

Experiment 4

ARM C-Interfacing - Emulation of Switch LED and Stepper Motor Control

Target:

- Implementing the given tasks using ARM C-interfacing
- Read the status of the switch and use the LEDs (8 LEDs) to display the status of each of the 8-bit DIP switches.
- Stepper motor control using ARM C-programming.(Emulation in KEIL software)

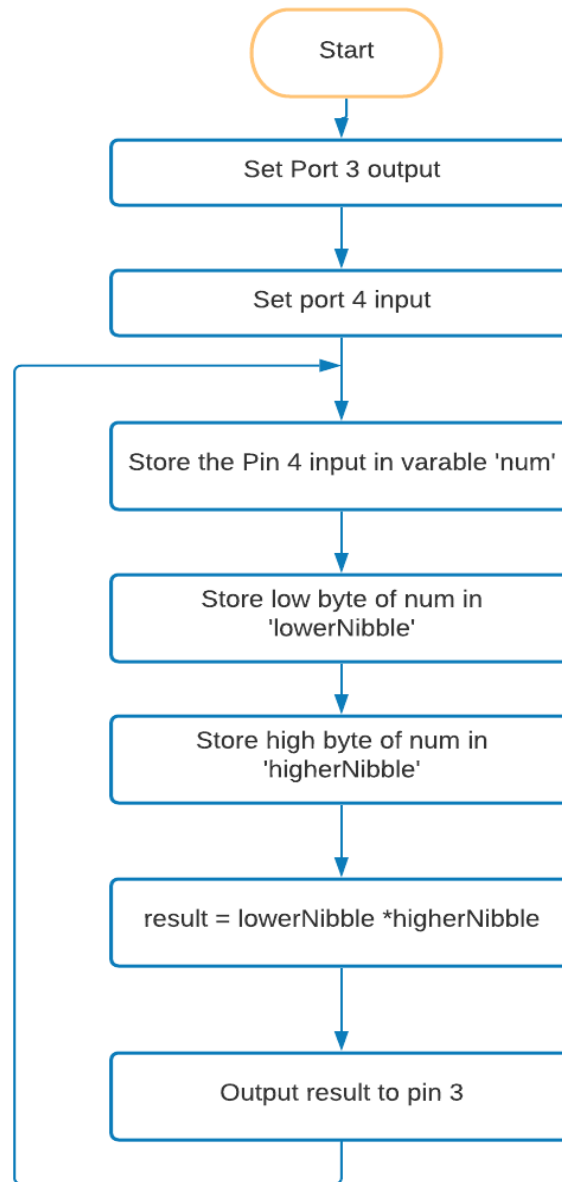
Questions:

1. Write a program (in C) to dis-assemble a byte into two nibbles from the DIP switch states, multiply and display the product in the LED.
2. Modify the demo code (StpprMtrCntrl.c) supplied to demonstrate the control of stepper motor to rotate in opposite direction.
3. Modify the demo code (StpprMtrCntrl.c) again to rotate the stepper motor by a specific angle(say 40°).

Solutions:

Question 1

flowchart:



Code:

```
#include "LPC23xx.h"
int main()
{
    int num;
    int result;
    int lowerNibble;
    int higherNibble;
    FIO3DIR = 0xFF;    //FIO3 is output port
    FIO4DIR = 0x00;    //FIO4 is input port

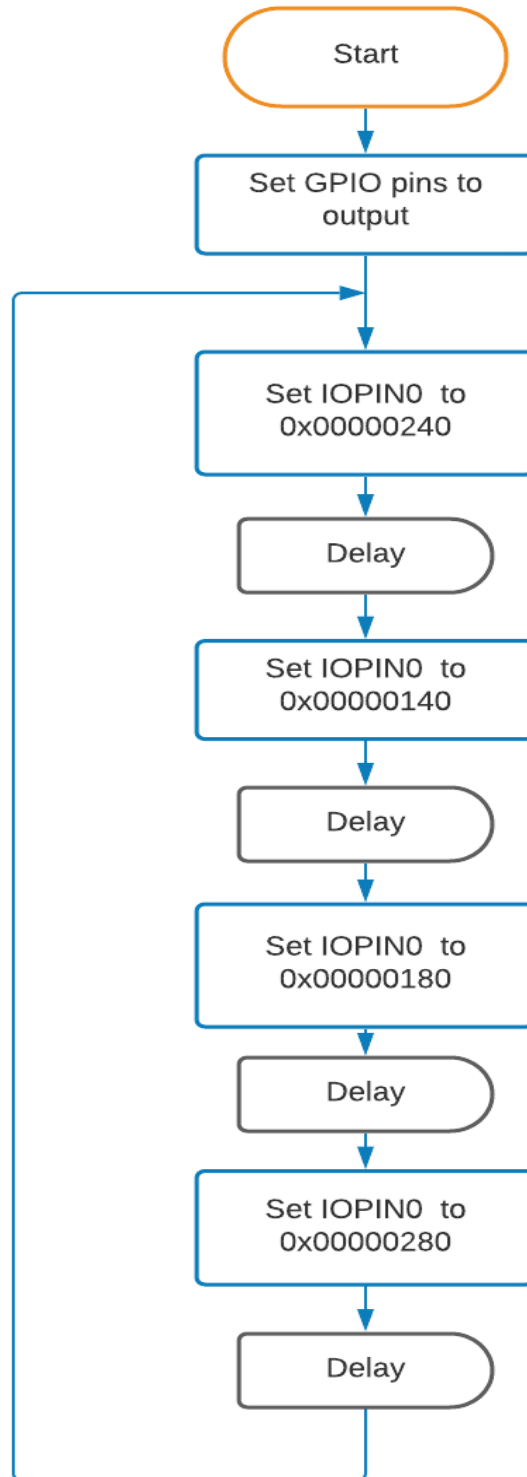
    while(1)
    {
        num = FIO4PIN;
        lowerNibble = num & 0x0F;
        higherNibble = num & 0xF0;
        higherNibble = higherNibble >> 4;
        result = higherNibble * lowerNibble;
        FIO3PIN = result;
    }
    return 0;
}
```

