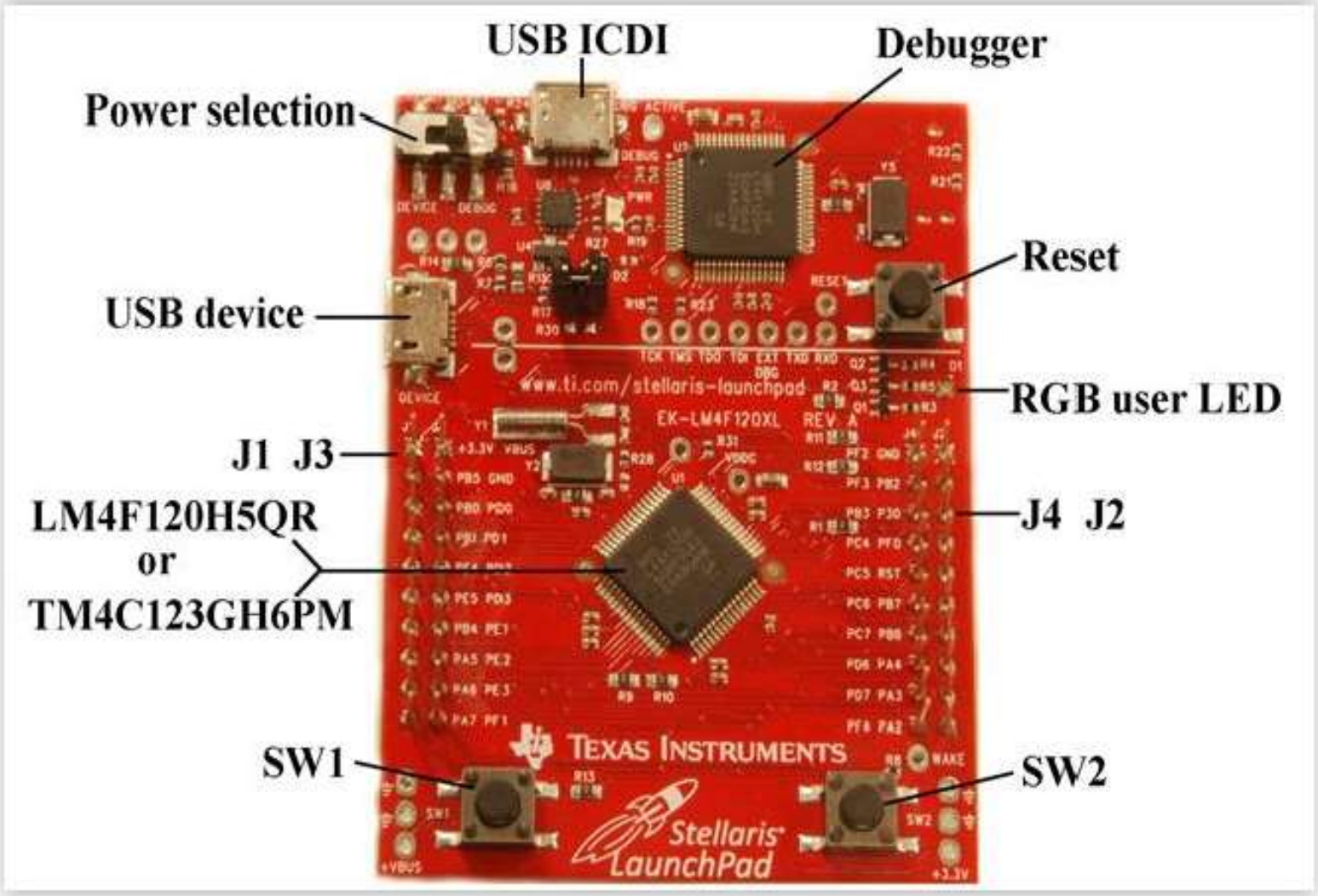
Embedded Lab - 1

Tiva C Series TM4C123GH6PM Port Initialization

TIVA TM4C123G LaunchPad

- 32-bit ARM® Cortex™-M4
- System clock frequency up to 80 MHz
- E-book Link: http://users.ece.utexas.edu/~valvano/Volume1/E-Book/



