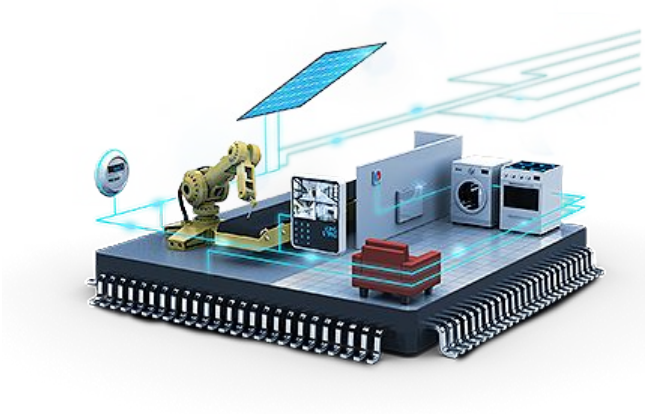
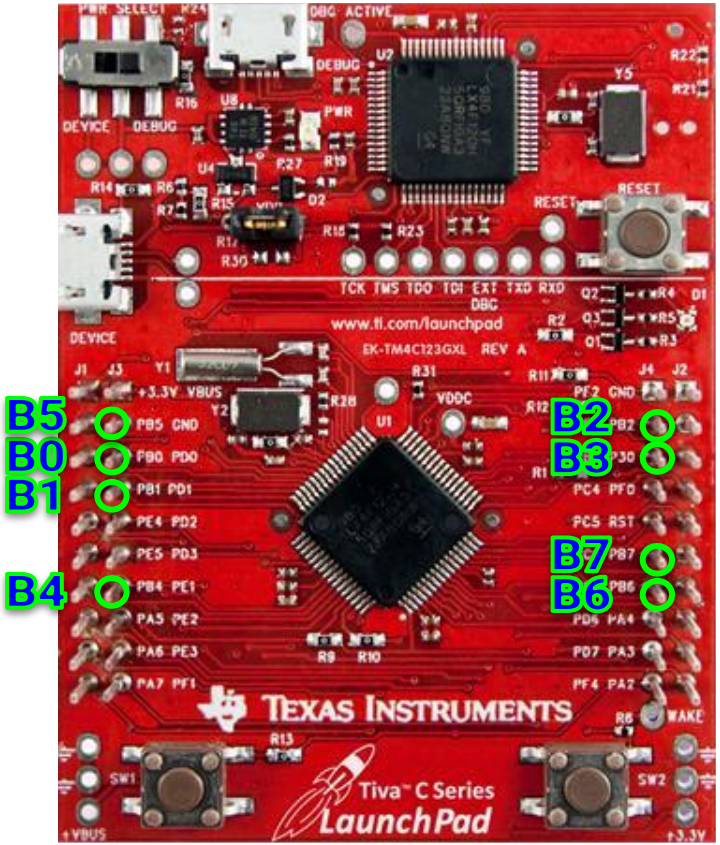
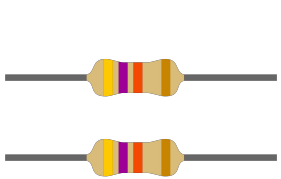
7 Parçalı Gösterge