Observation:

We can read the status of DIP switch states and display the result in the LED(after, multiplying higher and lower bytes)

Question 2

Flowchart:



Code:

```
#include "LPC23xx.h"

void delay(void)
{ int i,j;
  for(i=0; i<0xFF; i++)
    for(j=0; j<0xFF ; j++);
}

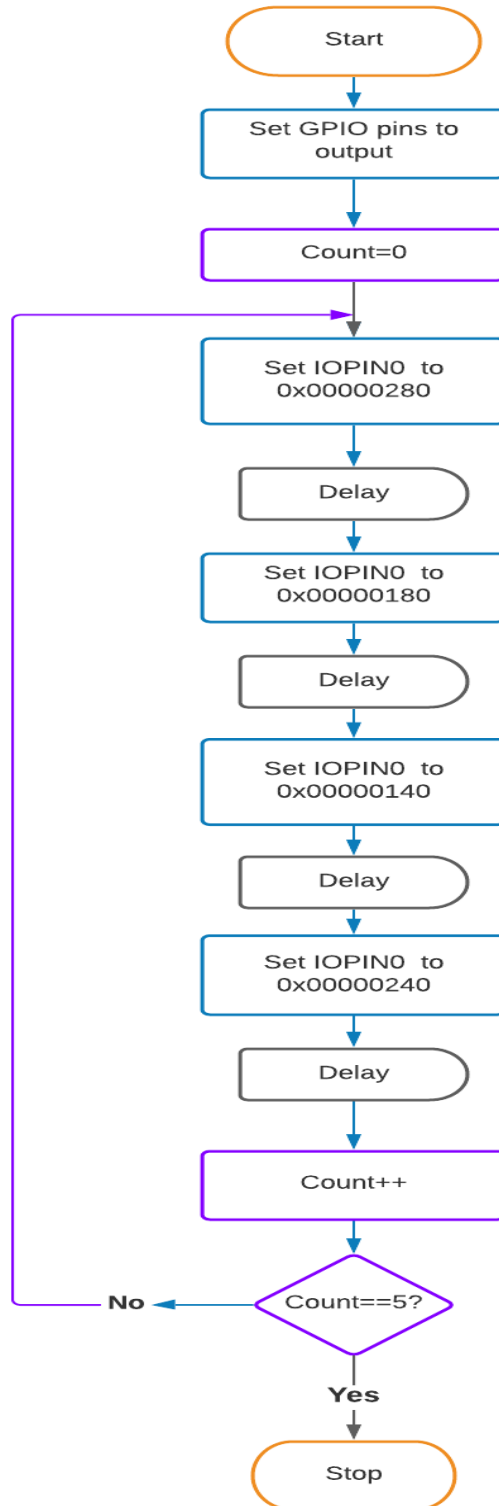
int main(void)
{
  IODIR0 = 0xFFFFFFFF;
  for(int k=0;k<5;k++)
  { IOPIN0 = 0x00000240;
    delay();
    IOPIN0 = 0x00000140;
    delay();
    IOPIN0 = 0x00000180;
    delay();
    IOPIN0 = 0x00000280;
    delay();
  }
}
```

Observation:

The stepper motor will rotate in opposite direction.

Question 3

Flowchart:



Code:

```
#include "LPC23xx.h"
void delay(void)
{ int i,j;
  for(i=0; i<0xFF; i++)
    for(j=0; j<0xFF ; j++);
}

int main(void)
{
  IODIR0 = 0xFFFFFFFF;
  for(int k=0;k<5;k++)      //given stepp angle is 2°
  { IOPIN0 = 0x00000280;    //for each for loop motor
    delay();                //will rotate 8°,so after 5
    IOPIN0 = 0x00000180;    //for loops, motor will
    delay();                //rotate 40° and then stop
    IOPIN0 = 0x00000140;
    delay();
    IOPIN0 = 0x00000240;
    delay();
  }
}
```

Observation:

The stepper motor rotates up to 40° and stop.

Inferences:

- Learnt C-interfacing of ARM processor.
- Learnt about DIP switch and their uses.
- Learnt how to read the status of DIP switches and display values using the LEDs.
- Learnt the working and the underlying principle of Stepper motor.
- Learnt how to connect the stepper motor to the processor and control it using ARM C-interfacing.
- Learnt how to rotate stepper motor in clockwise and anti-clockwise directions
- Learnt how to rotate it to a fixed angle and control speed by changing the delay using ARM C-interfacing.
- Learn about different types of I/O pins in ARM processor.