ARM Based Microcontroller to Simulate a BiPAP Machine

EE2016 Project (Microprocessor Theory and Lab)

July-Nov 2019

Arjun Menon Vadakkeveedu (EE18B104), Sayan Mitra (EE18B156), Aswin Rajesh (EE18B106), Dasu Sri Vishnu (EE18B130)

Introduction

The goal of the project is to simulate the behaviour of a BiPAP Machine using the ARM-based microprocessor NXP LPC2378.

A BiPAP machine is used to treat medical patients with conditions such as COPD, ILD and Asthma; unlike a Continuous Positive Airway Pressure machine (commonly known as a CPAP or ventilator), a BiPAP helps the patient during both inhalation as well as exhalation cycles using oxygen and vacuum pumps respectively.

In this project, we track a patient signal (using variations on a predefined model) and classify it as normal breathing or abnormal breathing; then appropriate routines are performed based on the class of the patient signal.

System Behaviour and Implementation

INPUT AND DATA PRE-PROCESSING MODULE

1. An ideal signal model is generated using a lookup table with 70 values. A triangular wave with two cycles of unequal duration, with time of duration in the ratio 3:4 is used as the ideal breathing pattern signal. Maximum pressure is assumed to be 720 units, while the

least pressure is 120 units. These values may be scaled up or down to appropriate units for real-world purposes.

2. The patient signal is assumed to be scaled-down noisy variations of the ideal breathing pattern. The noise present in the patient signal is assumed to Gaussian. In order to account for the different breathing conditions of people who may use a BiPAP machine, the patient signal is generated from the ideal signal as:

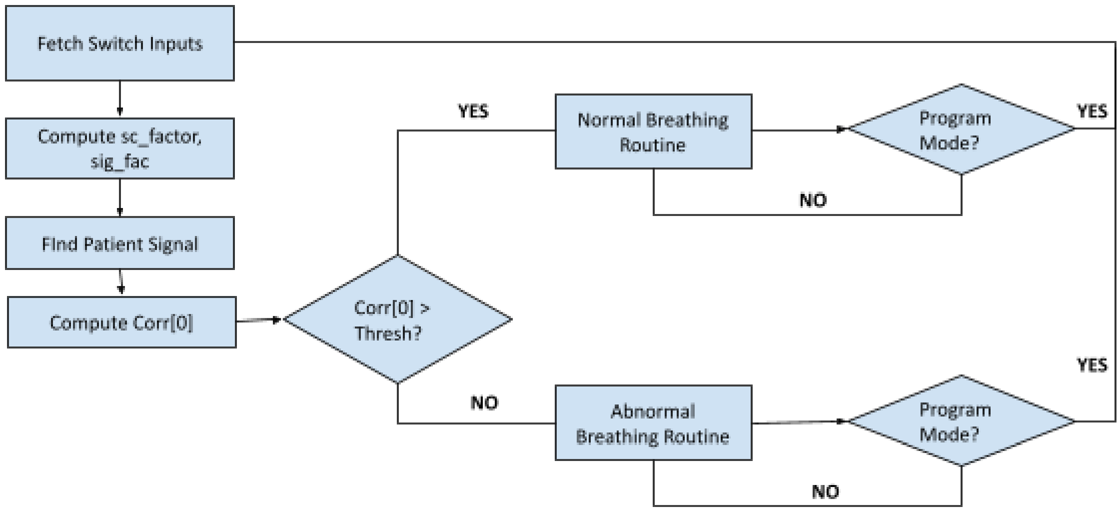
$$P(n) = I(n) * Sc_{factor} + G_{Noise} * \sigma_{scale}$$

   - $P(n) = Patient's\ Breathing\ Signal;\ I(n) = Ideal\ Signal$
   - $Sc_{factor} = scaling\ factor\ of\ ideal\ signal,\ such\ that\ Sc_{factor} \in \{0.9,\ 0.7,\ 0.5,\ 0.3\}$
   - $G_{Noise} = Gaussian\ Noise \simeq N(0,\ \sigma_o)$
   - $\sigma_{scale} = Scaling\ factor\ of\ Standard\ Deviation,\ \sigma_{scale} \in \{0,\ 1,\ 2,\ 3\}$
   - Thus the number of variants of the patient signal modelled in our prototype is 16