Port Initialization for GPIO usage

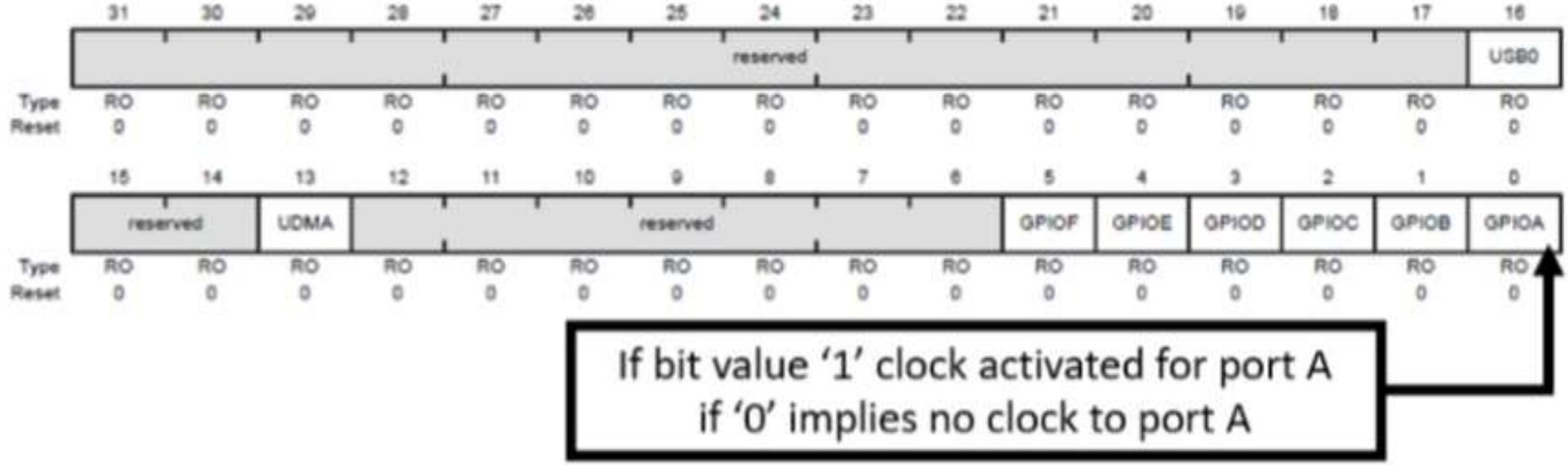
```
// Making PA2 an output
void port_Init(void)
{
   volatile unsigned long delay;
   SYSCTL_RCGC2_R |= 0x01;             // activate clock for Port A
   delay = SYSCTL_RCGC2_R;             // allow time for clock to start
   GPIO_PORTA_PCTL_R &= ~0x0000000F; // regular GPIO
   GPIO_PORTA_AMSEL_R &= ~0x04;        // disable analog function on PA2
   GPIO_PORTA_DIR_R |= 0x04;           // set direction to output
   GPIO_PORTA_AFSEL_R &= ~0x04;        // regular port function
   GPIO_PORTA_DEN_R |= 0x04;           // enable digital port
}
```

Port Initialization for GPIO usage

```
// Making PD3, PD0 an output
void PortD_Init(void)
{
  volatile unsigned long delay;
  SYSCTL_RCGC2_R |= 0x00000008;  // 1) activate clock for Port D
  delay = SYSCTL_RCGC2_R;        // allow time for clock to start
  // only PF0 needs to be unlocked, other bits can't be locked
  GPIO_PORTD_AMSEL_R &= ~0x00;   // 3) disable analog on PD
  GPIO_PORTD_PCTL_R = 0x00000000;// 4) PCTL GPIO on PD
  GPIO_PORTD_DIR_R |= 0x09;      // 5) PD30, PD3 as output
  GPIO_PORTD_AFSEL_R &= ~0x09;   // 6) disable alt funct on PD0, PD3
  GPIO_PORTD_DEN_R = 0x09;       // 7) enable digital I/O on PD0, PD3
}
```