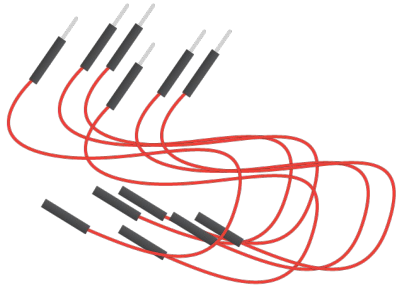


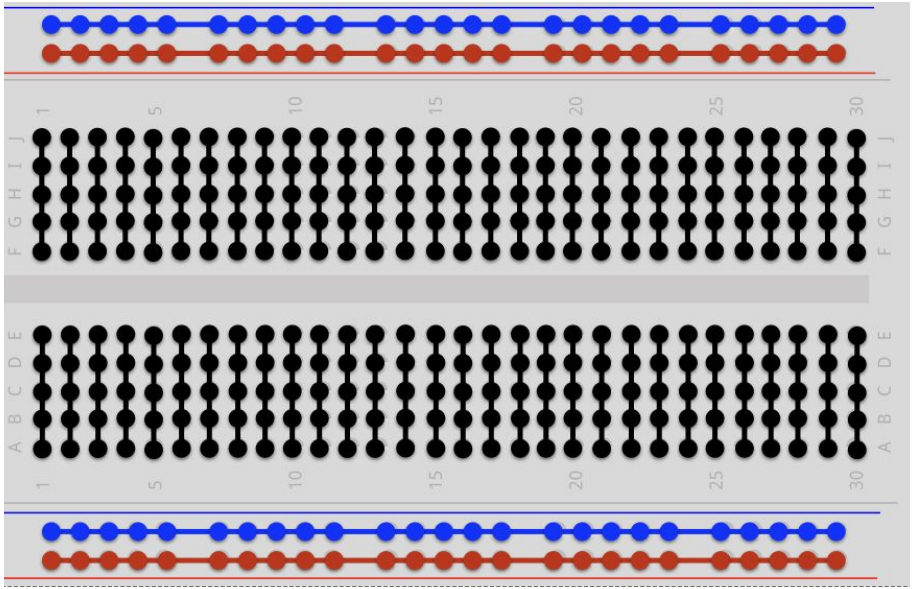
Suhap SAHIN

7 Parçalı Gösterge

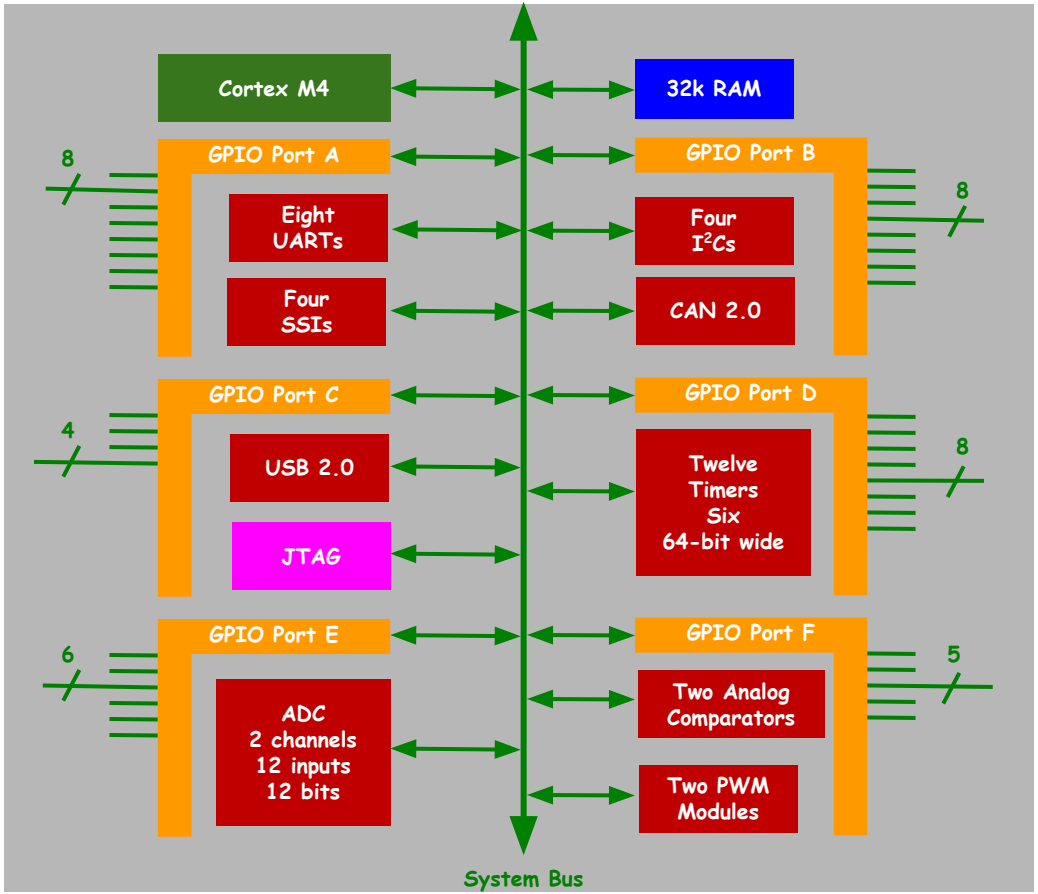




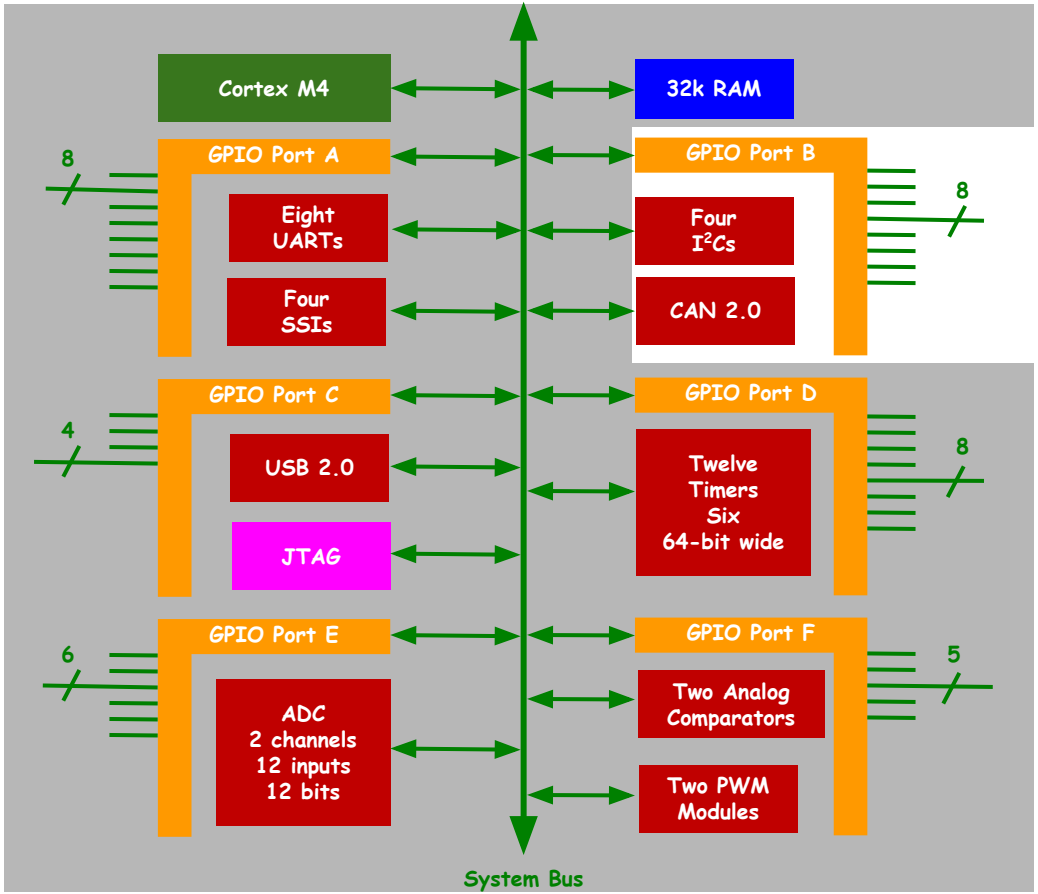




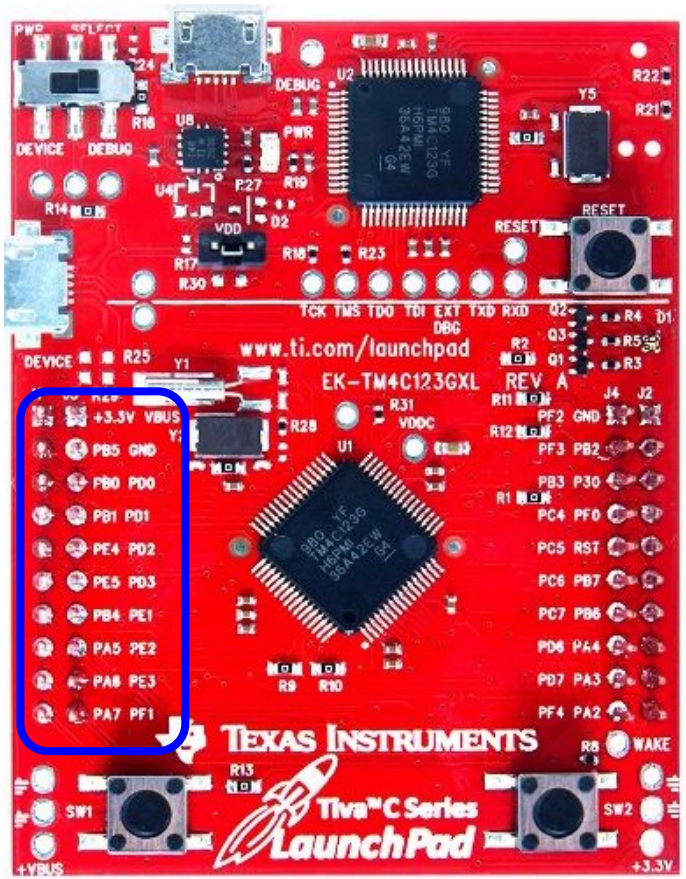
7 Parçalı Gösterge

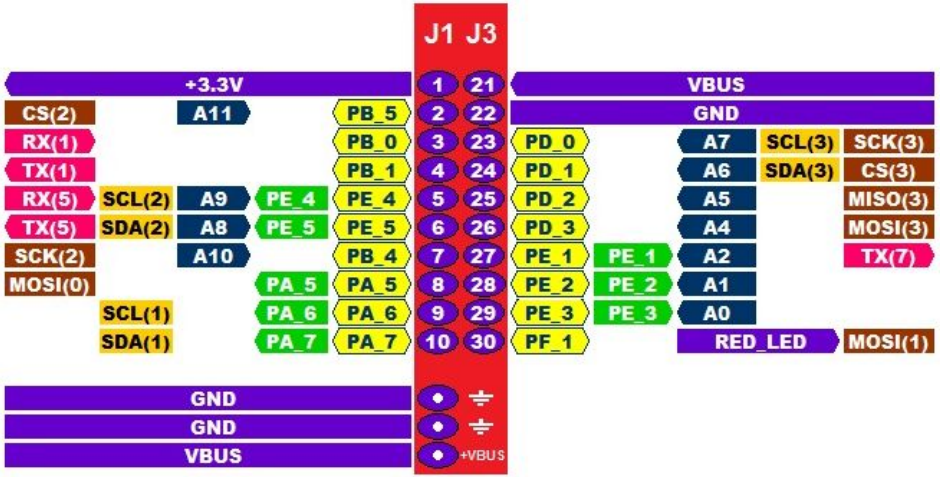


7 Parçalı Gösterge

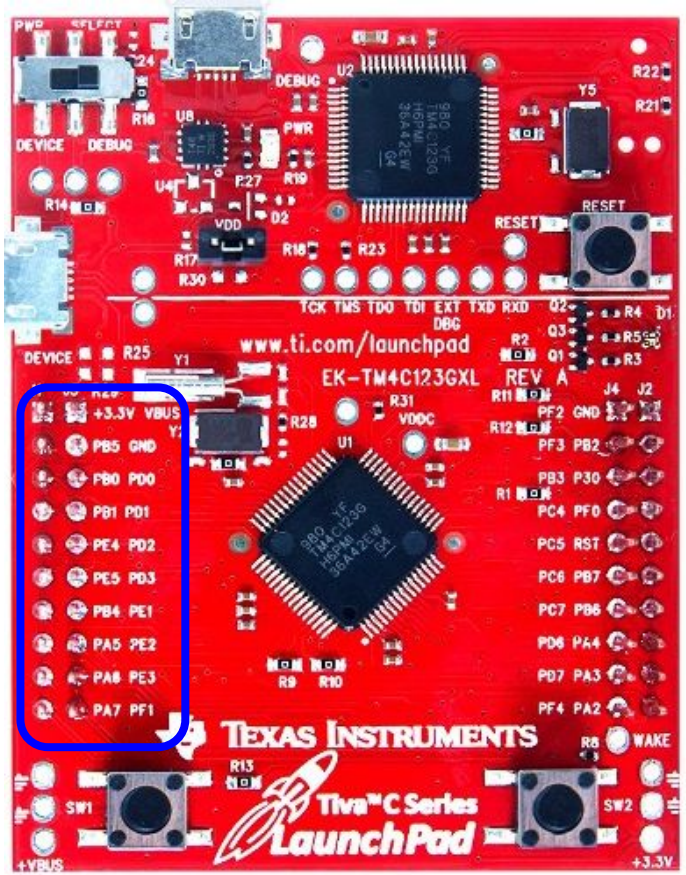


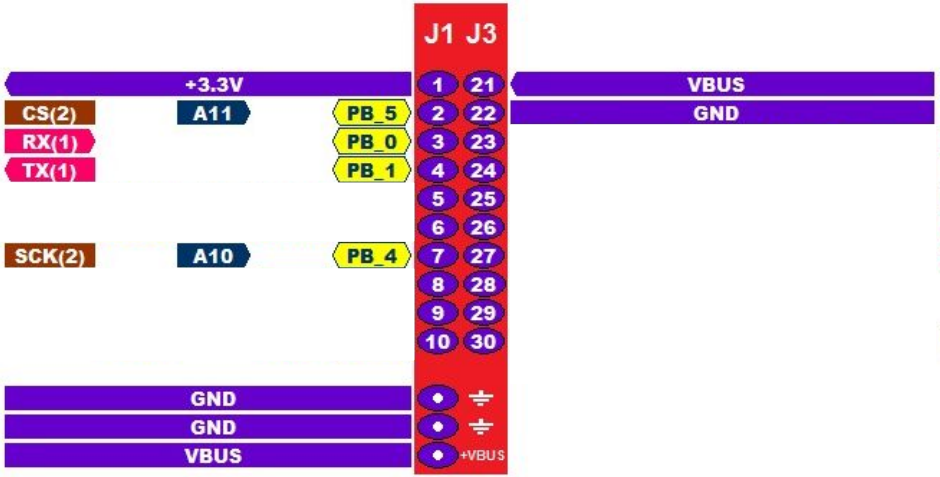
Tiva & Stellaris Port Baglantıları



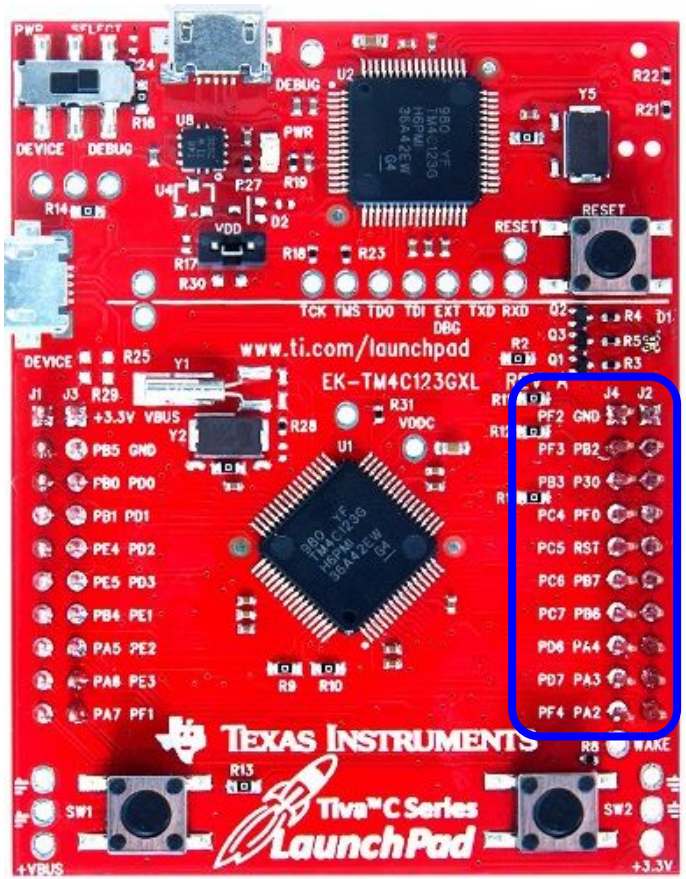


Tiva & Stellaris Port Baglantıları

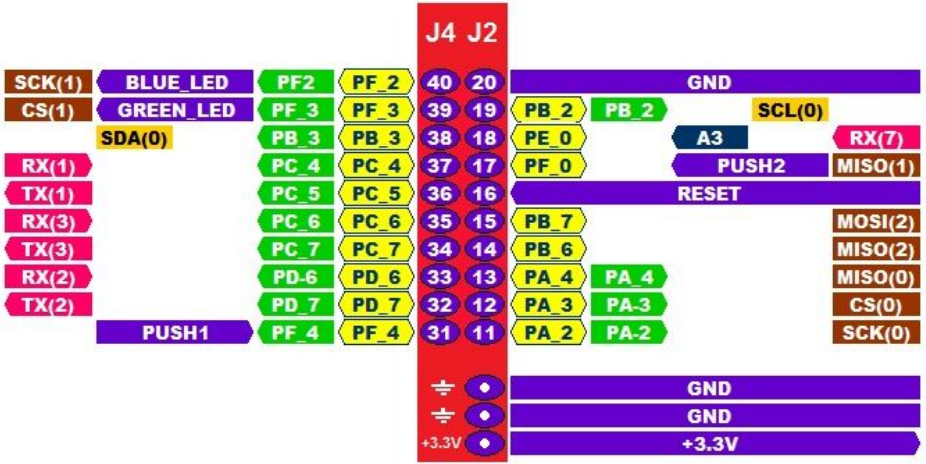




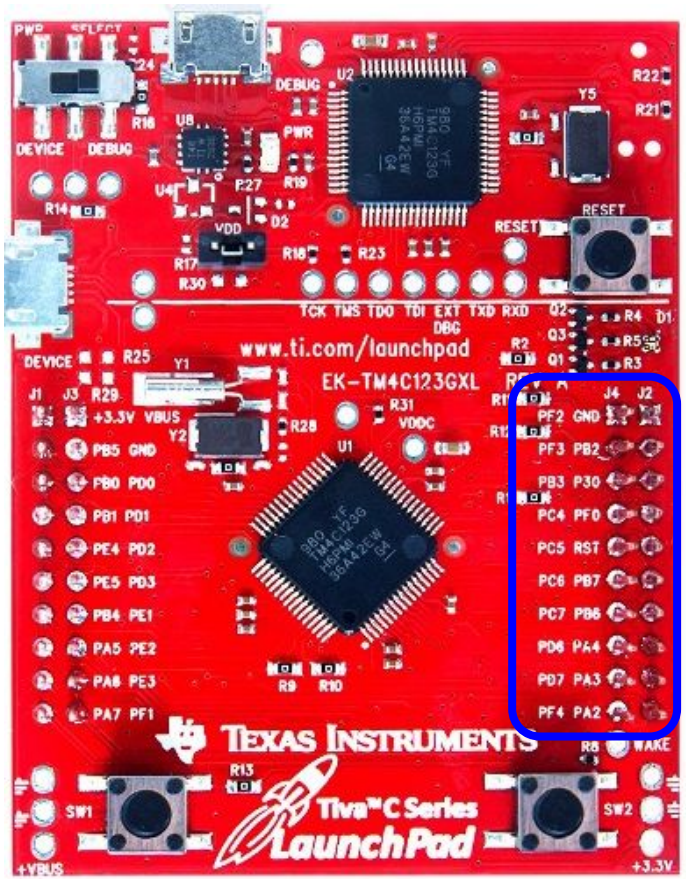
Tiva & Stellaris Port Baglantıları

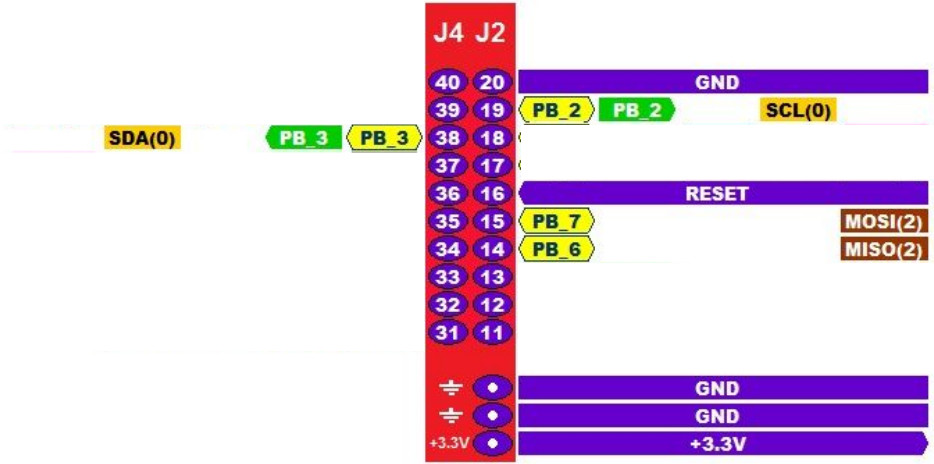


Hardware
digitalRead() and digitalWrite() PORTS
analogRead()
analogWrite()
I²C (TWI)
SPI
Hardware Serial