3. Samples from a Gaussian Distribution is generated using a Box-Muller Transform and stored in an array of 150 elements. Of these 150 elements, 70 elements are picked at random and added to the scaled-down ideal signal to generate the patient signal.
4. The cross-correlation of the patient signal is computed with the ideal signal at time n = 0, in order to get a quantitative measure of the similarity of the two signals. Based on the value of the correlation, the signal is classified as normal or abnormal.

$$y[n] = Corr(x[n],\ h[n]) = \sum_{k=-\infty}^{\infty} x[k].h[n+k]$$

5. The system operates in program mode when DIP Switch[0] is ‘0’, while it operates in execute mode when the switch is ‘1’. Switch[1:2] is used to set Sc_{factor}, Switch [3:4] to set σ_{scale} and Switch[5:6] to set output mode. The output of the system is unaffected when switches[1:6] are flipped when switch[0] is in execute mode.
6. In a real world implementation of this prototype, a control unit may be attached to the output of the pressure sensor to determine if the signal received belongs to a particular class among the predefined 16 classes (with a margin of error for each class); this class can then be fed as the input to the microcontroller.



Input and Data Pre-Processing Module

OUTPUT MODES:

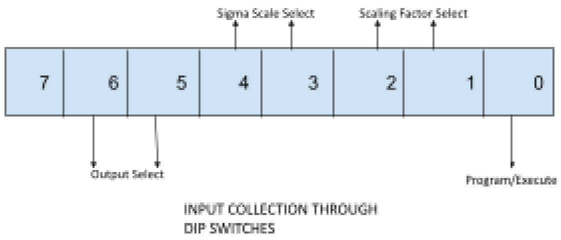
1. If the signal corresponds to Normal Breathing Pattern, LED[0] glows
2. If the signal is Abnormal, LED[1] glows and runs the Abnormal Routine
3. Using a PID Controller with the input as the error signal, a correction signal is obtained.
4. The system may be operated in multiple output modes based on user-supplied input from the DIP switches. The following are the output modes (Abnormal Routine):
    a. Display Corrected Signal (output of PID controller) on Oscilloscope- Using Digital-to-Analog Conversion of Error Signal obtained by finding the difference between the ideal signal and the patient signal.
    b. Display Ideal Signal Model on Oscilloscope
    c. Display Patient Signal Model on Oscilloscope
    d. Run Stepper Motor

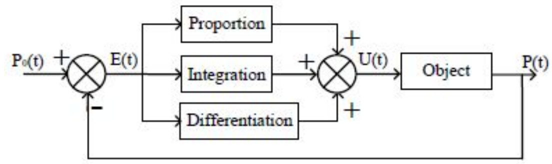
The Control System is implemented using the following iterative formulae:

$$P[n] = P[n-1] + Y[n-1] + (R[n] - R[n-1])$$

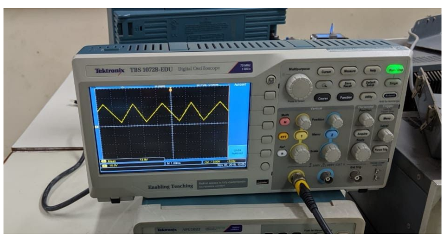
$$\varepsilon[n] = I[n] - P[n];\ E[n] = E[n-1] + \varepsilon[n]$$

$$Y[n] = K_p\varepsilon[n] + K_d(\varepsilon[n] - \varepsilon[n-1]) + K_i(E[n-1] + \varepsilon[n]);\ ----> PID\ feedback \quad control$$

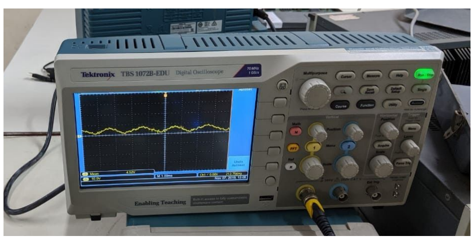




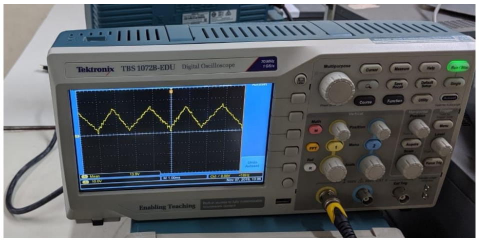
Outputs



Ideal Signal



Patient Signal



Corrected Signal