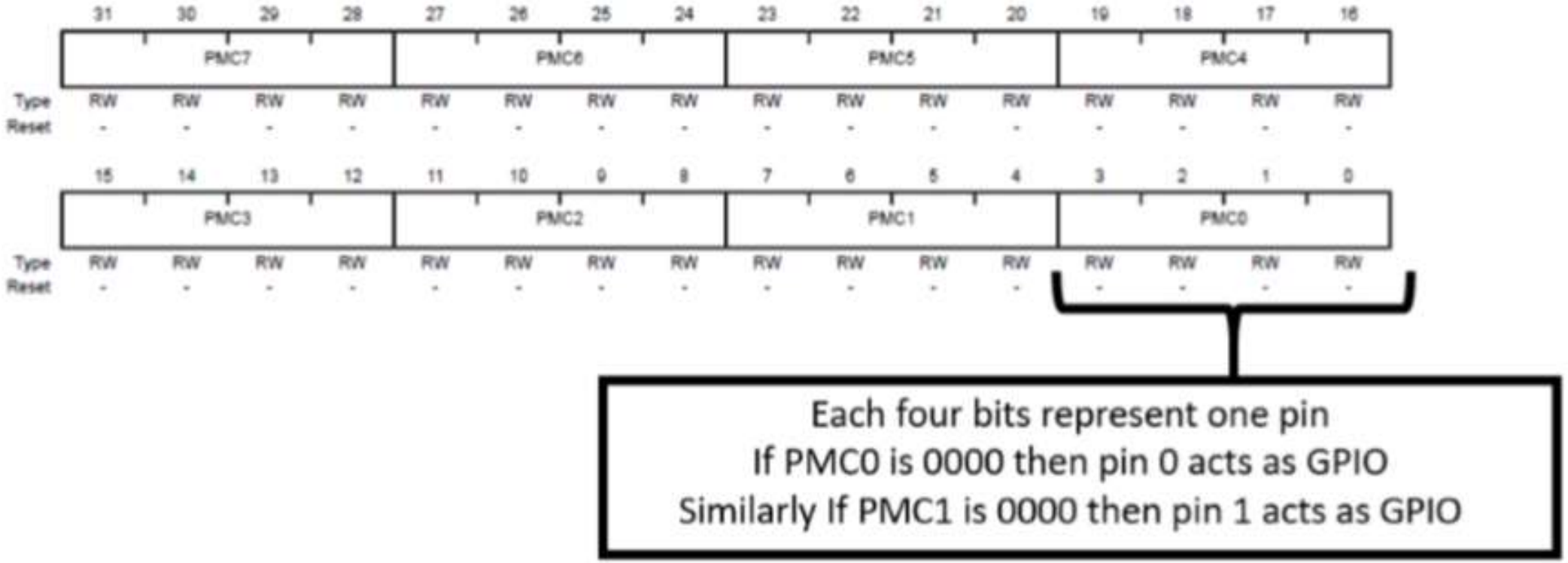
System Control Run mode Clock Gating Control 2 (SYSCTL_RCG2)

