**EE 337: Microprocessors Laboratory (Spring 2024)**
Indian Institute of Technology Bombay
Lab 10: 25 points Date: March 31, 2024

# Low-Pass Digital FIR Filter using Pt-51

Digital signal processing (DSP) is the use of processing techniques on digital devices (such as micro-processors) to perform a wide variety of signal processing operations.
DSP involves taking real-world signals like voice, audio, temperature, pressure, etc... that have been digitized (using ADC) and then mathematically manipulating them to either extract meaningful information from the signals or to improve the quality of the signals.
DSP provides several advantages over Analog Signal Processing (ASP), such as better noise immunity, higher accuracy, ability to incorporate practically useful techniques like error correction and data compression.

## Implementation Details

In this lab, you will be designing and implementing a low-pass digital FIR filter.
A low-pass filter is a filter that passes signals with a frequency lower than a selected cutoff frequency and attenuates signals with frequencies higher than the cutoff frequency.
An FIR filter is a filter whose output depends only on present and past inputs, but not on past outputs. The general equation for FIR filter is given below:

$$y[n] = a_0x[n] + a_1x[n-1] + a_2x[n-2] + .... + a_{k-1}x[n-k-1]$$

$$\implies y[n] = \sum_{i=0}^{k} a_i x[n-i]$$

here, $y$ represents the output sequence, $x$ is the input sequence, $a_0, a_1, a_2, ...., a_{k-1}$ are filter coefficients. Since there are $k$ coefficients, the number of taps of the filter is $k$.

Steps-

i Designing filter of required cut-off frequency in Python (using SciPy library) and obtaining the filter coefficients. Let sampling frequency be 20 kHz.

ii Establishing connection between PC and Pt-51 using UART protocol using Python (using PySerial library).

iii Generating a sinusoidal waveform in Python (using NumPy library) and sending the samples serially from PC to Pt-51.

iv Implementing filtering algorithm in C to run on Pt-51. First, you will need to store the filter coefficients in a constant array. Then you will need to store the present input sample and past $k-1$ input samples, and then compute the corresponding output sample value using the given equation.

iv Sending the output samples from Pt-51 serially back to the PC.

v Plotting (using Matplotlib library) and analyzing the result on PC. Your plots should indicate that your filter is actually passing low frequency signals and attenuating high frequency signals.

- Components Required-
  - Pt-51 $\mu$C
  - USB-UART Module
- Software/Tools Required-
  - Keil $\mu$Vision
  - FLIP
  - Python (with NumPy, SciPy, Matplotlib, PySerial)

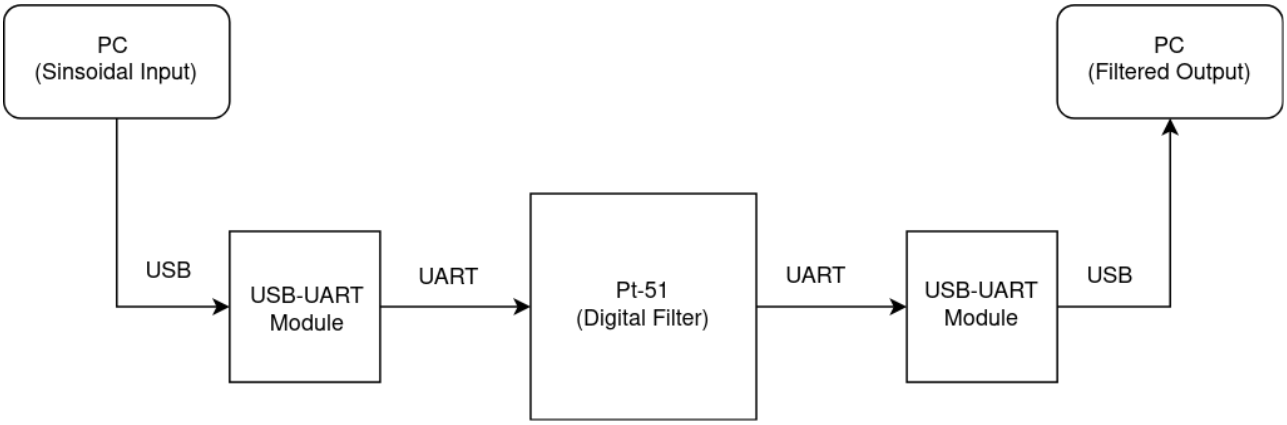


Figure 1: Block Diagram of Overall System

A simple example demonstrating the usage of Python (+libraries) and communication between Pt-51 and PC using Python have been provided. Refer to them for more clarity.

**Marks Distribution**

1. [4 points] Design 8-tap low pass filter of cut-off frequency 2000 Hz using Python. Obtain filter coefficients.
2. [10 points] Implement filtering algorithm on Pt-51.
3. [6 points] Interface Pt-51 with USB-UART module to communicate the input samples and filtered output samples to and from the PC via UART protocol.
4. [5 points] Provide signals of different frequencies, receive the filtered output signals on PC and plot them using Python.