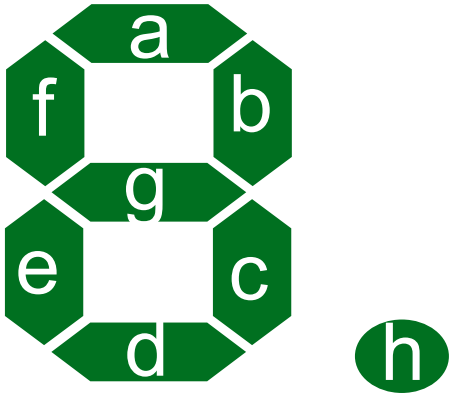


Tiva & Stellaris Port Baglantıları

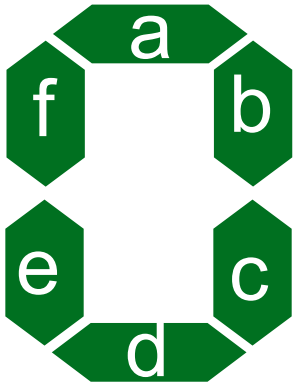




7 Parçalı Gösterge



7 Parçalı Gösterge



SIFIR

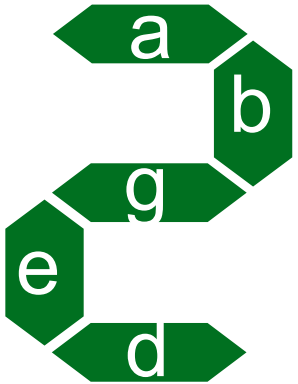
h	g	f	e	d	c	b	a
0	0	1	1	1	1	1	1

7 Parçalı Gösterge

h	g	f	e	d	c	b	a
0	0	0	0	0	1	1	0

BİR

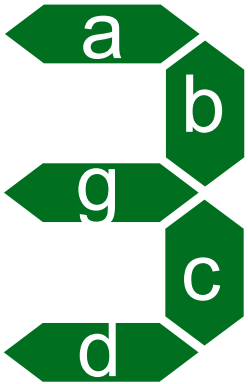
7 Parçalı Gösterge



iKi

h	g	f	e	d	c	b	a
0	1	0	1	1	0	1	1

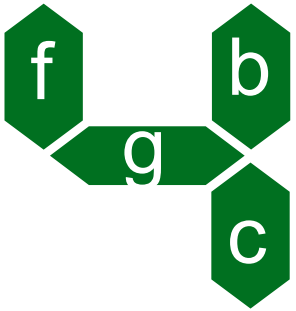
7 Parçalı Gösterge



üç

h	g	f	e	d	c	b	a
0	1	0	0	1	1	1	1

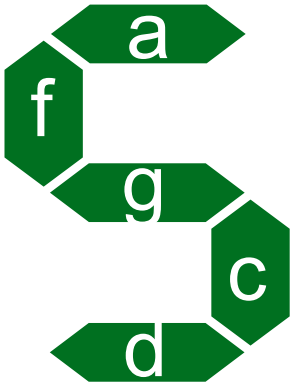
7 Parçalı Gösterge



h	g	f	e	d	c	b	a
0	1	1	0	0	1	1	0

DÖRT

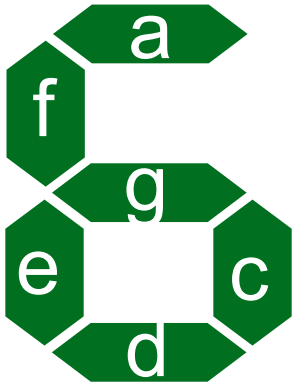
7 Parçalı Gösterge



BEŞ

h	g	f	e	d	c	b	a
0	1	1	0	1	1	0	1

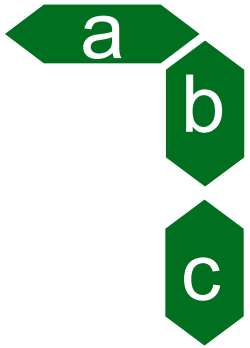
7 Parçalı Gösterge



ALTI

h	g	f	e	d	c	b	a
0	1	1	1	1	1	0	1

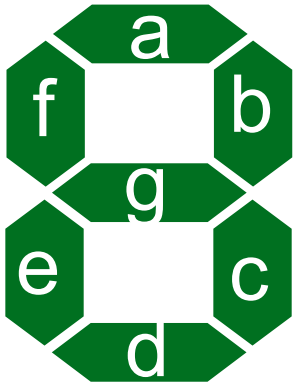
7 Parçalı Gösterge



h	g	f	e	d	c	b	a
0	0	0	0	0	1	1	1

YEDİ

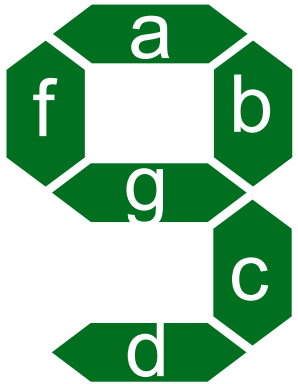
7 Parçalı Gösterge



SEKİZ

h	g	f	e	d	c	b	a
0	1	1	1	1	1	1	1

7 Parçalı Gösterge



h	g	f	e	d	c	b	a
0	1	1	0	1	1	1	1

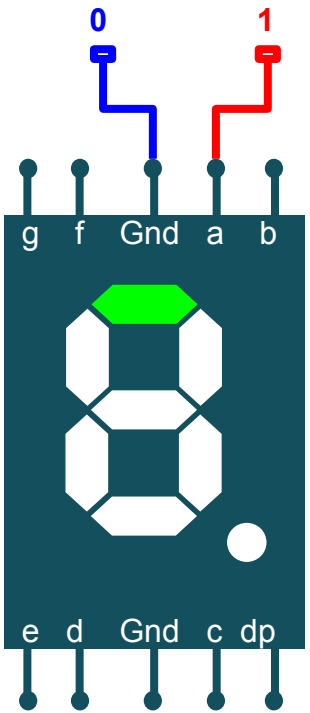
DOKUZ

7 Parçalı Gösterge

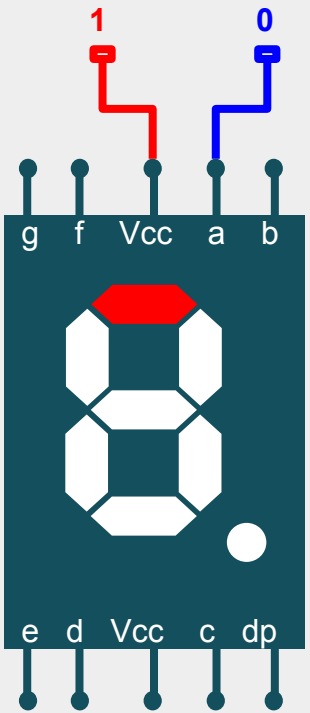
0	0b0111111
1	0b0000110
2	0b1011011
3	0b1001111
4	0b1100110
5	0b1101101
6	0b1111101
7	0b0000111
8	0b1111111
9	0b1101111