RCGC2: pg-462



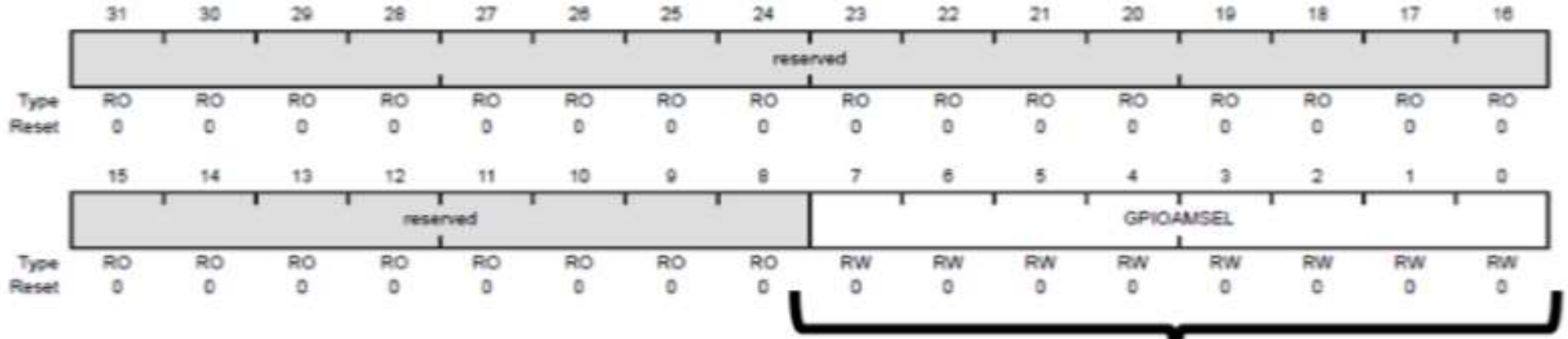
PCTL

PCTL : pg-686



AMSEL

AMSEL : pg-684

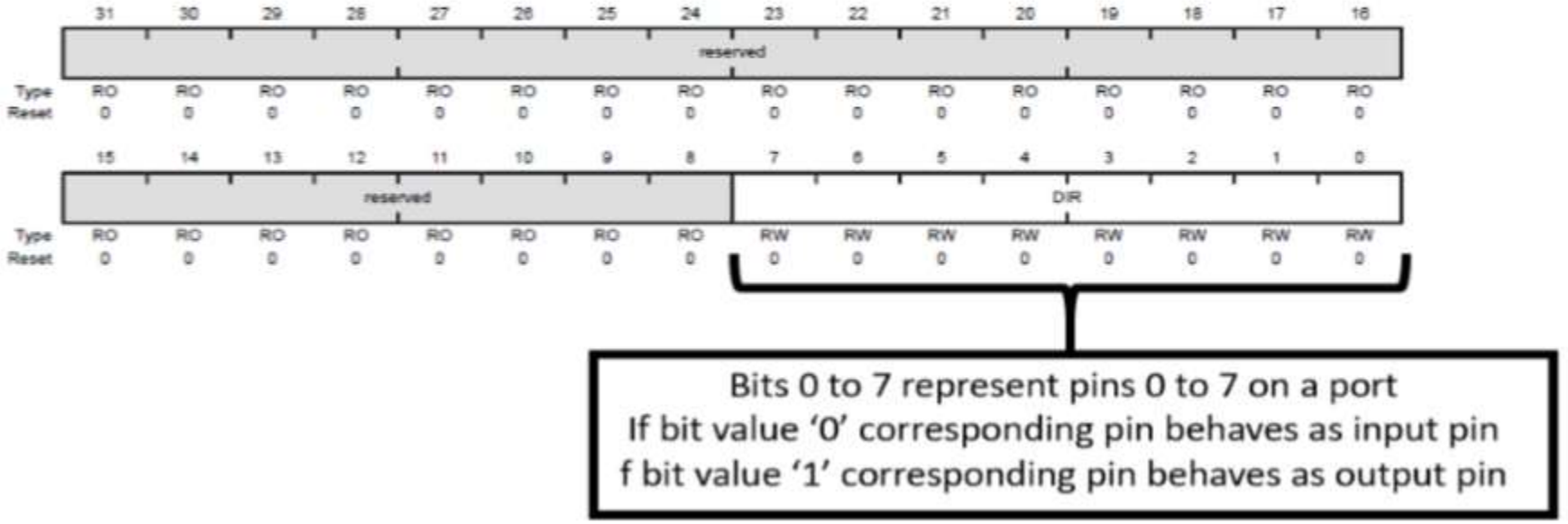


Bits 0 to 7 represent pins 0 to 7 on a port
If bit value ‘0’ corresponding pin works as non-Analog pin
f bit value ‘1’ corresponding pin behaves as analog pin

DIR

- We set the direction register (e.g., GPIO_PORTF_DIR_R) to specify which pins are input and which are output.

