication. The signal should not contain any unnecessary noise and can therefor be filtered by the MCU before transmitting to the server.

The RPi based server is accessed on the internet by the end user and can be visualized using external libraries that works on both computers and mobile units.

Preliminary tasks

The time planning for the enumerated tasks is visualized as a Gantt scheme in Figure 1.

1. Setup project repository, as well as report and documentation workflow. Formulate preliminary time plan and goals. Look for suitable hardware.
2. Get started with the AVR toolchain, write Makefiles and configure a development environment. Configure ARM compatible Linux on the RPi. Write UART routines and verify that the communication is working. Extend the communication to contain temperature data from the sensor.
3. Apply a filter to the signal. Design the website, configure databases and visualization of data.
4. Extend the website with more features. Collect or generate dummy-data in order to present a proof of concept.
5. Present the project, i.e all project phases, and finalize the report

Grading criteria and grade to be achieved[1]

- Grade 3. The temperature is measured by the microcontroller and presented through a visualization on a website that is accessible from the internet. The website is hosted on a Raspberry Pi computer configured with an appropriate ARM compatible Linux distribution.
- Grade 4. The thermostat signal is processed through a Kalman filter or similar. After the initial fermentation, the ambient conditions governs the temperature of the beer. The code is portable and it is possible to extend the system with temperature control in a fermentation chamber.
- Grade 5. In addition to the stated grading criteria, the data is accessed in a manner similar to an interactive process view that a production company might order as a web application.

Grade to be achieved: 5

[1] Suggestions. To be discussed.

Preliminary time schedule

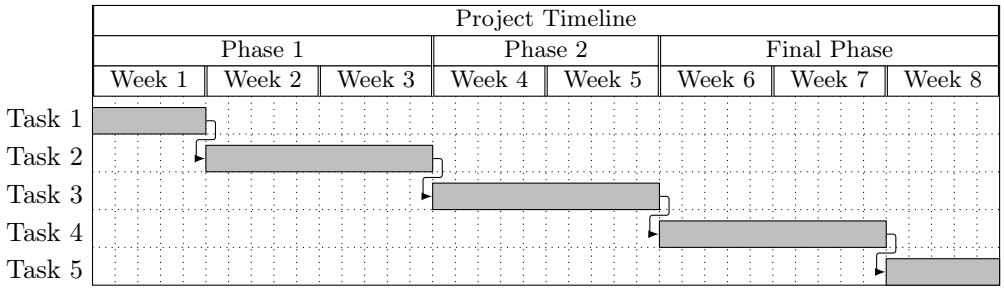

Figure 1: Gantt scheme representing the project timeline.