7 Parçalı Gösterge

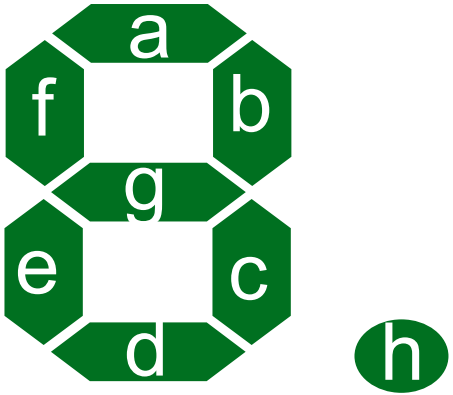
Ortak Katot



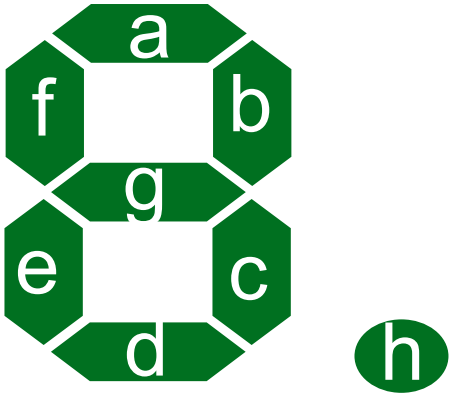
Ortak Anod

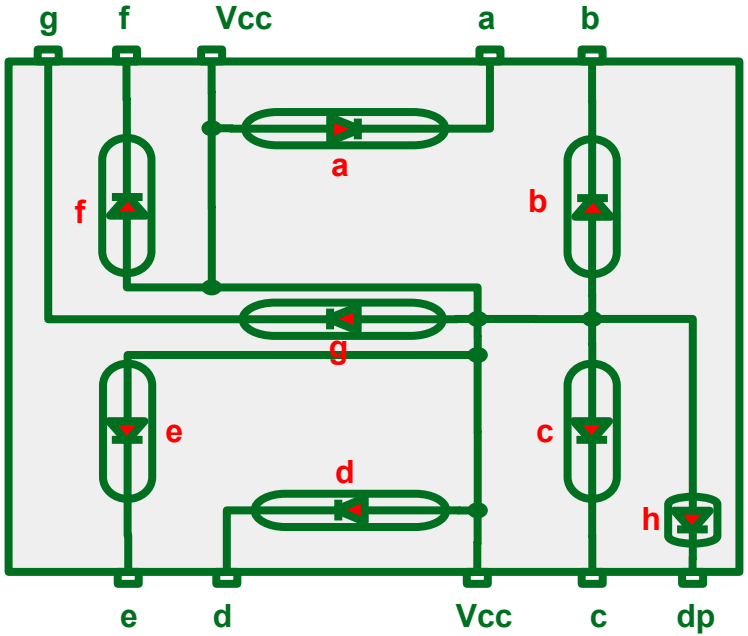


7 Parçalı Gösterge

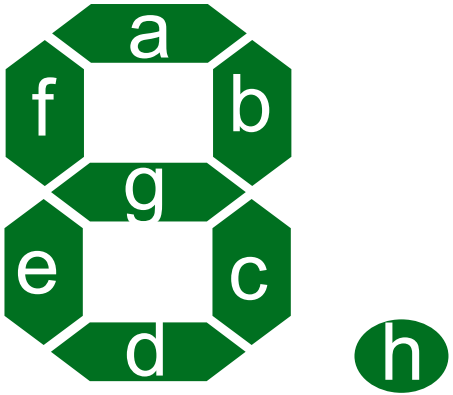


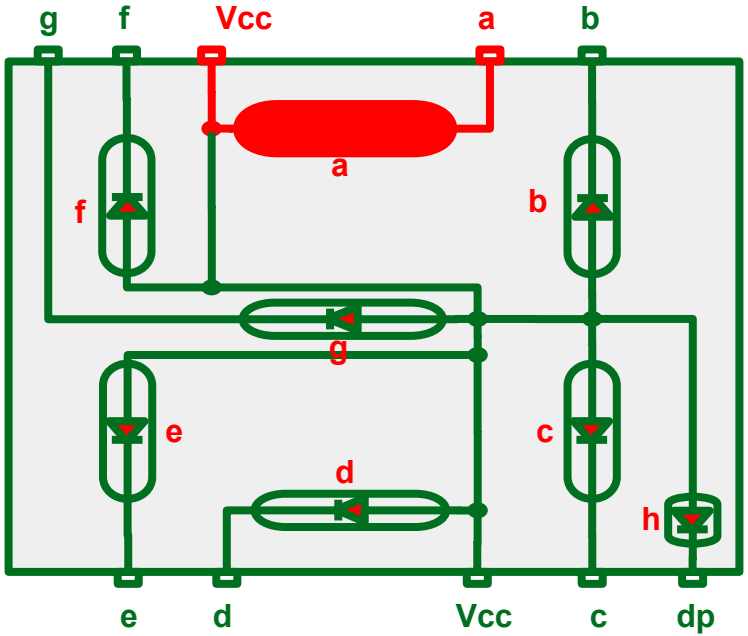
7 Parçalı Gösterge



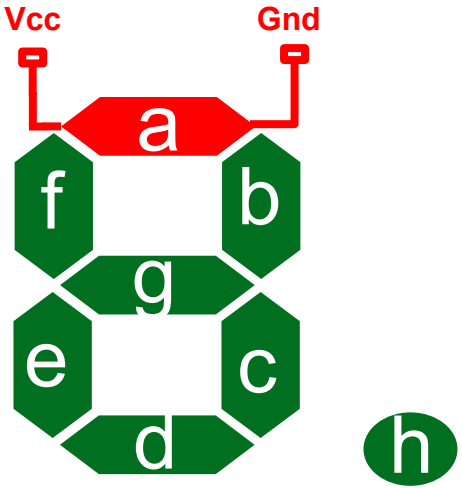


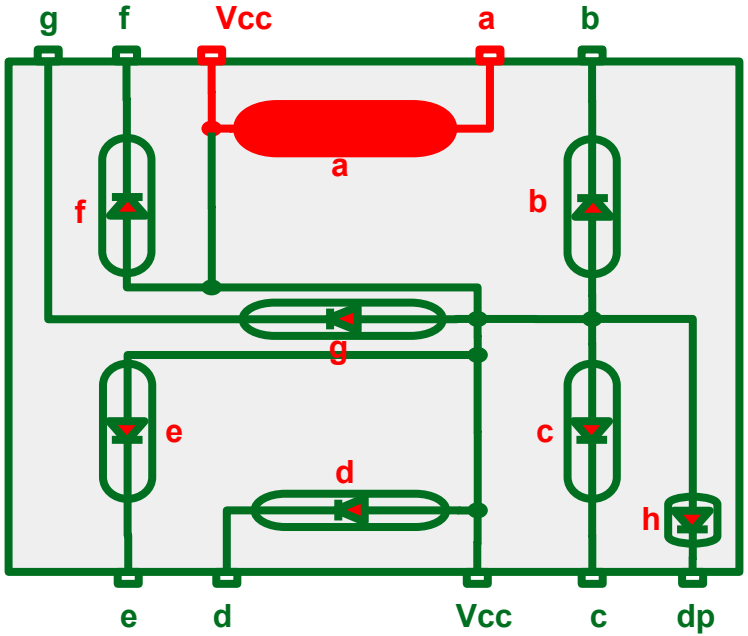
7 Parçalı Gösterge



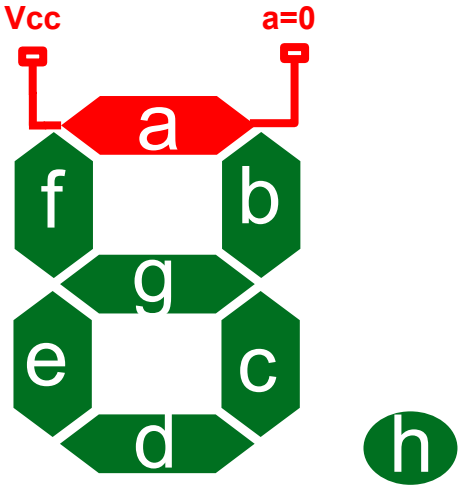


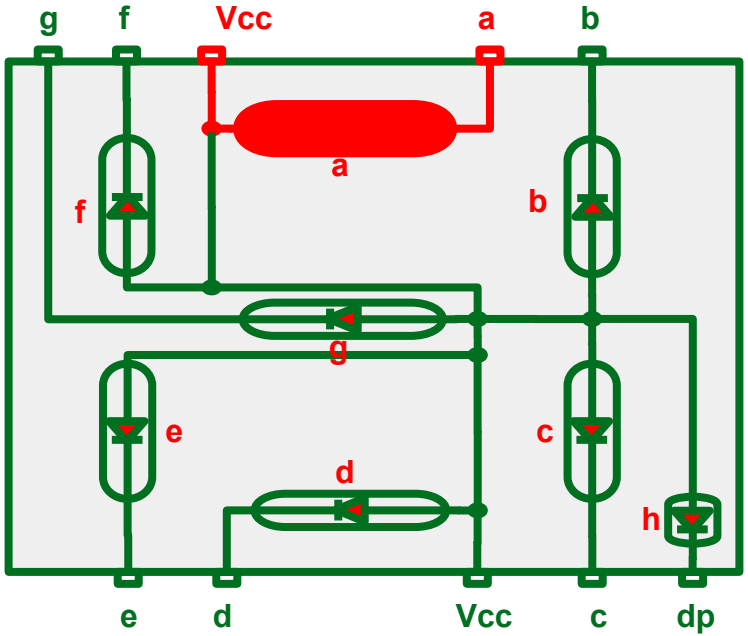
7 Parçalı Gösterge



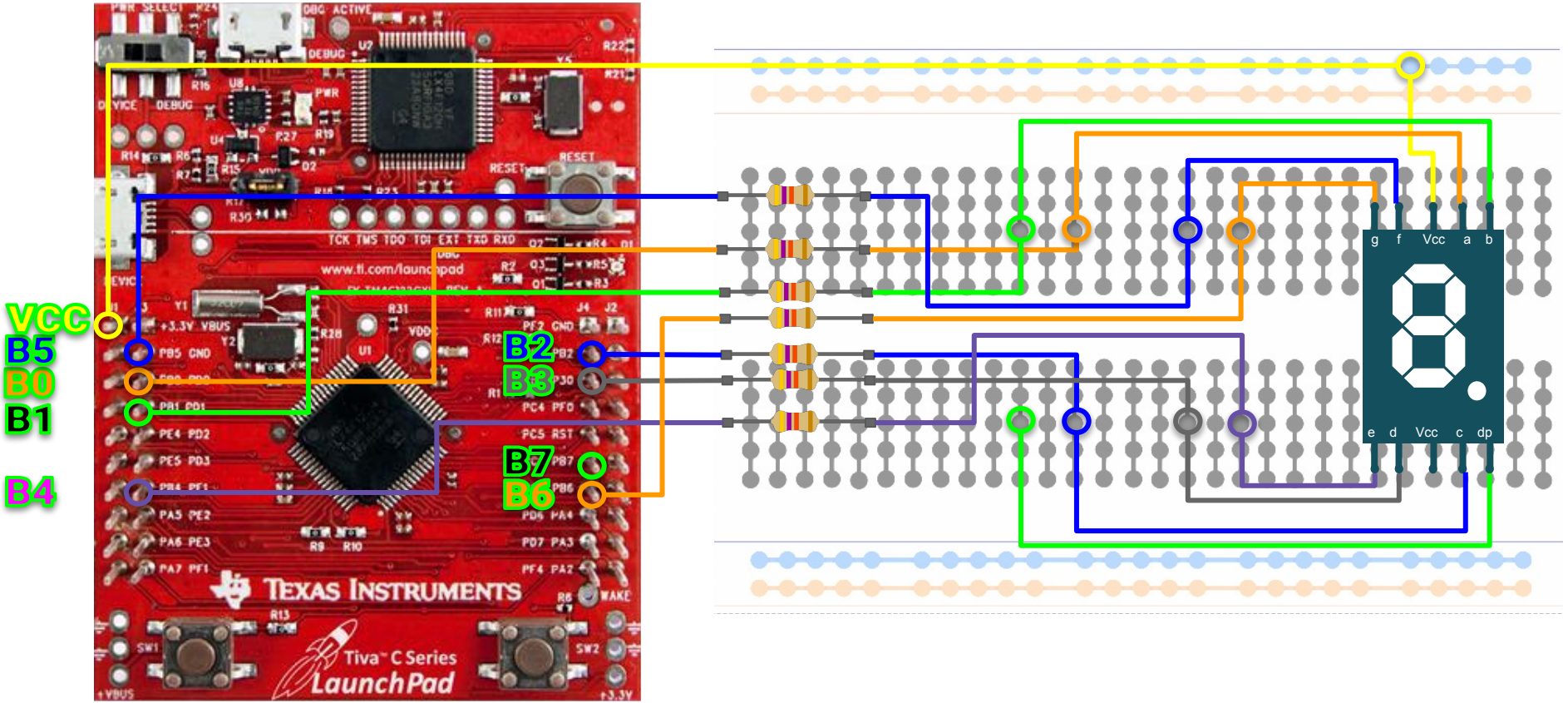


7 Parçalı Gösterge

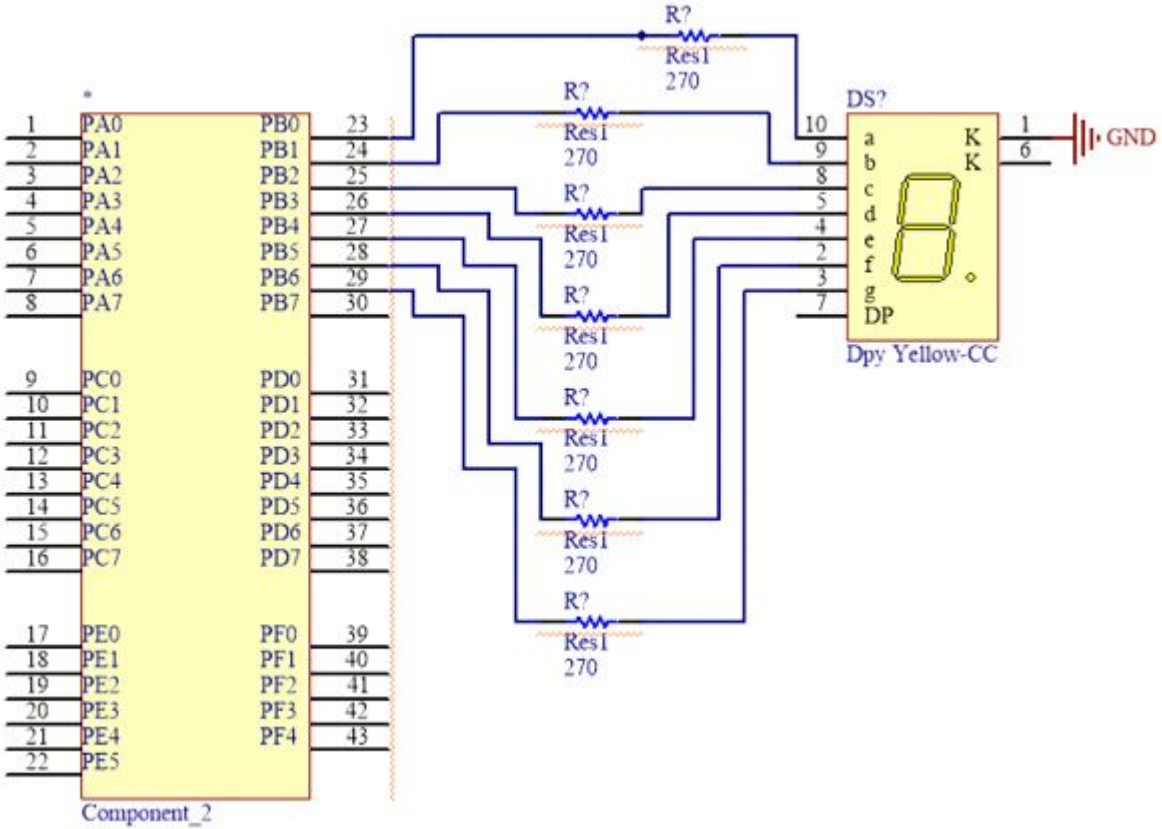




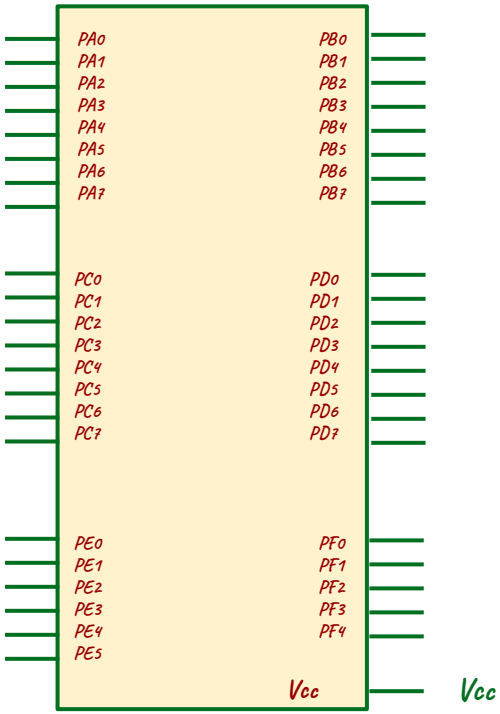
Elektronik Devre Kurulumu

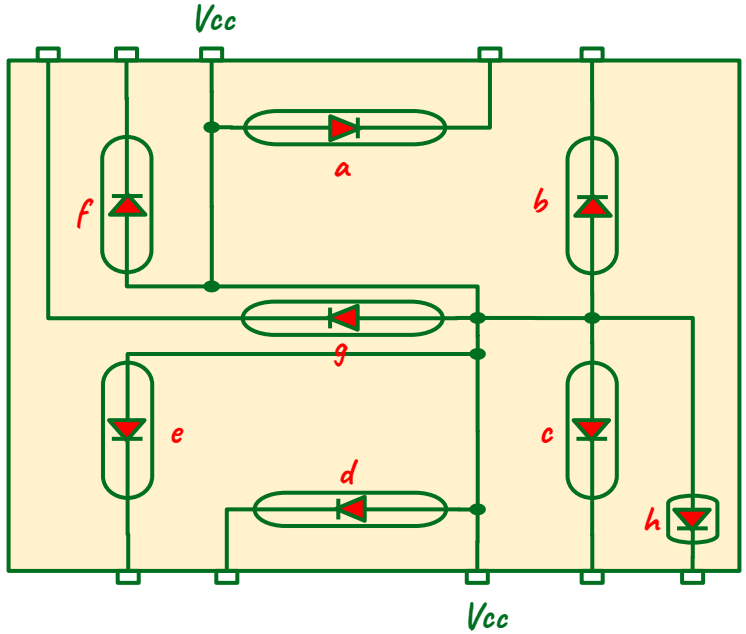


Elektronik Devre Kurulumu

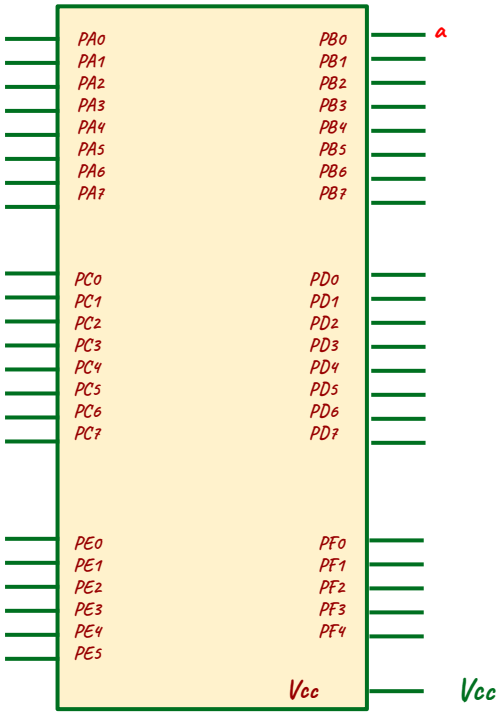


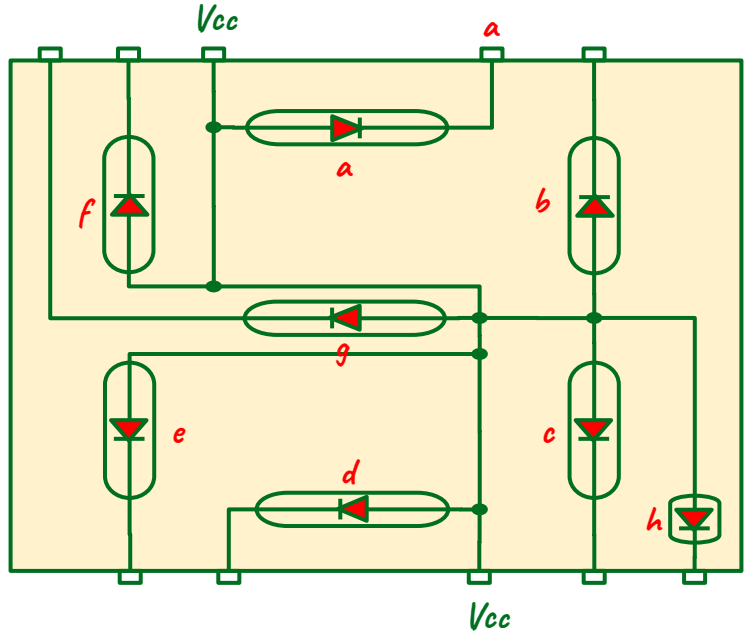
Elektronik Devre Kurulumu



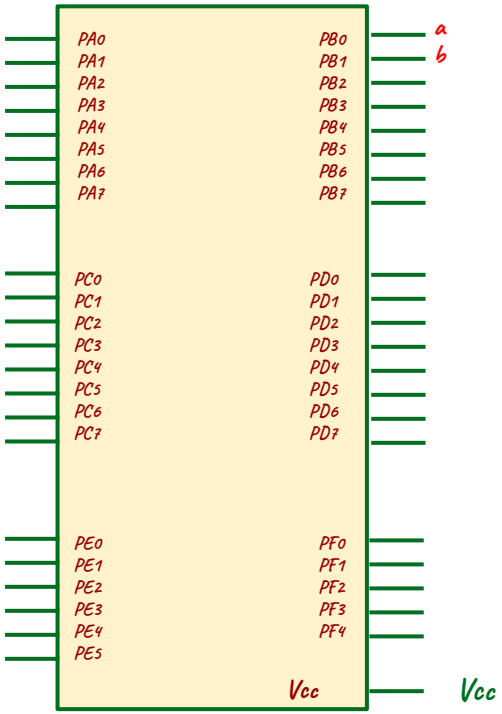


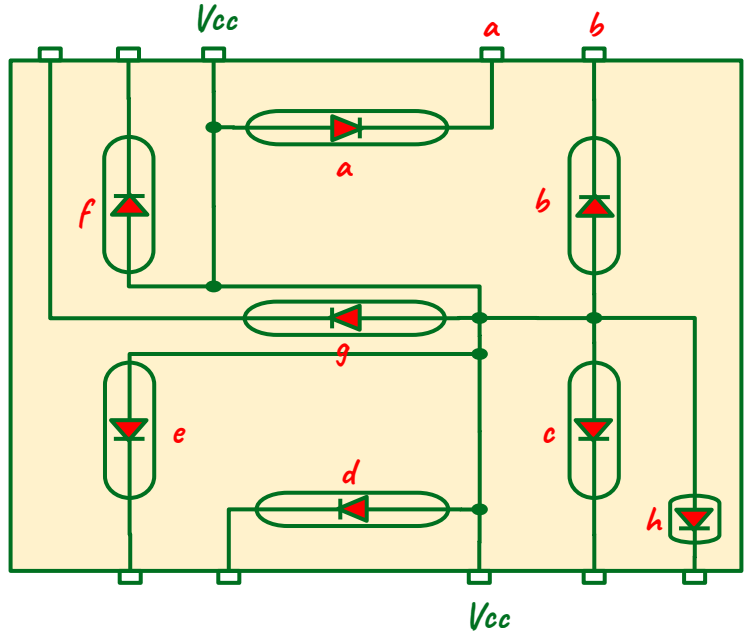
Elektronik Devre Kurulumu



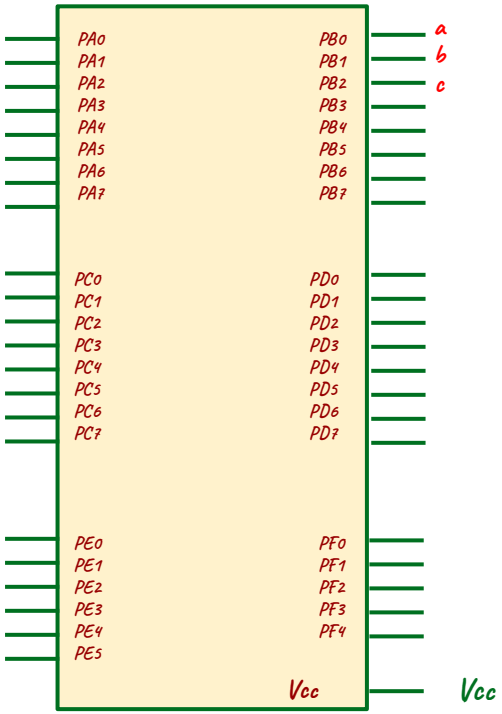


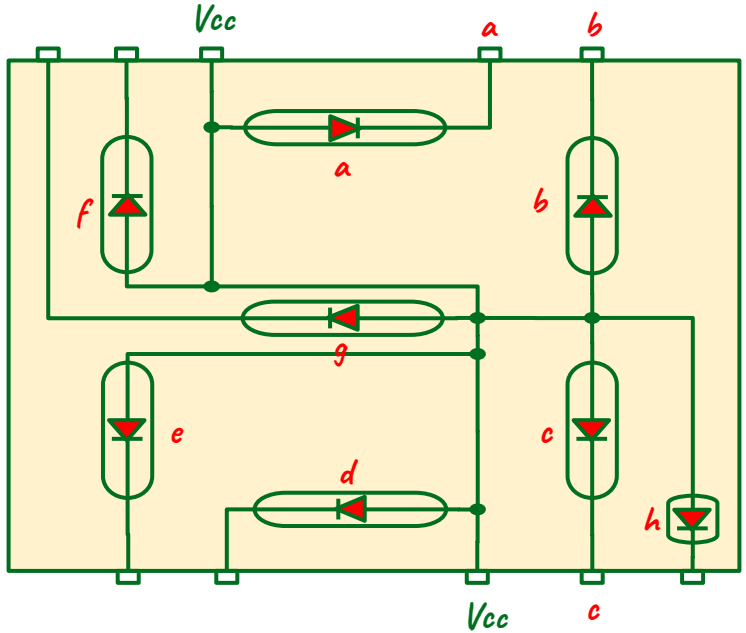
Elektronik Devre Kurulumu



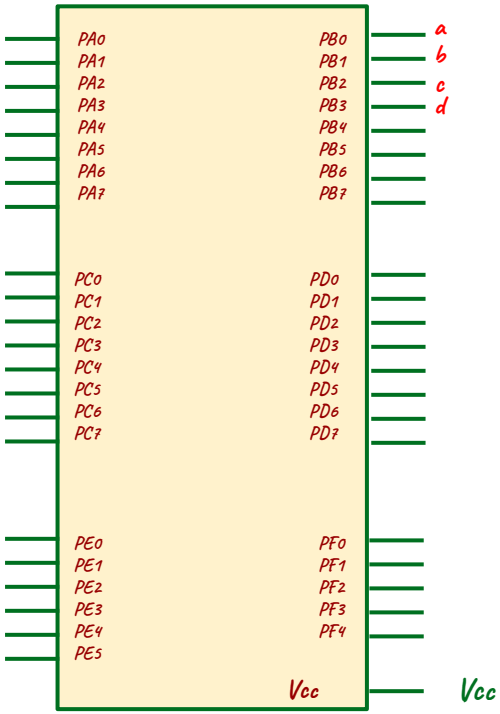


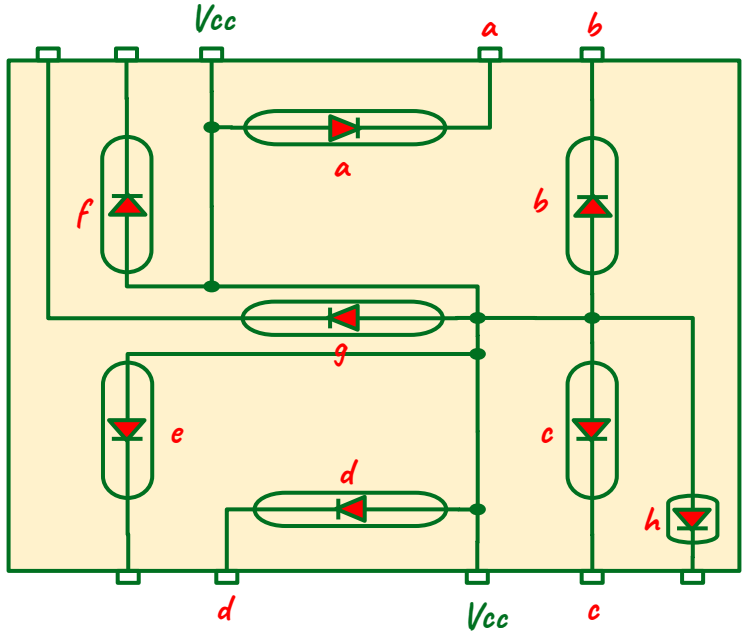
Elektronik Devre Kurulumu



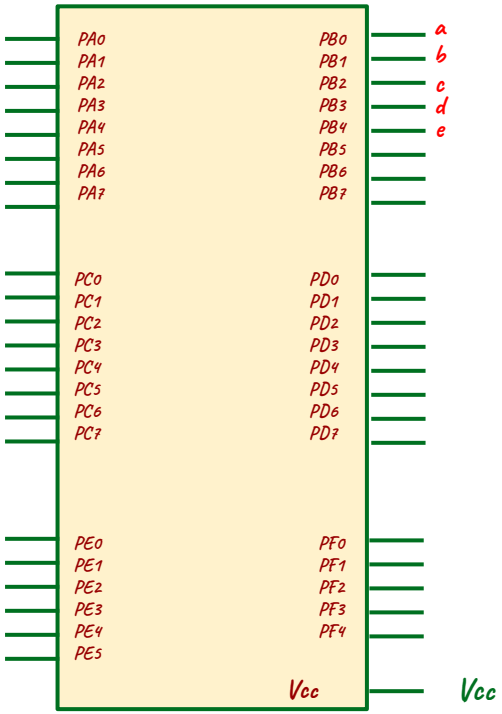


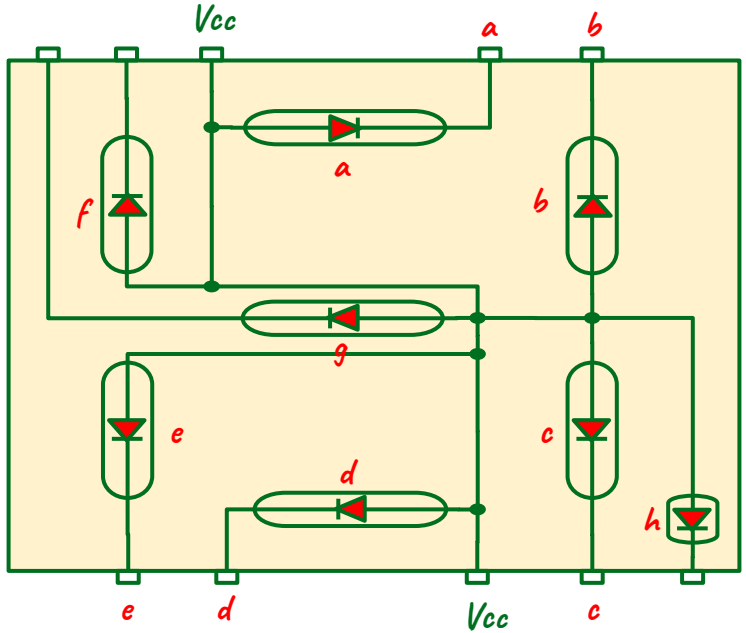
Elektronik Devre Kurulumu



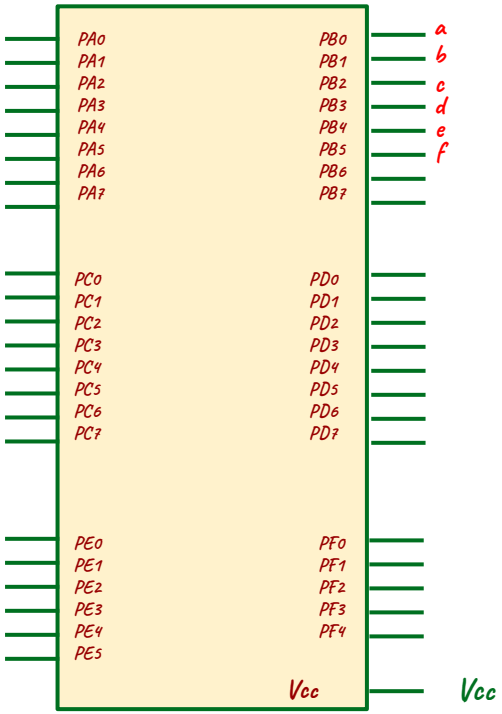


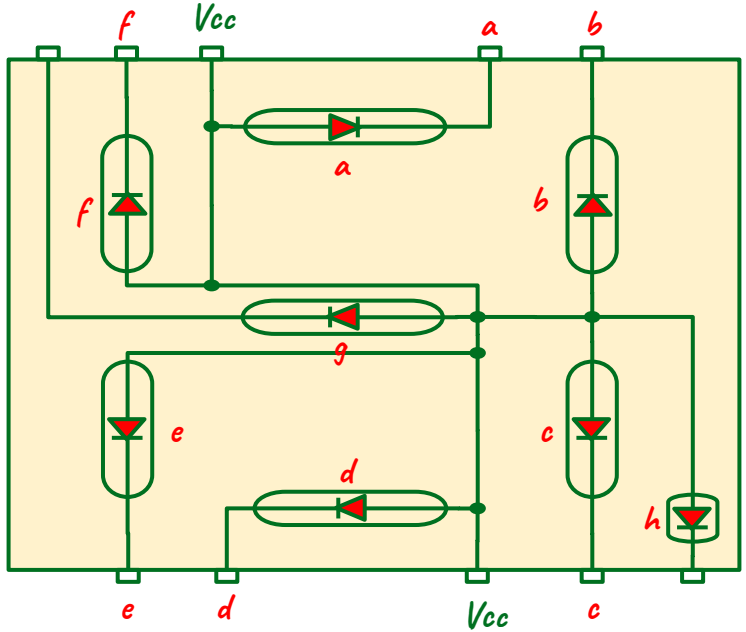
Elektronik Devre Kurulumu



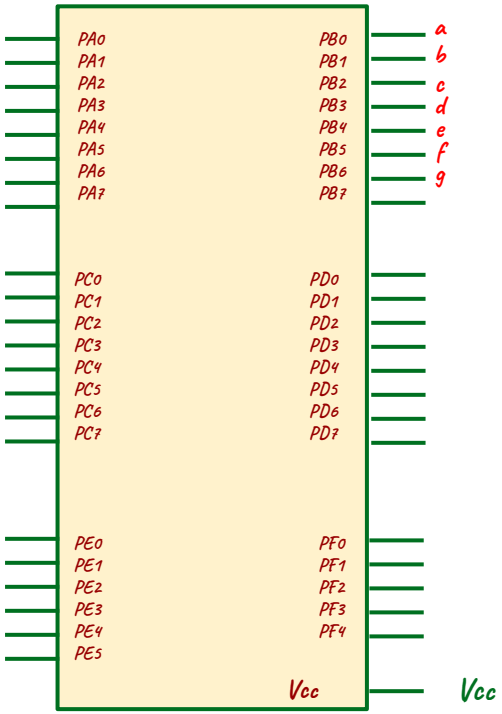


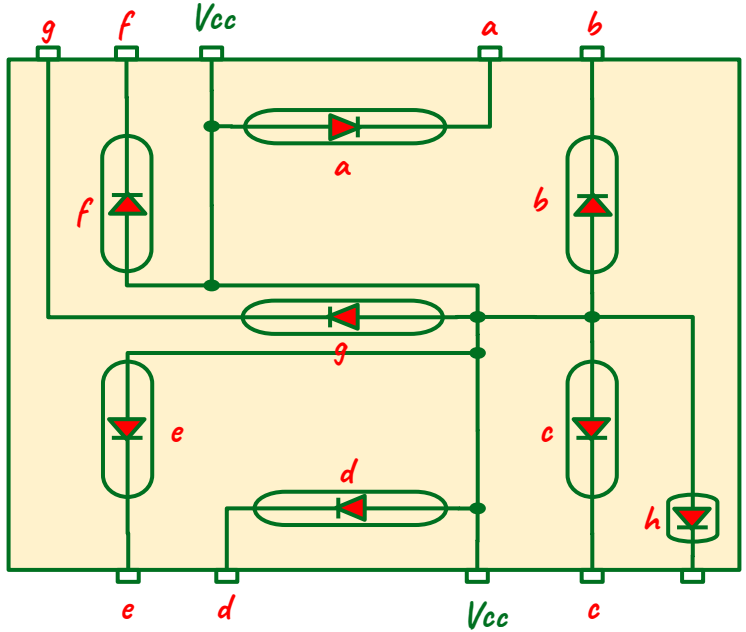
Elektronik Devre Kurulumu



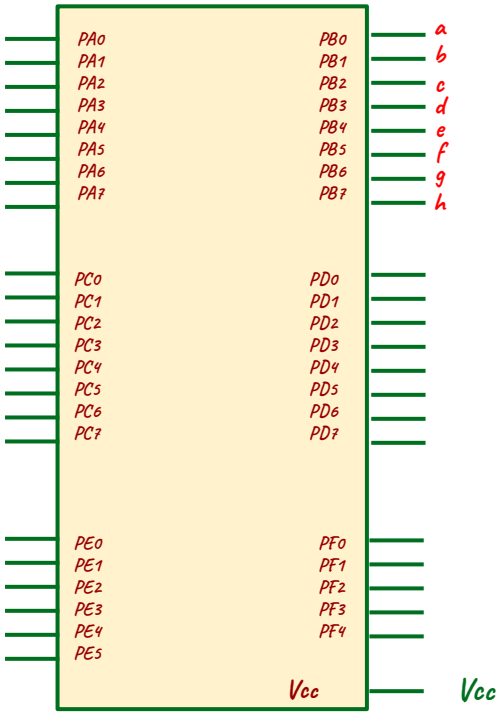


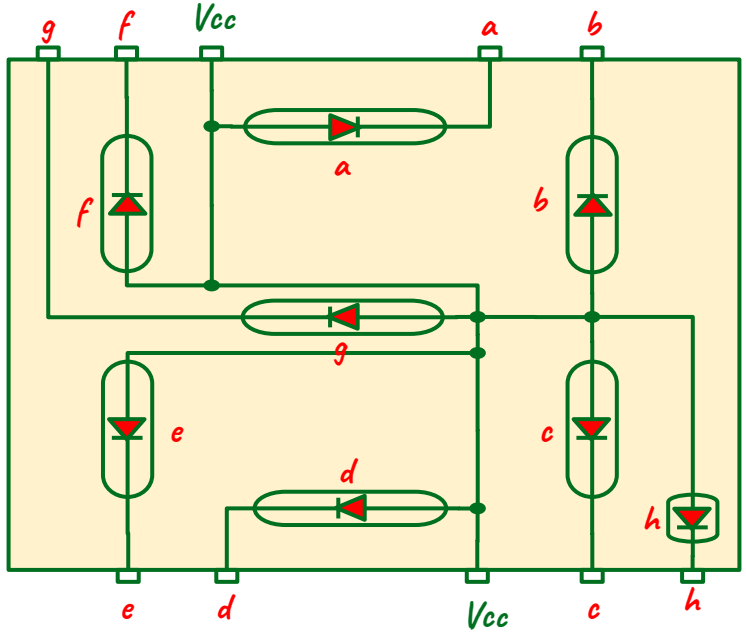
Elektronik Devre Kurulumu





Elektronik Devre Kurulumu





Elektronik Devre Kurulumu

```
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
void init_port_B() {
        volatile unsigned long delay;
        SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOB;          // Port B'nin saati aktifleştir
        delay = SYSCTL_RCGC2_R;                        // Gecikme



}
```

Elektronik Devre Kurulumu

```
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
void init_port_B() {