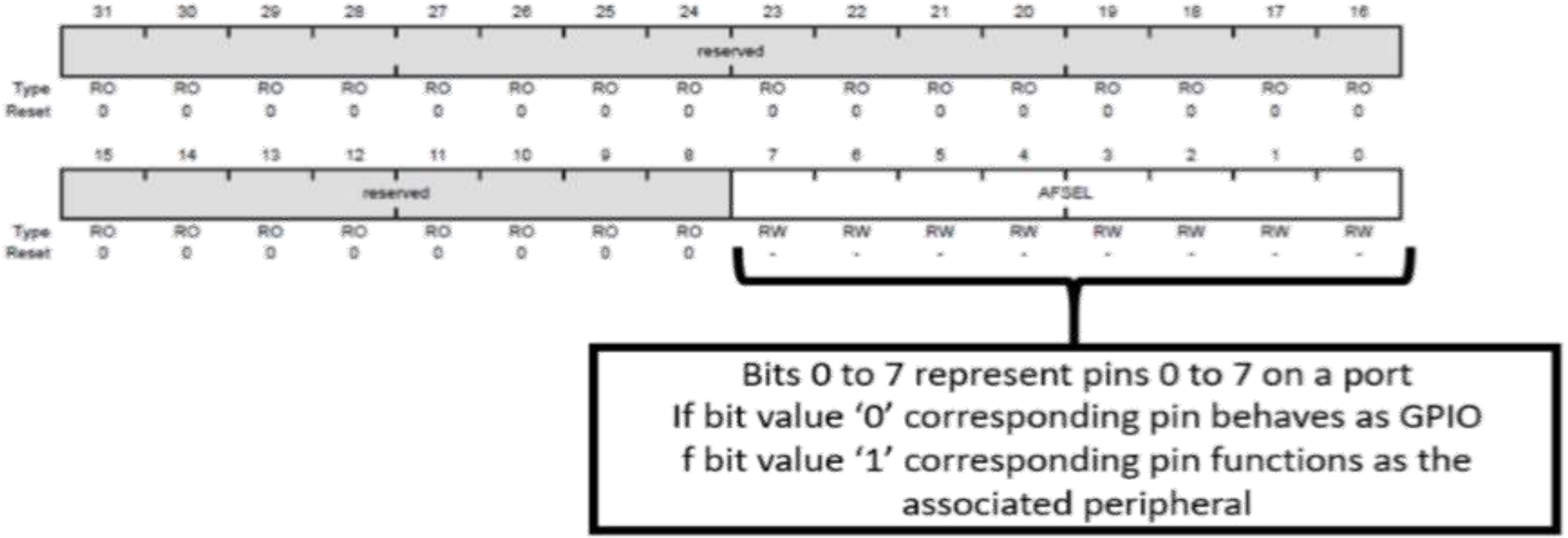
DIR: pg-660



AFSEL

- Individual port pins can be general purpose I/O (GPIO) or have an alternate function. We set bits in the alternate function register (e.g., GPIO_PORTF_AFSEL_R) when we wish to activate the alternate functions.

