# SNS COLLEGE OF TECHNOLOGY

**Coimbatore-35**
**An Autonomous Institution**

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A++' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

# DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

## 19ECB211 – Microcontroller Programming & Interfacing

II YEAR/ IV SEMESTER

UNIT 5 – Advanced Microcontrollers

TOPIC 4 – Inputs and Outputs of  MSP430

# INPUTS AND OUTPUTS OF MSP430

# INTRODUCTION

The MSP430 is a 16-bit microcontroller that has a number of special features not commonly available with other microcontrollers: - Complete system on-a-chip — includes LCD control, ADC, I/O ports, ROM, RAM, basic timer, watchdog timer, UART, etc.

MPI/Dr.B.Sivasankari/Professor/ECE/SNSCT

# GENERAL PURPOSE INPUT OUTPUT (GPIO)

Each port is assigned several 8-bit registers that control the function of the pins and provides information on their current status.

• PxSEL and PxSEL2 – These registers selects whether the pin operates in GPIO mode or is used for a specialized function as described in the pinout. PxSEL2 is not always available and is used to augment the number of multiplex options if the pin has various specialized functions. Setting PxSEL to 0 selects GPIO mode.

# GPIO

• **PxDIR** – If the pin is set to operate as GPIO, the bits in this register select whether a pin is a high impedance input (0) or an output(1).

• **PxOUT** – If the pin is set to operate as GPIO Output, this pin selects High (1) or Low (0) output.

• **PxIN** – If the pin is set to operate as GPIO Input, this pin indicates whether the voltage at the pin is High(0) or Low(0).
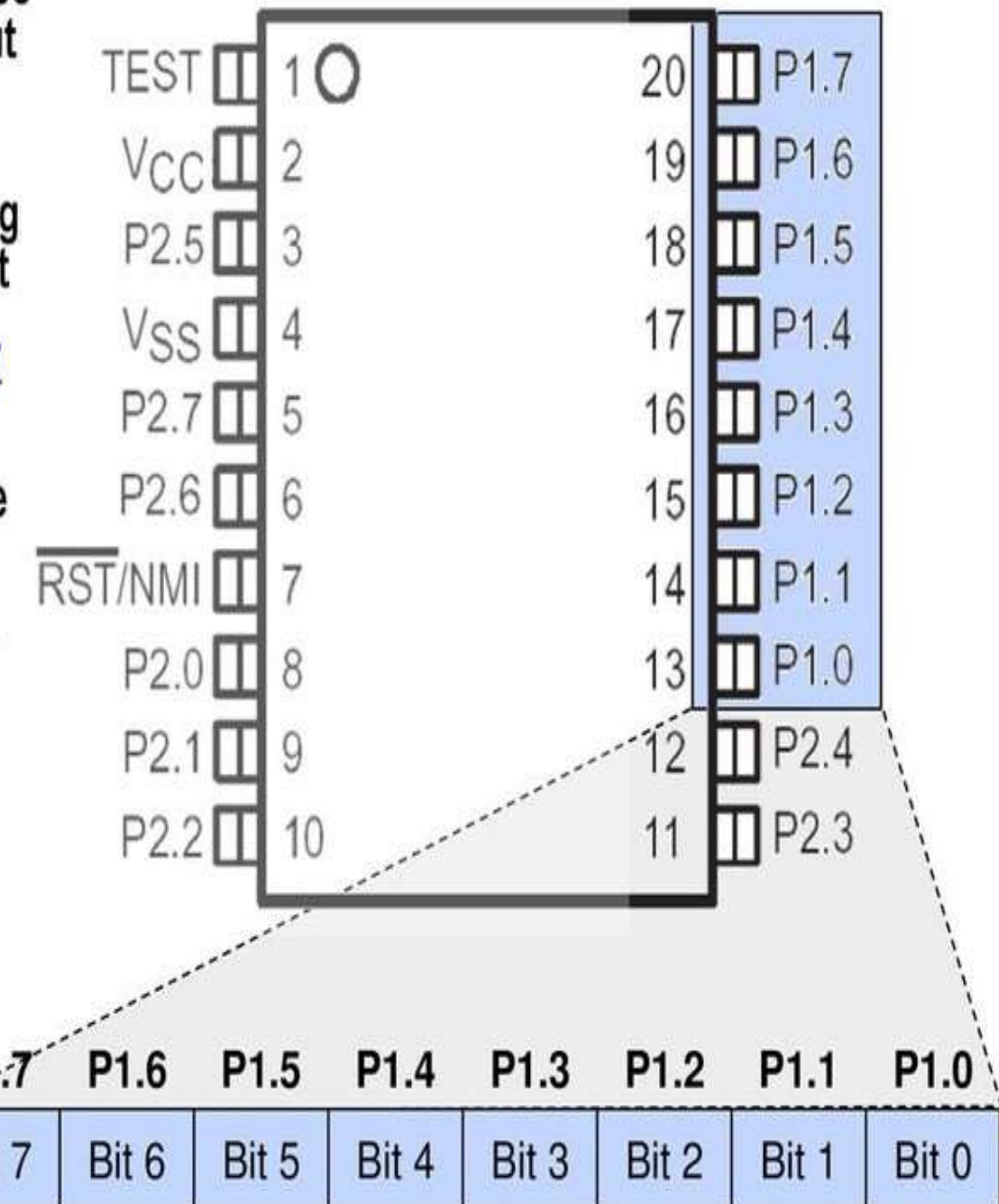
# INTERRUPT CAPABILITY



