# Workshop on “Embedded C Programming: Attaining Ninja Level”

## Objective

This workshop is aimed at people with fair C programming skills but who may not have programmed an embedded controller for an all inclusive and comprehensive stand-alone application. It is an effort to identify critical aspects of embedded C programming and to train participants in those aspects. Embedded C programming differs from conventional C programming in the aspect of handling asynchronous user and external inputs and to respond to them without losing or skipping input events. The ability of an embedded system to respond in a timely fashion to inputs and to process these inputs and provide appropriate outputs in critical. Delay in recognizing the inputs, several of which may occur together may be unacceptable.

This workshop is aimed to spread awareness about ‘good’ programming practices, documenting the code for future reference and readability, understanding interrupts and processing asynchronous inputs, creating and handling shared data structures etc.

## Approach

The participants are required to implement a specified task on a microcontroller of their choice as part of the workshop. They can use a microcontroller development board and interface the required peripherals that are available on the VOYAGER board. Detailed information on the VOYAGER board is available on request, but the hardware specifications of the peripherals required for the specified task are discussed here.

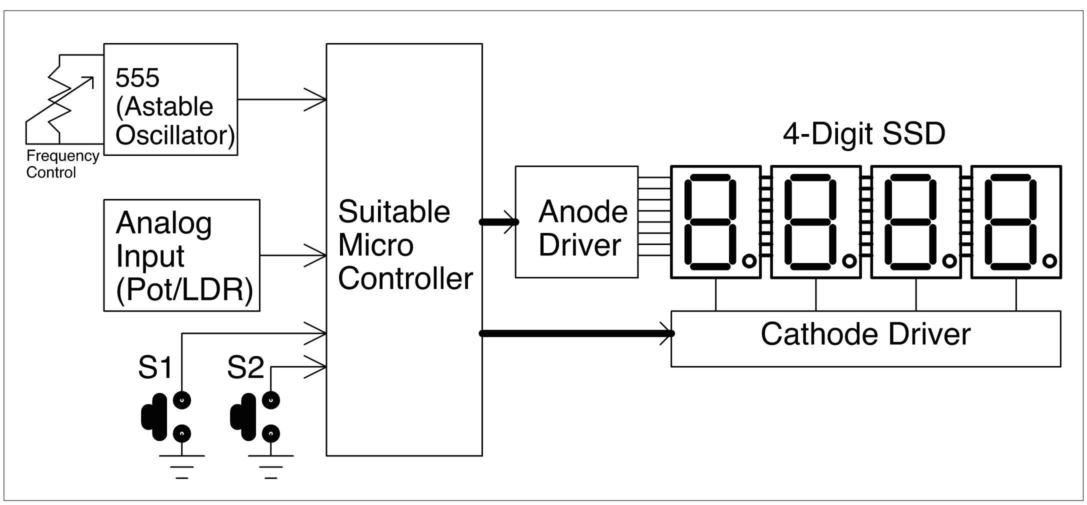



Figure 1: Block diagram of the system.

## System Specifications

A microcontroller based system capable of operating in 5 different modes, as detailed below, is to be designed. The system consists of two user switches, a 4-digit seven segment display (SSD), a 555 timer based astable multivibrator with user adjustable frequency through a potentiometer and an analog input voltage from another potentiometer or a light dependent resistor (LDR) as illustrated in the block diagram in figure 1.
The program will start up running in Mode 1. Each press of the user switch S1, at any point of time, should change the program to the next mode. If S1 is pressed while in Mode 5, it should switch back to Mode 1.
The decimal point of each digit of the 4-digit SSD will be used to indicate the current mode. Modes 1-4 will be indicated by the decimal point of the respective digit. Mode 5 will be indicated by all the decimal points being ON as illustrated in figure 2.

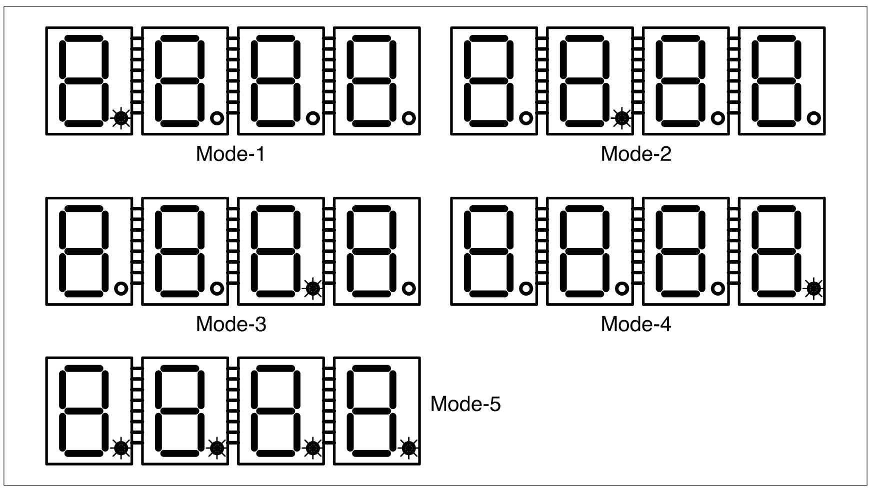



**Figure 2: Mode indication.**

### Operating Modes

**Mode 1:** Display the measured analog value (10-bit / 12-bit) on the 4-digit SSD.
**Mode 2:** Implement a 4-bit hexadecimal counter on the 4-digit SSD. The user switch S2 will be used to increment the counter when pressed. If the switch S2 is pressed and held down longer than 1s, the counter will continuously increment at a reasonably fast rate.
**Mode 3:** Display the frequency generated by the 7555 Astable Multivibrator in Hertz on the 4-digit SSD.
**Mode 4:** Same functionality as Mode 1, where the update rate of the 4-digit SSD is controlled by the frequency generated by the 7555 Astable Multivibrator.
**Mode 5:** Same functionality as Mode 2, where the update rate of the 4-digit SSD is controlled by the frequency generated by the 7555 Astable Multivibrator.

