Michigan Tech

EE5726: Embedded Sensor Networks

Assignment #02

Akhil Kurup

amkurup@mtu.edu

Fall 2016 : September 11 2016

Problem 1

A vibration sensor outputs an analog signal with a peak to peak voltage of **5V** at a frequency of **100Hz**. Here, since no additional property of the signal is specified, we will assume that the signal **is periodic** and that the **max. frequency component** in the signal is **100Hz**.

(a) When digitizing an analog signal, the **Nyquist rate** must be followed. The Nyquist rule states that when an analog signal is digitized, *the sampling rate of the ADC must be at least twice that of the max. input signal frequency.* If this rule is not followed, the data integrity cannot be guranteed and data may be lost. Therefore, in order to preserve all the information in the given input signal, the sampling rate should be:

$(2 \times 100)Hz \quad = 200Hz$

Therefore, if we sample the input analog signal at **atleast 200Hz**, we would not lose any data.

(b) The input analog has some interesting effect which can be observed with a resolution of 0.025V. Now,

Resolution of an ADC is given by : $\frac{\mathbf{V}}{\mathbf{2^n - 1}}$

Required resolution = **0.025V** ; **V = Peak voltage = 5V** ;

Therefore, substituting in above equation we get : $0.025 = \frac{5}{2^n - 1}$

therefore : $(2^n - 1) = \frac{5}{0.025} \quad => \quad (2^n - 1) = 200$

$2^n = 201 \quad => \quad \mathbf{n = 7.651}$

Now, since resolution has to be a whole number, we take it as **8**

Therefore, we can conclude that, **if the ADC has a resolution of atleast 8 bits, we can observe the interesting event at 0.025V**.

Problem 2

To provide a programming interface for *Wireless Sensor Networks* (*WSN`s*), various programming methods are chosen. With the development in current processor technologies and memory models, it`s relatively easy to port an Operating System (OS) to these WSN`s. For different OS`s, based on the application, there are two main types of implementation *Event-based programming* and *Process-based concurrency*. Their detail`s have been discussed here.

Event based programming

In this method of application development, the WSN node waits for an event to occur and appropriate action to be taken. The OS typicalls polls for events and an *event-handler* is used to notify the system of a known event. An event may be anything from *availability of data from a sensor*, *arrival of a data packet*, *expiration of a timer*, etc.

- **Advantages** The programming of this model is simple and is less resource hungry. Since frequent memory jumps are not required as compared to Process based concurrency, memory management is more effecient and chances of errors are less.
- **Disadvantages** As an event is being processed, another event may occur and the processor may not be ready to handle the event. This may lead to loss of events and possible corruption of data.

Process based concurrency

Concurrent programming is a seemingly *parallel* excution of a code on a single CPU. Most current OS`s support concurrent programming in order to service requests without dropping any important events. A *Process based approach* would be benefited with such concurrency.

A process based programming model, changes states and switches tasks based on the state of the processes rather than on incomming events. In practical applications, the node of a WSN has to *reply to radio signals*, *accept data from sensors* and *perform computations*. Therefore, a process based concurrent programming paradigm would ensure that all these tasks get allotted time as per their priority and no task / event is lost.

- **Advantages** This programming model can handle multiple events concurrently and therefore, there is less chances of loosing data.
- **Disadvantages** However, due to frequent task switches, greater memory and processing resources are required. The chances of error increase during these memory switches due to misaligned or incorrect memory addresses. This may possibly also lead to more power consumption.