Some GPIOs in the MSP430 have the capability to generate an interrupt and inform the CPU when a transition has occurred.
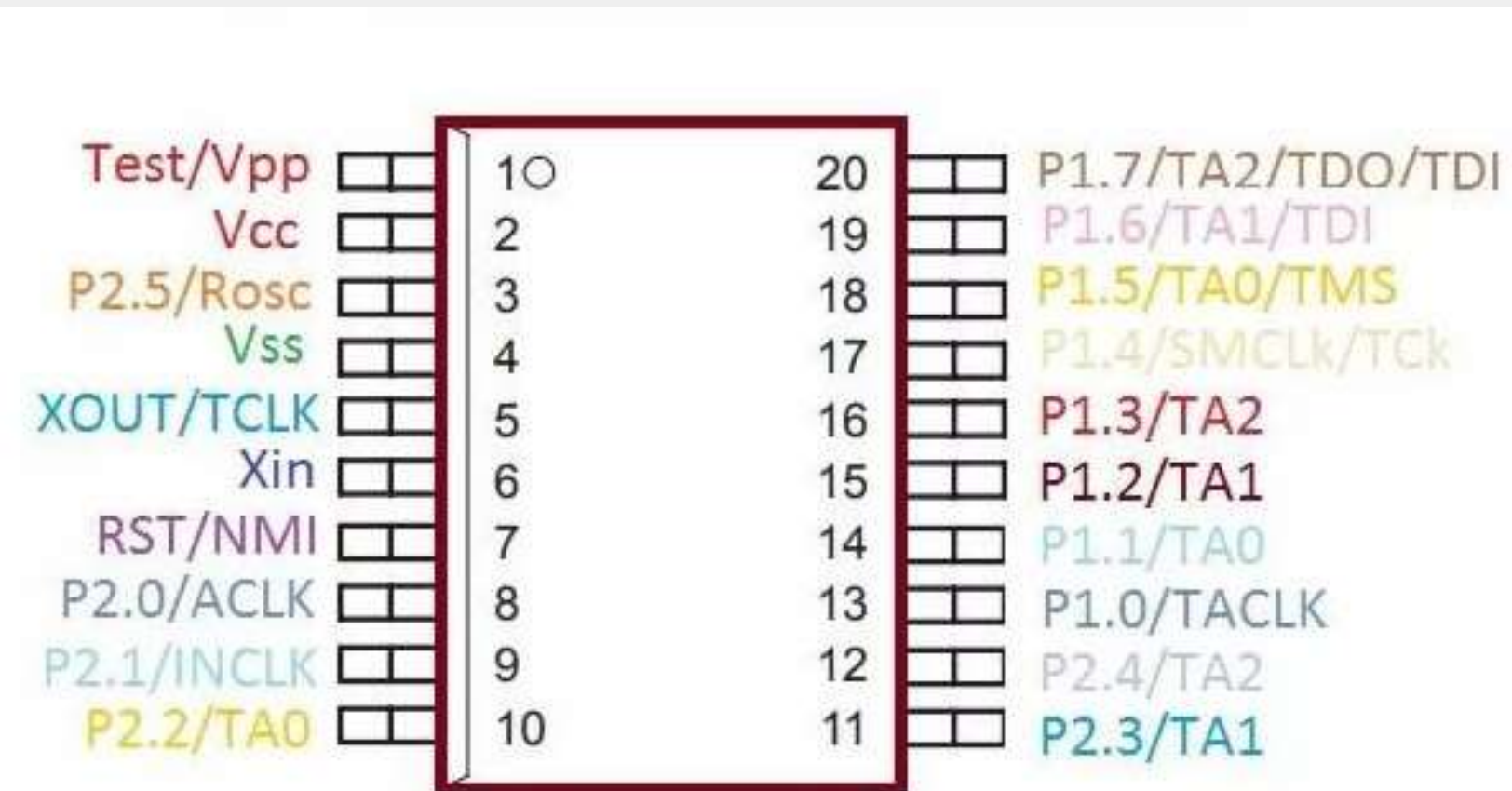
• The MSP430 allows flexibility in configuring which GPIO will generate the interrupt, and on what edge (rising or falling). The registers controlling these options are as follows:

• PxIE – Each bit enables (1) or disables (0) the interrupt for that particular pin.

• PxIES – Selects whether a pin will generate an interrupt on the rising-edge (0) or the falling-edge (1)

• PxIFG – Interrupt flag register indicating whether an interrupt has occurred on a particular pin.
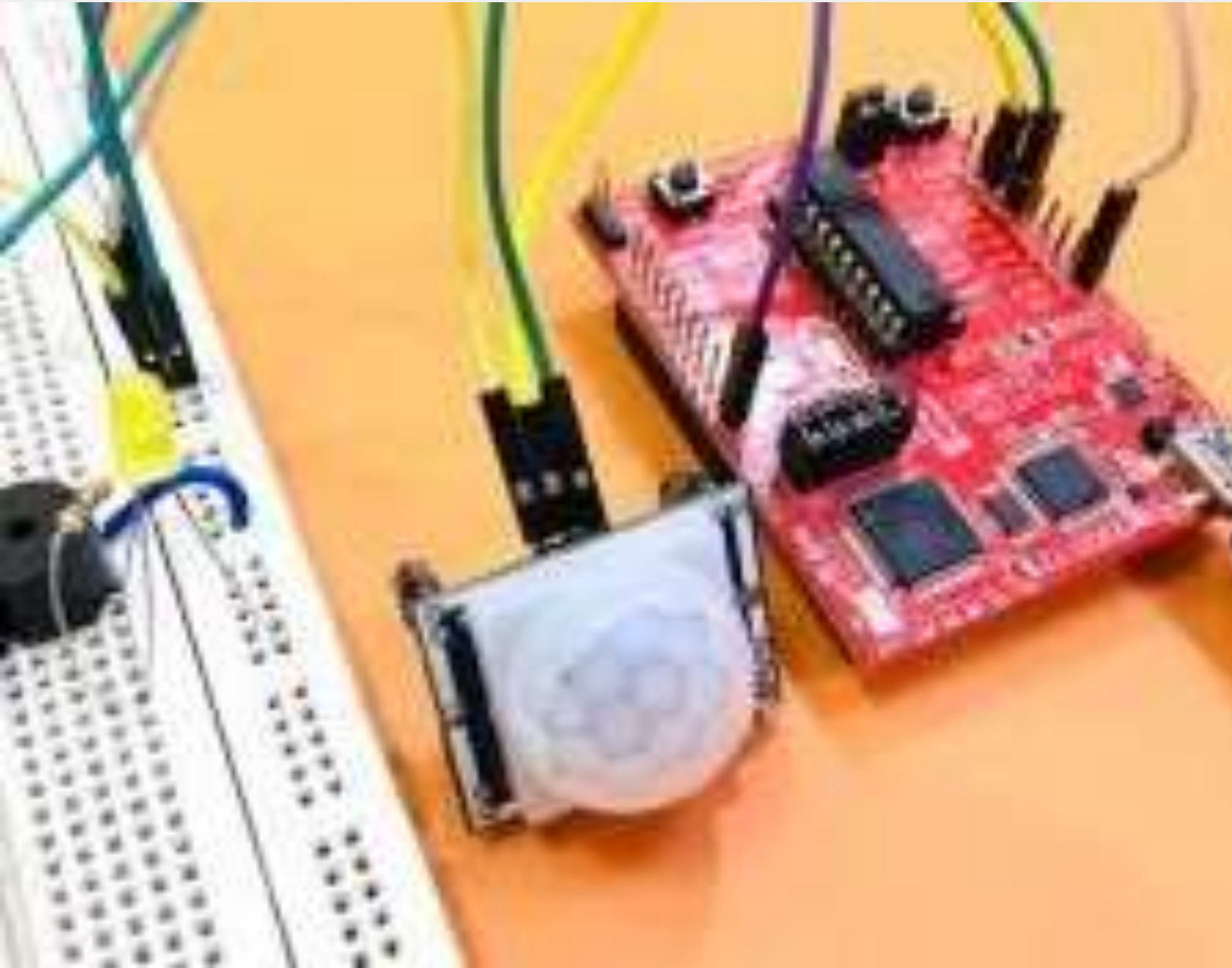
# ELECTRICAL SPECIFICATIONS

The reality is that a microcontroller generates a finite voltage representing '1' and '0'. Except for MSP430 with dual voltage rail capability, the HIGH (1) generated by the GPIO will be close to VCC. The MSP430 like most devices has a limit on the amount of current that can be drawn from the pins, both individually and as a whole.

For example:        With 6mA of current draw at a pin of the MSP430 specifies that the following should hold:

• VOH – Output Voltage High Level VCC-0.3V
• VOL – Output Voltage Low Level VCC+0.3V