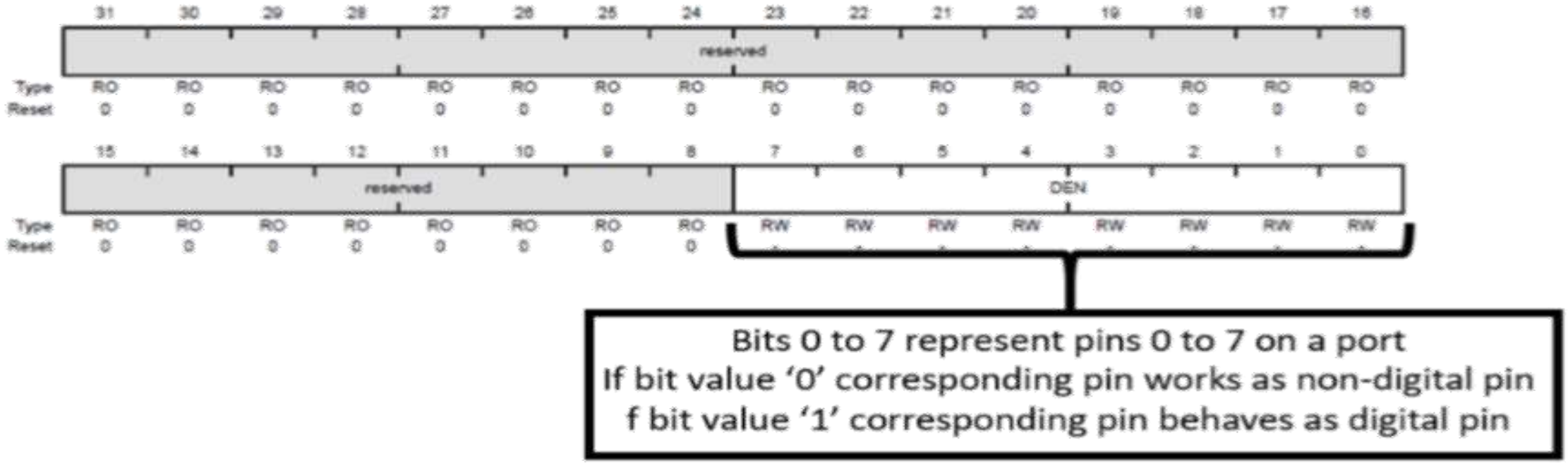
AFSEL : pg-669



DEN

- For each I/O pin we wish to use whether GPIO or alternate function, we must enable the digital circuits by setting the bit in the enable register (e.g., GPIO_PORTF_DEN_R).