        volatile unsigned long delay;
        SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOB;          // Port B'nin saati aktifleştir
        delay = SYSCTL_RCGC2_R;                        // Gecikme
        GPIO_PORTB_DIR_R |= 0xFF;                      // Port B'nin tüm bitleri cikis olarak ayarla


}
```

Elektronik Devre Kurulumu

```
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
void init_port_B() {
        volatile unsigned long delay;
        SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOB;          // Port B'nin saati aktifleştir
        delay = SYSCTL_RCGC2_R;                        // Gecikme
        GPIO_PORTB_DIR_R |= 0xFF;                      // Port B'nin tüm bitleri cikis olarak ayarla
        GPIO_PORTB_AFSEL_R &= ~0xFF;                   // Alternatif fonksiyonlar kapat

}
```

Elektronik Devre Kurulumu

```
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
void init_port_B() {
        volatile unsigned long delay;
        SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOB;          // Port B'nin saati aktifleştir
        delay = SYSCTL_RCGC2_R;                        // Gecikme
        GPIO_PORTB_DIR_R |= 0xFF;                      // Port B'nin tüm bitleri cikis olarak ayarla
        GPIO_PORTB_AFSEL_R &= ~0xFF;                   // Alternatif fonksiyonlar kapat
        GPIO_PORTB_DEN_R |= 0xFF;                      // Tüm Port B için Digital çalışmayı aktifleştir
}
```

Seven Segment Tablosu

0	0b0111111
1	0b0000110
2	0b1011011
3	0b1001111
4	0b1100110
5	0b1101101
6	0b1111101
7	0b0000111
8	0b1111111
9	0b1101111

Seven Segment Tablosu

```
uint8_t kodlar[10] = {
    ~0b0111111,
    ~0b0000110,
    ~0b1011011,
    ~0b1001111,
    ~0b1100110,
    ~0b1101101,
    ~0b1111101,
    ~0b0000111,
    ~0b1111111,
    ~0b1101111
};
```


```
0b0111111
0b0000110
0b1011011
0b1001111
0b1100110
0b1101101
0b1111101
0b0000111
0b1111111
0b1101111
```

Seven Segment Tablosu

```
int main() {
        volatile unsigned long delay;









}
```

Seven Segment Tablosu

```
int main() {
        volatile unsigned long delay;
        uint8_t kodlar[10] = {
           ~0b0111111,