# INITIALIZING GPIO

The following example shows how to configure the GPIO as well as use the pin muxing:
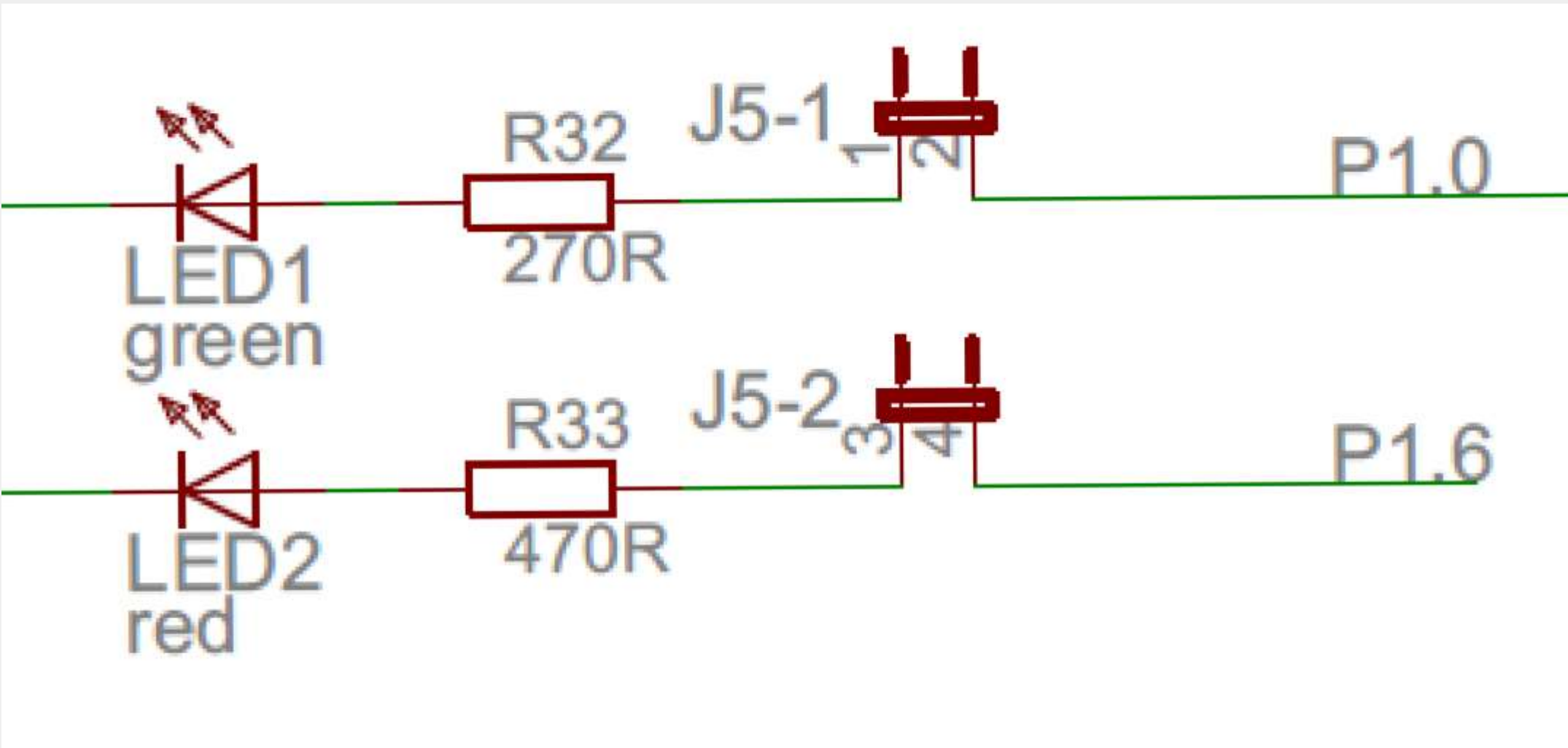
Configuring P1.0 as a GPIO Output set to High(1)

Coding:

```
#include <msp430.h>
void main()
{
WDTCTL = WDTPW + WDTHOLD; // Stop watchdog timer
P1SEL &= (~BIT0); // Set P1.0 SEL for GPIO
P1DIR |= BIT0; // Set P1.0 as Output
P1OUT |= BIT0; // Set P1.0
 HIGH
}
```

# L E D



An LED can be connected directly to the MSP430 pin.

When the pin is low, the voltage across the LED is almost zero, so it remains off. But, when the pin is high, the voltage causes the LED to turn on.

A microcontroller is typically better at sinking current than sourcing it. LEDs can be connected between VCC and the MSP430: The polarities are then reversed. When the MSP430 pin is high, the LED is off because the potential difference between the LED terminals is very small. When the MSP430 pin is set to '0' (Ground), the potential difference is large and the LED turns on.

# POWERUP DEFAULTS

Upon powerup of the MSP430, before any code is executed, the registers controlling the MSP430 are cleared to defaults. This includes PxSEL set to 0, along with PxDIR, meaning that pins are generally configured as inputs with high impedance.

# THANK YOU