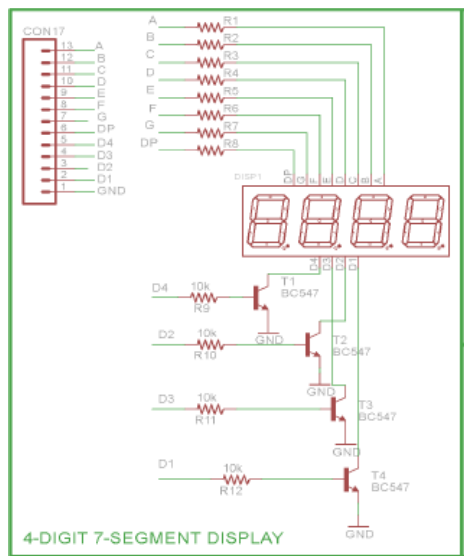
The description of each of the hardware peripherals required for the implementation of the program is available in the "Hardware Description" section.

## Important Instructions

1. The program code must be properly indented (using tab stops and not spacebar) for easy readability.
2. The program code must have comments where necessary.
3. The connections between the microcontroller development board and the peripherals on the VOYAGER board must be clearly mentioned at the start of the program in a multi-line commented section.

## Hardware Description

Several peripherals on board the VOYAGER are used in this exercise. The board allows the user to select between a system supply voltage (Vcc) of 5V or 3.3V depending on the requirement of the microcontroller. Depending on the microcontroller chosen, the appropriate system supply voltage (Vcc) must be selected on the VOYAGER. It must be noted that a common ground connection between the microcontroller development board and the VOYAGER must be made in order for most of the peripherals to work with the microcontroller. While individual GND connections are provided for each of the peripherals that require them, they are interconnected on the board and hence a single GND connection between the microcontroller board and the VOYAGER will suffice.

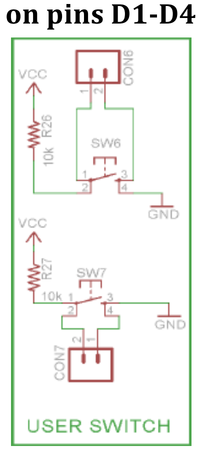



### 4-digit Seven Segment Display

A four digit seven segment display is available on the VOYAGER board. A common cathode display is used on the board. The display requires 12 pins to control; 8 for the segments (A-G and DP) and one each to enable each digit. An NPN transistor circuit is used to enable each digit, as the common cathode must be capable of sinking 8 times the current through each segment.
Time Multiplexing techniques must be used in order to display a four-digit number on the SSD, as only one digit can be turned on at a time.

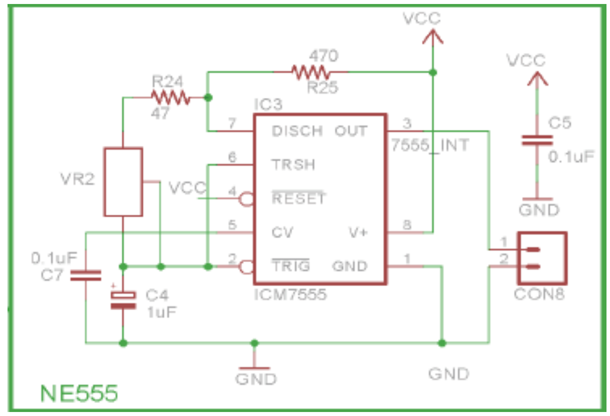
**A logic HIGH on any segment (A-G & DP) turns it ON while a Logic HIGH on pins D1-D4 turns ON the corresponding digit.**

### User Switch




The VOYAGER board has two push button switches for user input. The switches are pulled up using a 10k-ohm pull up resistor to Vcc, the VOYAGER board system supply voltage.

The switches can be connected to any digital I/O pin of the microcontroller. Optionally, they may be connected to pins that support the pin-change interrupt feature.
Push button switches are notorious for the switch bounce phenomenon, and therfore, appropriate debouncing measures should be used.
**The switch is read as HIGH when not pressed and LOW when pressed.**




## 7555 Timer based Astable Multivibrator

A 7555 timer based Astable Multivibrator is present on the VOYAGER board. It produces a variable frequency output, in the range of around 10 Hz to 3000 Hz. The output frequency can be adjusted using the potentiometer provided on the board. The output is a rectangular wave of around 50-70% duty cycle. The voltage level will be same as the system supply voltage (Vcc) selected on the VOYAGER board.
**The output frequency signal can be connected to any digital input pin of the microcontroller. Optionally, it can also be connected to a pin that has pin-change interrupt capability or a one that has input capture capability.**

## Analog Input

A variety of analog input peripherals are available on board the VOYAGER. For this exercise, it is recommended that either the Potentiometer or the LDR input be used, as these provide the most dynamic range of analog voltages. It must be noted that in order to use the analog peripherals on the VOYAGER, an analog reference voltage, AVcc, must be externally supplied to the VOYAGER. Typically, the analog reference voltage on most microcontroller development boards are same as the system Vcc. Appropriate connection of the analog reference voltage must be made from the development board used to the VOYAGER in order to be able to use the analog inputs. The grounds for both analog and digital peripherals are connected together and therefore a single common ground between the VOYAGER board and the microcontroller board will suffice.
**The output from the analog peripheral (POT/LDR) must be connected to an analog input pin of the microcontroller.**

**Good luck!**