           ~0b0000110,
           ~0b1011011,
           ~0b1001111,
           ~0b1100110,
           ~0b1101101,
           ~0b1111101,
           ~0b0000111,
           ~0b1111111,
           ~0b1101111
        };









}
```

Seven Segment Tablosu

```
int main() {
        volatile unsigned long delay;
        uint8_t kodlar[10] = {
           ~0b0111111,
           ~0b0000110,
           ~0b1011011,
           ~0b1001111,
           ~0b1100110,
           ~0b1101101,
           ~0b1111101,
           ~0b0000111,
           ~0b1111111,
           ~0b1101111
        };
        init_port_B();
        int sayi = 0;




}
```

Seven Segment Tablosu

```
int main() {
        volatile unsigned long delay;
        uint8_t kodlar[10] = {
           ~0b0111111,
           ~0b0000110,
           ~0b1011011,
           ~0b1001111,
           ~0b1100110,
           ~0b1101101,
           ~0b1111101,
           ~0b0000111,
           ~0b1111111,
           ~0b1101111
        };
        init_port_B();
        int sayi = 0;
        while (1) {



        }
}
```

Seven Segment Tablosu

```
int main() {
        volatile unsigned long delay;
        uint8_t kodlar[10] = {
           ~0b0111111,
           ~0b0000110,
           ~0b1011011,
           ~0b1001111,
           ~0b1100110,
           ~0b1101101,
           ~0b1111101,
           ~0b0000111,
           ~0b1111111,
           ~0b1101111
        };
        init_port_B();
        int sayi = 0;
        while (1) {
           GPIO_PORTB_DATA_R = kodlar[sayi];


        }
}
```

Seven Segment Tablosu

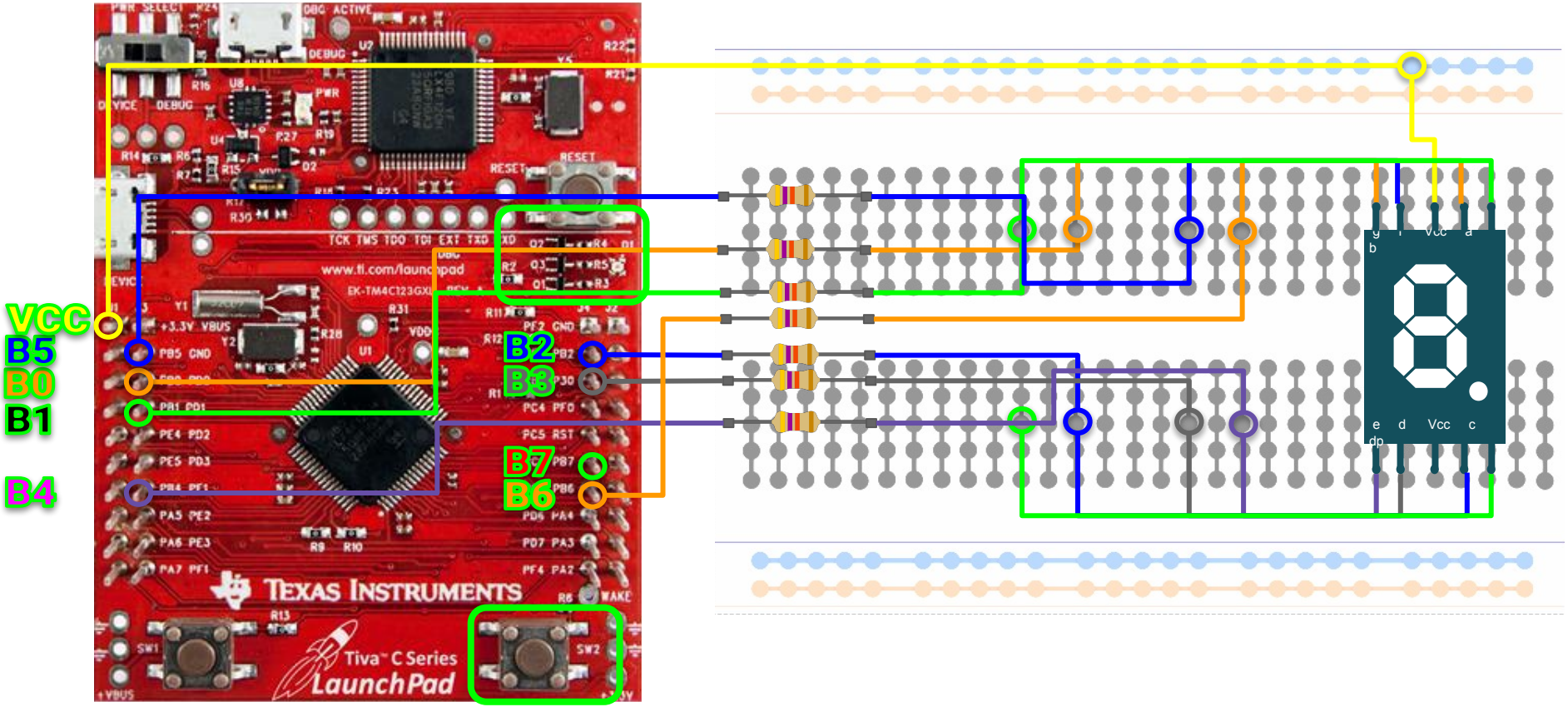
```
int main() {
        volatile unsigned long delay;
        uint8_t kodlar[10] = {
           ~0b0111111,
           ~0b0000110,
           ~0b1011011,
           ~0b1001111,
           ~0b1100110,
           ~0b1101101,
           ~0b1111101,
           ~0b0000111,
           ~0b1111111,
           ~0b1101111
        };
        init_port_B();
        int sayi = 0;
        while (1) {
           GPIO_PORTB_DATA_R = kodlar[sayi];
           for (delay = 0 ; delay < 1000000 ; delay++);

        }
}
```