DEN: pg-680

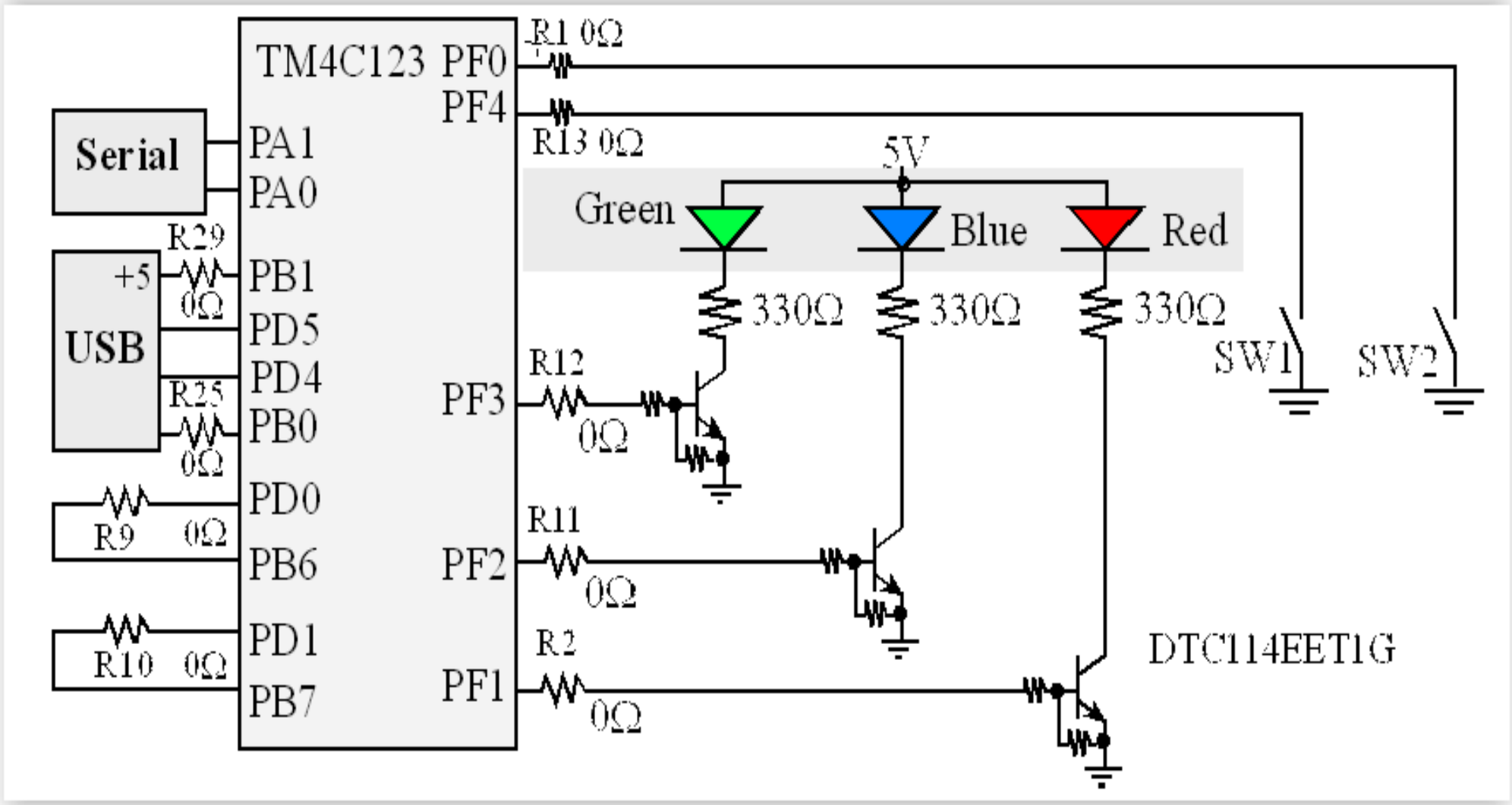


Basic Bit-wise Operations for Register Access

- OR |
  - E.g., REG |= 0x08
  - Means, REG = REG | 0x08. Sets bit 3 (0x08h= 1000 binary) to 1 without changing the other bits. Handy to set to 1 just the bits you want

- AND &
  - E.g., REG &= ~(0x08)
  - Means, REG = REG & ~(0x08) = REG & ~(0000 1000) = REG & (1111 0111). Sets bit 3 to 1 without changing the other bits. Handy to set to 0 (clear) just the bits you want

Switch and LED interface



Port F intialization

```
void PortF_Init(void)
{
   volatile unsigned long delay;
   SYSCTL_RCGC2_R |= 0x00000020;     // 1) activate clock for Port F
   delay = SYSCTL_RCGC2_R;           // allow time for clock to start
   GPIO_PORTF_LOCK_R = 0x4C4F434B;   // 2) unlock GPIO Port F
   GPIO_PORTF_CR_R = 0x1F;           // allow changes to PF4-0
   // only PF0 needs to be unlocked, other bits can't be locked
   GPIO_PORTF_AMSEL_R = 0x00;        // 3) disable analog on PF
   GPIO_PORTF_PCTL_R = 0x00000000;   // 4) PCTL GPIO on PF4-0
   GPIO_PORTF_DIR_R = 0x0E;          // 5) PF4,PF0 in, PF3-1 out
   GPIO_PORTF_AFSEL_R = 0x00;        // 6) disable alt funct on PF7-0
   GPIO_PORTF_PUR_R = 0x11;          // enable pull-up on PF0 and PF4
   GPIO_PORTF_DEN_R = 0x1F;          // 7) enable digital I/O on PF4-0
}
```