

Button Monitor

Rev. 1.0

Author: Donald Walters

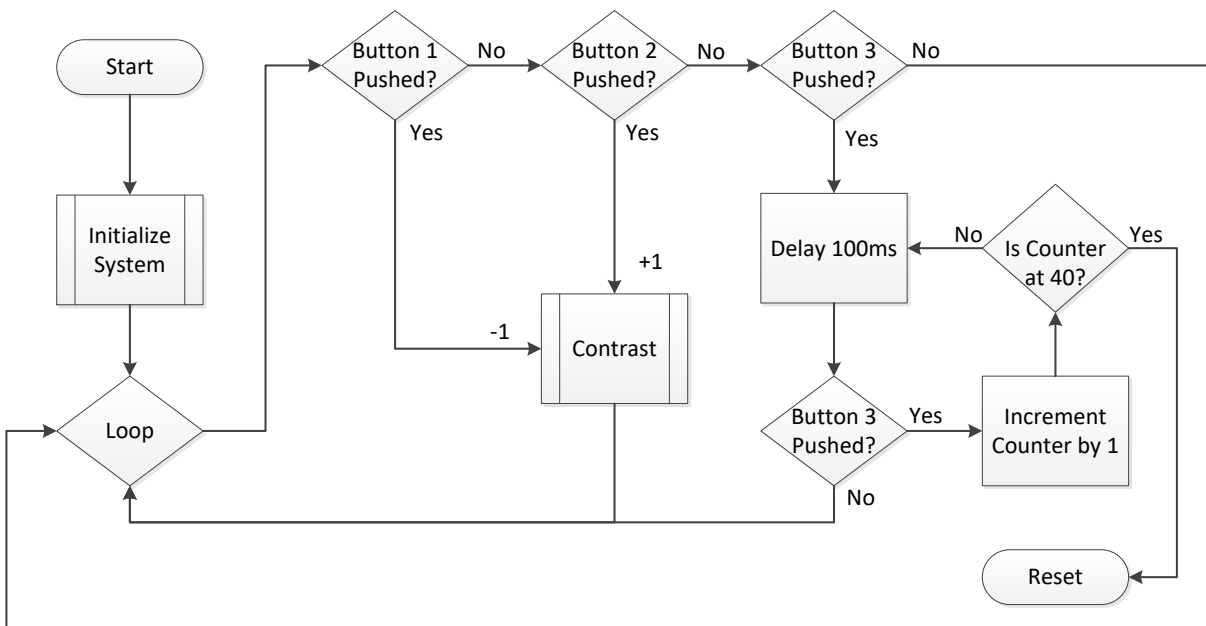
Date: 3/7/2013

Introduction

Button Monitor is a segment of code that resides in the 'main' function to support user interactions. Three buttons are provided to the user by the embedded system. The buttons are used for system reset and contrast control of an attached LCD Panel.

Design Concept

The first operation is a call to initialize the system which sets up the internal processor hardware to support the buttons. After that, the main loop is entered and continuous monitoring of the buttons begins. A 'button()' function is used to return a byte value for which each of the 3 least significant bits represent the three buttons (bit 0 for button 1, bit 1 for button 2, and bit 2 for button 3). If button 1 or 2 is pushed, then the contrast function is called with a respective decrementing or incrementing value. If button 3 is pushed, a secondary loop initiates to determine if the button is held down for 4 seconds. After 4 seconds, the call to reset the processor is made and the initialization process starts over. If at any time the operator release button 3 before 4 seconds has passed, the system returns to the main loop operations. The operations for the Button Monitor are represented in the flowchart below.



Implementation

The Button Monitor requires the following prototypes and their associated defined functions.

```
void initSystem(); // initialization to enable supporting hardware for the buttons
void contrast(int level); // negative value to decrease, positive to increase
unsigned int button(); // Each bit in byte returned represents a pressed button
```

The following is code segment for the reset button:

```
if(button() & 0x04 > 0){
    int count = 0;
    while((button() & 0x04) > 0){
        delay(100ms);
        count++;
        if(count >= 40){
            reset();
        }
    }
}
```

Conclusion

The Button Monitor provides basic setup, contrast control, and system reset for an embedded system utilizing an LCD display panel. Any additional operations to be added must be done within the main loop. Operations that are required before a manual system reset (i.e.: logging the reset) needs to be before 'reset()' is called within the 'if(count >= 40)' statement.