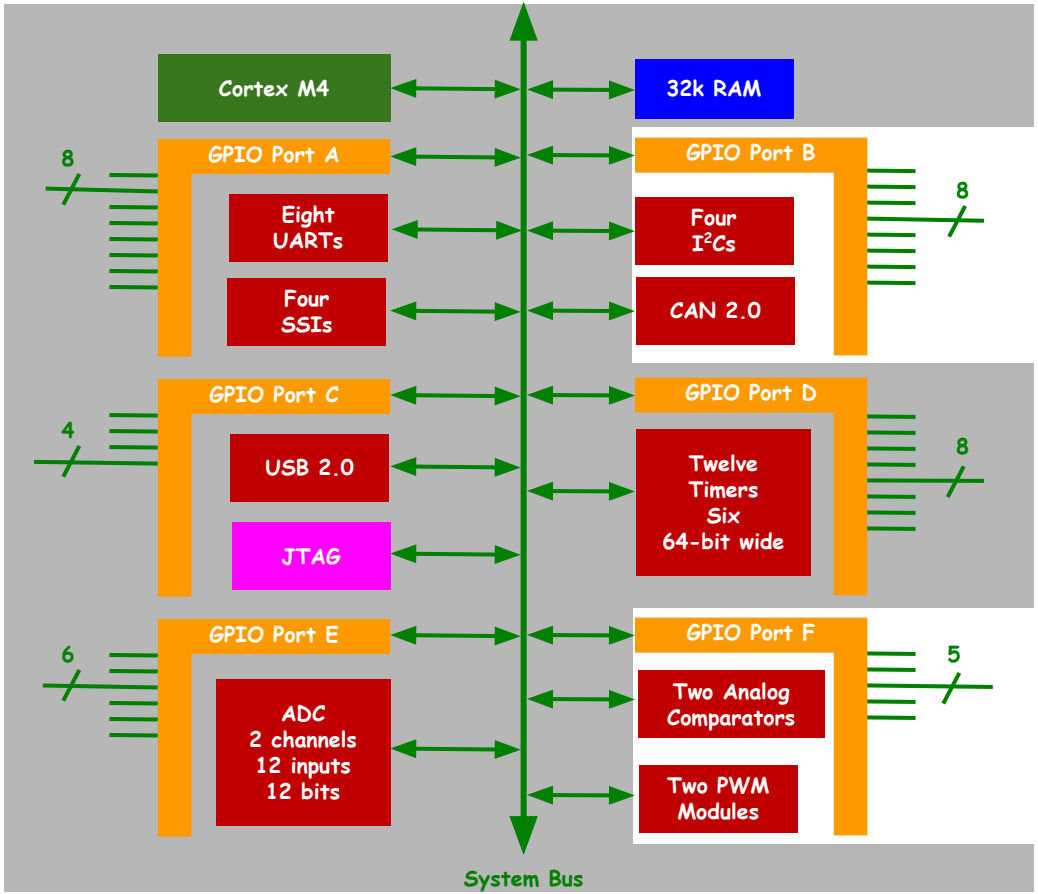
Seven Segment Tablosu

```
int main() {
        volatile unsigned long delay;
        uint8_t kodlar[10] = {
           ~0b0111111,
           ~0b0000110,
           ~0b1011011,
           ~0b1001111,
           ~0b1100110,
           ~0b1101101,
           ~0b1111101,
           ~0b0000111,
           ~0b1111111,
           ~0b1101111
        };
        init_port_B();
        int sayi = 0;
        while (1) {
           GPIO_PORTB_DATA_R = kodlar[sayi];
           for (delay = 0 ; delay < 1000000 ; delay++);
           sayi = (sayi + 1) % 10;
        }
}
```

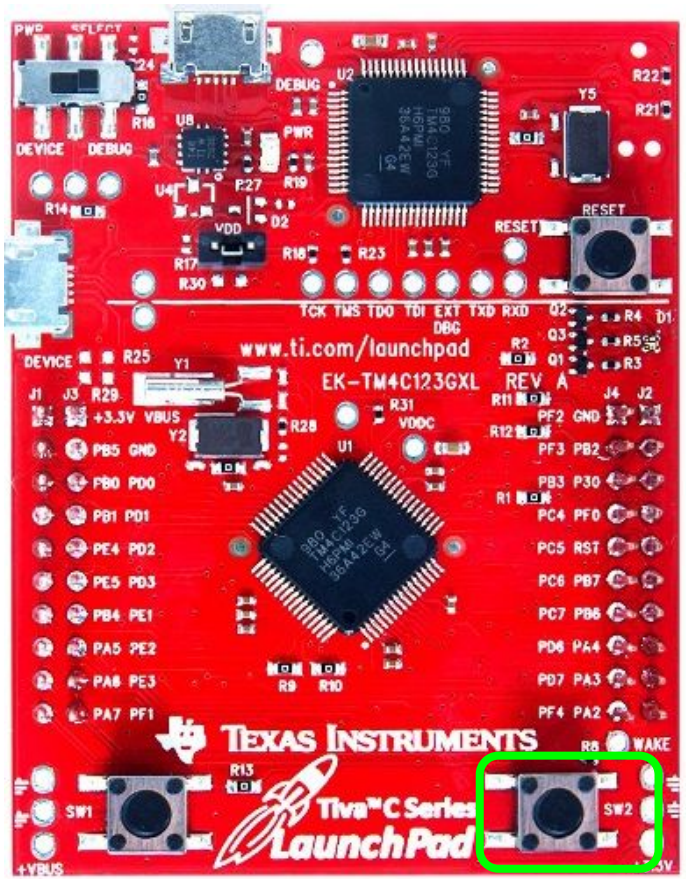
Seven Segment Buton Led



Seven Segment Buton Led

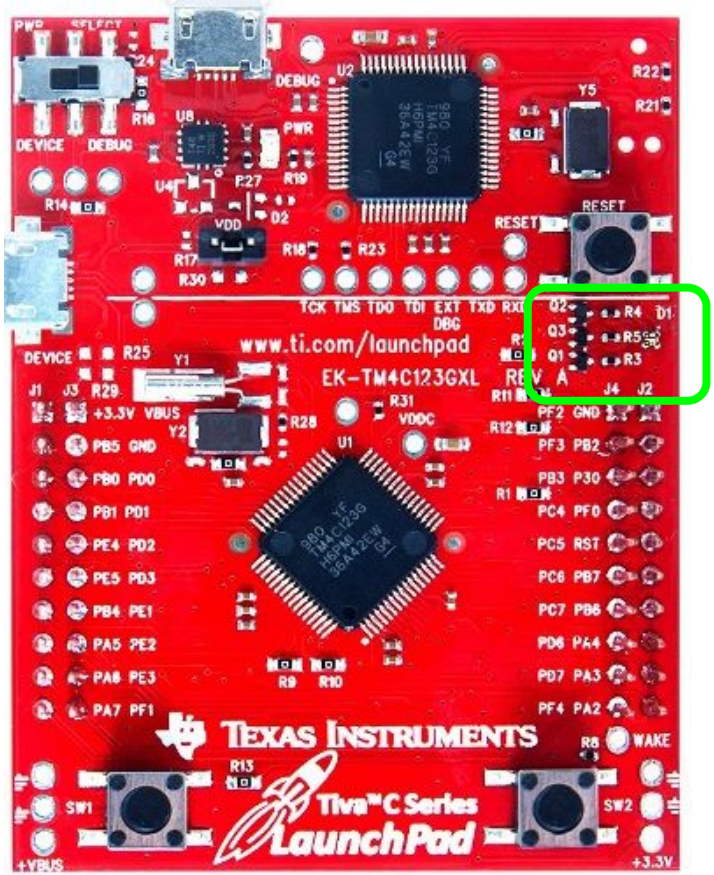


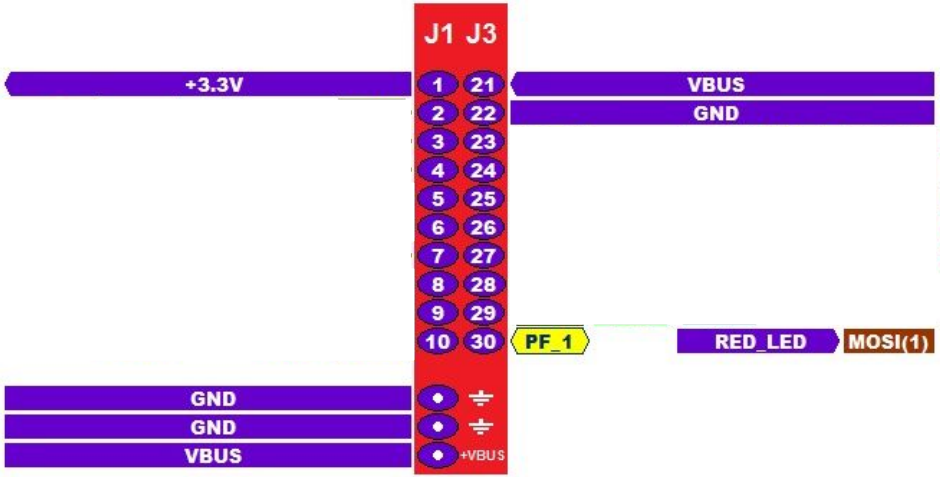
Tiva & Stellaris Port Baglantıları



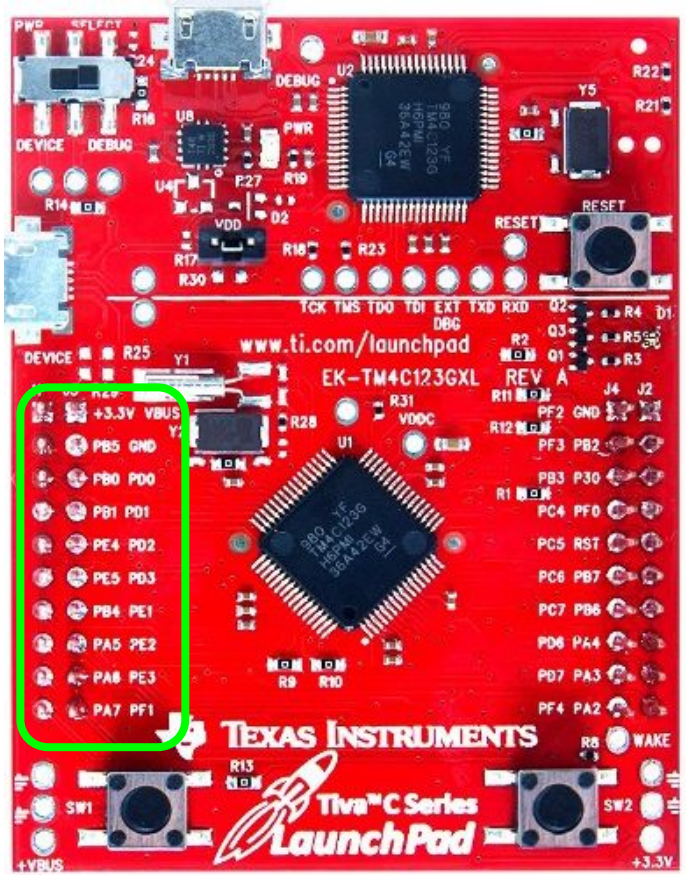
Hardware
digitalRead() and digitalWrite() PORTS
analogRead()
analogWrite()
I²C (TWI)
SPI
Hardware Serial

