Experiment no.2

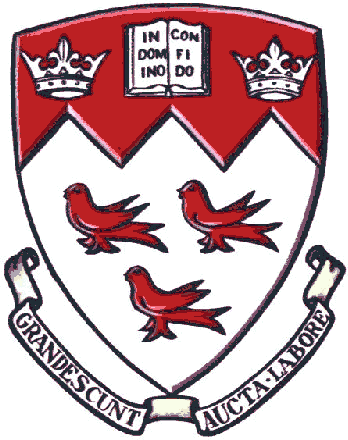
Sensor Data Acquisition, Digitizing, Filtering, and Digital I/O

Maxim Goukhstein ID: 260426769

Olivier Laforest ID:260469066

Department of Electrical and Computer Engineering

McGill University, Montreal



February 21st, 2015

ECSE 426 Microprocessor Systems

**Abstract**

The goal of the experiment presented in this report is to implement a temperature data acquisition system using the STM32F407 Discovery board and display the acquired data through the use of the on board Light Emitting Diodes (LEDs) in order to create a simple output display. This report will show how the built-in temperature sensor of the STM32F407 Discovery board as well as the analogue to digital converter and LEDs were used to achieve the desired system. It will also be shown how filtering of the raw data was done and how pulse width modulation was utilized in order to realize the desired display effects.

**Problem Statement**

In this experiment, the internal processor temperature sensor of the STM32F407 Discovery board is used to get temperature readings of the microprocessor that will be converted into a visual LED display in order to let the user know if the temperature is increasing, decreasing or if the temperature has reach an upper threshold. The display is to be created using the LEDs that are positioned in a diamond shape on the board (i.e. LED 3 to LED6). While in normal operation (i.e. below the upper threshold), only one LED should be on at any one time. For each increase of 2 degrees Celsius, the display should cycle through the four LED lights in a clockwise fashion. In other word, if LED3 is currently lit, after an additional increase of 8 degrees Celsius, the display should have cycled through the four LEDs and LED3 should be lit again. For every decrease of 2 degrees Celsius, the display should cycle through the four LEDs in a counter clockwise manner. If the temperature of the microcontroller exceeds an upper temperature threshold, the display should enter an overheating alarm mode. The alarm mode consists in the four LEDs simultaneously flashing in a fade-in/fade-out manner. When the temperature falls back under the threshold, the alarm mode should be exited and normal mode should resume. Several challenges are associated with the LED display. While in normal operation, the transitions between LEDs should be as definitive as possible (i.e. the LED should ideally not flicker back and forth during a transition from one LED to the next). While in the alarm mode, all four LEDs must smoothly fade-in and fade-out from all the way off to fully on in a cyclic manner. In alarm mode, the LEDs should not be flickering on and off.

**Theory and Hypothesis**

According to [1], the temperature sensor of the microcontroller on the STM32F407 Discovery board is an analogue sensor which outputs a voltage that varies linearly with temperature that ranges between 1.8V to 3.6V. The temperature sensor is internally connected to an Analogue to Digital Converter (ADC) which allows the analogue sensor readings to be converted to digital values. These digitized temperatures readings can then be used in an embedded C program to implement the LED output display. From the observation of the raw data of the digitized temperature readings outputted by the ADC shown figure 1 and from the fact that sensor data is prone to noise [2], it can be seen that the temperature sensor output is significantly affected by noise. It can be seen in figure 1 that raw data samples can vary by more than within only a few samples. Filtering of the raw data is therefore necessary.

Figure 1: Digitized temperature sensor readings outputted from the ADC

From the experiment 2 specifications [2], it is required to use a Kalman filter in order to filter the raw data. The Kalman filter is an optimal estimator for one-dimensional linear systems with Gaussian noise [3]. It is typically used to smooth noisy data and supply estimates of the filter's parameters [3]. From experiment 1 specifications [4], the parameters for the Kalman filter are the following:

|  |  |
| --- | --- |
|  | (1) |
|  | (2) |
|  | (3) |
|  | (4) |
|  | (5) |

These parameters need to be initialized to appropriate values before the filter can be used. To get a better insight into how to initialize some of these parameters it is useful to take a look at the equations governing each update of the Kalman filter which is performed every time a new measurement of the temperature is outputted from the ADC.

|  |  |
| --- | --- |
|  | (6) |
|  | (7) |
|  | (8) |
|  | (9) |

From a quick inspection of (6), (7), (8) and (9), it can be seen that the only 2 independent parameters (i.e. the parameters that are not changed by the updating process) are the process noise covariance (1) and the measurement noise covariance (2), therefore these parameters will have the most impact. From experimentation, the estimation error covariance (4) converges after only a few updates and its initial value is irrelevant. From equation (7), it can be seen that the kalman gain's initial value is not important since as soon as the first update is performed, the parameter will be set to its appropriate filtering value as it only depends on the estimation error covariance (4), which as mentioned above converges after a few updates and the measurement noise covariance (2) which is constant. The parameter (3) is the actual filtered measurement value and its initial value can be set to the first temperature measurement produced out of the ADC. Therefore, the two parameters which have an important impact on the filtered data are the measurement noise covariance (2) and the process noise covariance (1). From the documentation of the Discovery board [5], it can be found that the precision of the temperature sensor is and consequently the measurement noise covariance will be equal to,

|  |  |
| --- | --- |
|  | (10) |

The only parameter left to set is the process noise covariance (1) and since all the other degrees of freedom of the kalman filter have been fixed or are irrelevant and finding the process noise covariance of a sensor is not trivial, the appropriate value for (1) can be set by trial and error on a set of raw data. The value of (1) can be varied until the filtered data exhibit the desired characteristics (i.e. the filtered data follows the local average values of the unfiltered data, and the filtered data's variation from one sample to the next remains small enough for the proper operation of the LED display, otherwise the LEDs are going to flicker back and forth when the temperature approaches a transition value).

**Implementation**

**Testing and Observations**

**Conclusion**

**References**

|  |  |
| --- | --- |
| [1] | "www.st.com," STMicroelectronics, 2013. [Online]. Available: http://www.st.com/st-web-ui/static/active/en/resource/technical/document/datasheet/DM00037051.pdf. [Accessed 21 February 2015]. |
| [2] | A. Suyyagh, "ECSE 426 Microprocessor Systems Lab 2: Sensor Data Acquisition, Digitizing, Filtering, and Digital I/O," 2015. |
| [3] | R. Faragher, "Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation," *IEEE Signal Processing Magazine,* pp. 128-132, 20 August 2012. |
| [4] | A. Suyyagh, "ECSE 426 Microprocessor Systems Lab 1: One-Dimensional Kalman Filter," 2015. |
| [5] | "www.st.com/," STMicroelectronics, 2014. [Online]. Available: http://www.st.com/st-web-ui/static/active/en/resource/technical/document/reference\_manual/DM00031020.pdf. [Accessed 21 February 2015]. |
| [6] | "www.st.com," STMicroelectronic, 2014. [Online]. Available: http://www.st.com/st-web-ui/static/active/en/resource/technical/document/user\_manual/DM00039084.pdf?s\_searchtype=keyword. [Accessed 21 February 2015]. |

Appendix