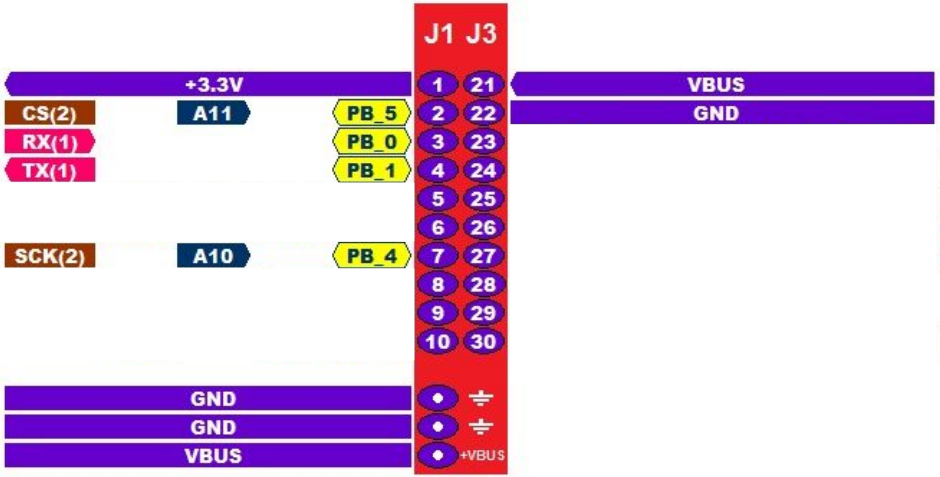
Tiva & Stellaris Port Baglantıları



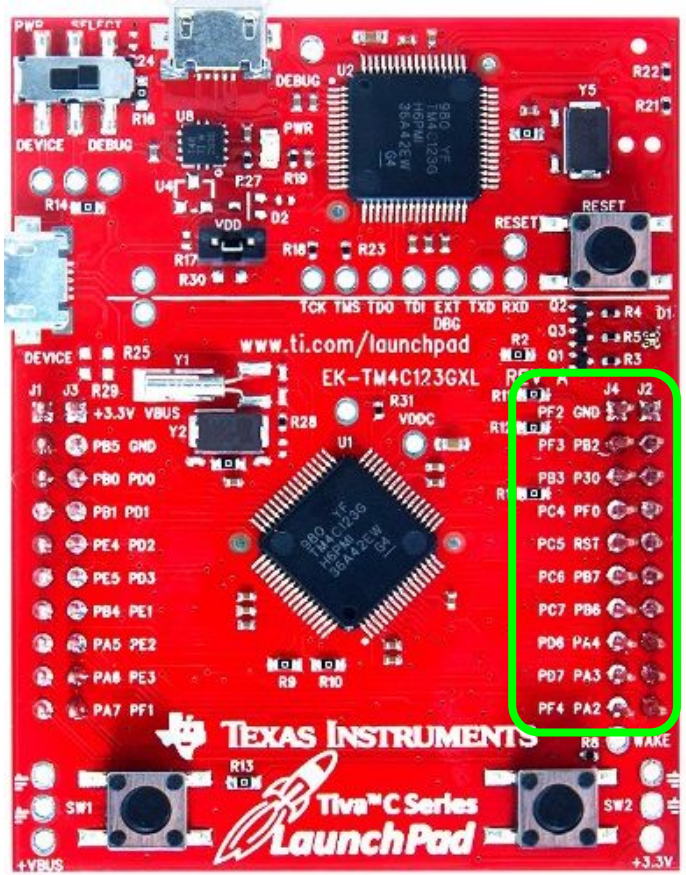


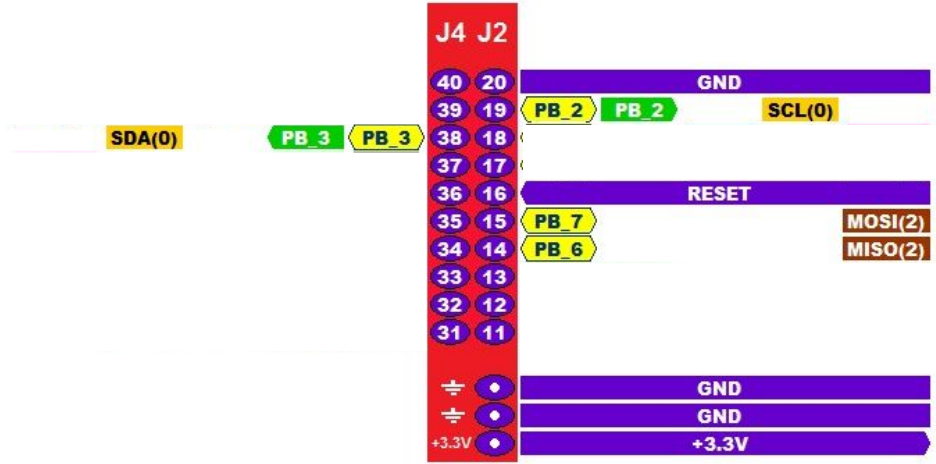
Tiva & Stellaris Port Baglantıları



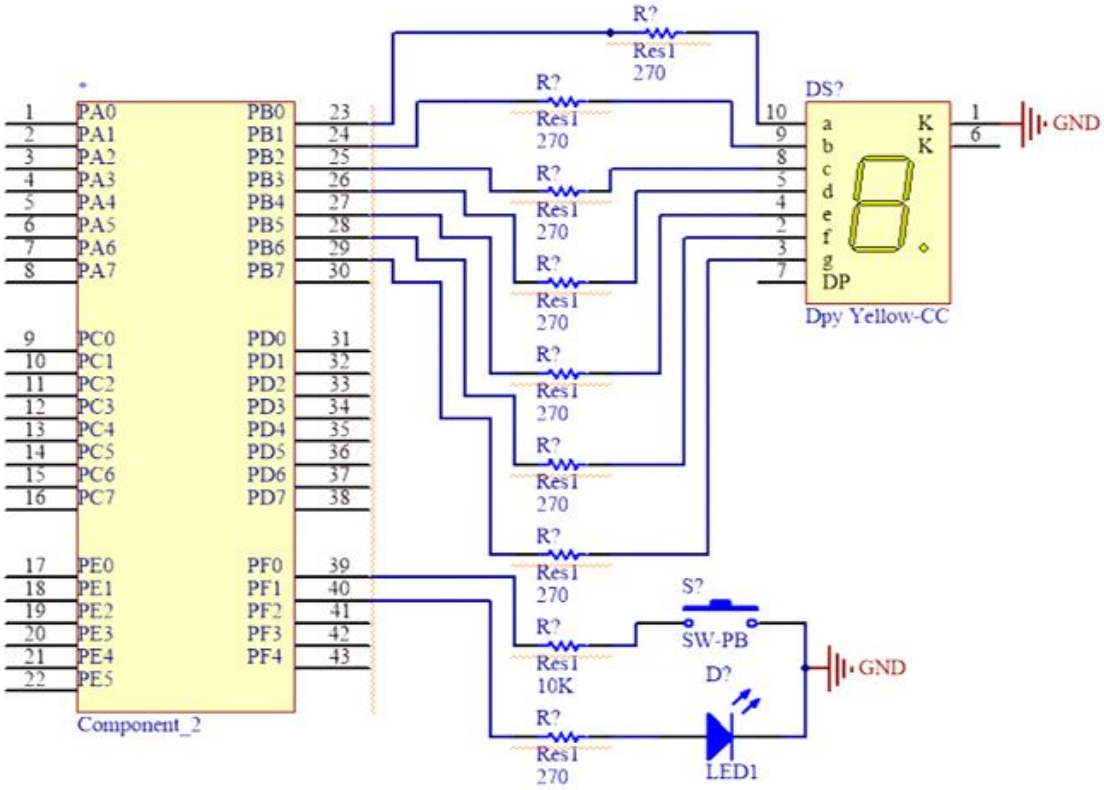


Tiva & Stellaris Port Baglantıları





Tiva & Stellaris Port Baglantıları



Es Zamanlı islem yapma

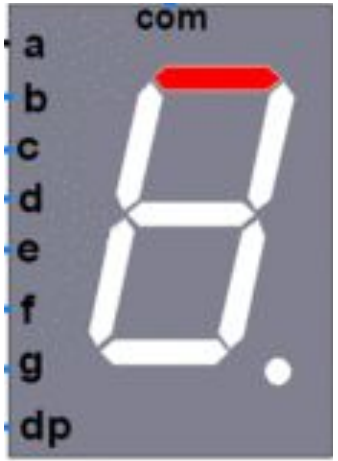
Es Zamanlı islem yapma

islem_2

islem_1

Es Zamanlı islem yapma

islem_1



islem_2

Mesgul Bekleme

```
void init_port_F() {
        volatile unsigned long tmp;
        SYSCTL_RCGCGPIO_R |= 0x00000020;          // Port F'nin saatini aktifleştir
        tmp = SYSCTL_RCGCGPIO_R;                  // Saatin başlaması için gecikme
        GPIO_PORTF_LOCK_R = 0x4C4F434B;           // Port F GPIO kilidini aç
        GPIO_PORTF_CR_R = 0x1F;                   // PF4-0 kilidini aç
        GPIO_PORTF_AMSEL_R = 0x00;              // PF anlog I/O kapat
        GPIO_PORTF_PCTL_R = 0x00000000;         // PF4-0 GPIO olarak ayarla
        GPIO_PORTF_DIR_R = 0x0E;                  // PF4,PF0 giriş, PF3-1 çıkış
        GPIO_PORTF_AFSEL_R = 0x00;                // PF7-0 Alternatif fonksiyonları kapat
        GPIO_PORTF_PUR_R = 0x11;                  // PF0 ve PF4 üzerindeki pull-up direncini aktifleştir
        GPIO_PORTF_DEN_R = 0x1F;                  // PF4-0 digital I/O aktifleştir
}
```

Mesgul Bekleme

```
void init_port_B() {
        volatile unsigned long delay;
        SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOB;          // Port B'nin saati aktifleştir
        delay = SYSCTL_RCGC2_R;                        // Gecikme
        GPIO_PORTB_DIR_R |= 0xFF;                      // Port B'nin tüm bitleri cikis olarak ayarla
        GPIO_PORTB_AFSEL_R &= ~0xFF;                   // Alternatif fonksiyonlar kapat
        GPIO_PORTB_DEN_R |= 0xFF;                      // Tüm Port B için Digital çalışmayı aktifleştir
}
```

Mesgul Bekleme

```
volatile int sayi = 0;
//Seven segment sayacı
void islem_1() {









}
```

Mesgul Bekleme

```
volatile int sayi = 0;
//Seven segment sayacı
void islem_1() {
	volatile unsigned long delay;
	uint8_t kodlar[] = {
		0b00111111,
		0b00000110,
		0b01011011,
		0b01001111,
		0b01100110,
		0b01101101,
		0b01111101,
		0b00000111,
		0b01111111,
		0b01101111
	};



}
```

Mesgul Bekleme

```
volatile int sayi = 0;
//Seven segment sayacı
void islem_1() {
	volatile unsigned long delay;
	uint8_t kodlar[] = {
		0b00111111,
		0b00000110,
		0b01011011,
		0b01001111,
		0b01100110,
		0b01101101,
		0b01111101,
		0b00000111,
		0b01111111,
		0b01101111
	};
	GPIO_PORTB_DATA_R = kodlar[sayi];


}
```

Mesgul Bekleme

```
volatile int sayi = 0;
//Seven segment sayacı
void islem_1() {
	volatile unsigned long delay;
	uint8_t kodlar[] = {
		0b00111111,
		0b00000110,
		0b01011011,
		0b01001111,
		0b01100110,
		0b01101101,
		0b01111101,
		0b00000111,
		0b01111111,
		0b01101111
	};
	GPIO_PORTB_DATA_R = kodlar[sayi];
	for (delay = 0 ; delay < 2000000 ; delay++);

}
```

Mesgul Bekleme

```
volatile int sayi = 0;
//Seven segment sayacı
void islem_1() {
	volatile unsigned long delay;
	uint8_t kodlar[] = {
		0b00111111,
		0b00000110,
		0b01011011,
		0b01001111,
		0b01100110,
		0b01101101,
		0b01111101,
		0b00000111,
		0b01111111,
		0b01101111
	};
	GPIO_PORTB_DATA_R = kodlar[sayi];
	for (delay = 0 ; delay < 2000000 ; delay++);
	sayi = (sayi + 1) % 10;
}
```

Mesgul Bekleme

```
// button basiliysa ledi yak, degilse sondur
void islem_2() {



}
```

Mesgul Bekleme

```
// button basiliysa ledi yak, degilse sondur
void islem_2() {
	if (GPIO_PORTF_DATA_R & 0b00001 == 0) {

	 }


}
```

Mesgul Bekleme

```
// button basiliysa ledi yak, degilse sondur
void islem_2() {
	if (GPIO_PORTF_DATA_R & 0b00001 == 0) {
		GPIO_PORTF_DATA_R |= 0b00100;
	 }


}
```

Mesgul Bekleme

```
// button basiliysa ledi yak, degilse sondur
void islem_2() {
	if (GPIO_PORTF_DATA_R & 0b00001 == 0) {
		GPIO_PORTF_DATA_R |= 0b00100;
	} else {
		GPIO_PORTF_DATA_R &= ~0b00100;
	}
}
```

Mesgul Bekleme

```
int main() {




}
```

Mesgul Bekleme

```
int main() {
        init_port_B();
        init_port_F();




}
```

Mesgul Bekleme

```
int main() {
        init_port_B();
        init_port_F();
        // surekli islem_1 ve islem_2'yi yap
        while (1) {



        }
}
```

Mesgul Bekleme

```
int main() {
        init_port_B();
        init_port_F();
        // surekli islem_1 ve islem_2'yi yap
        while (1) {
                islem_1();
                islem_2();
        }
}
```

Sorular