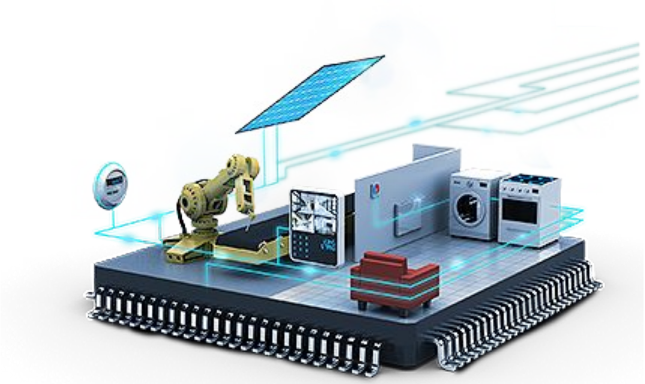
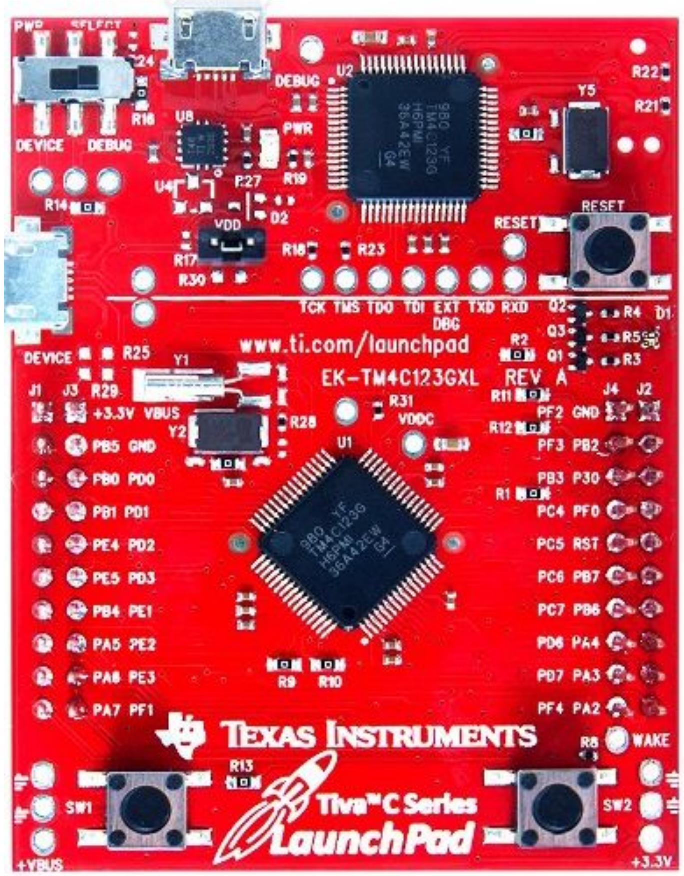
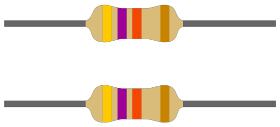
Seri Port &
7 Segment Display

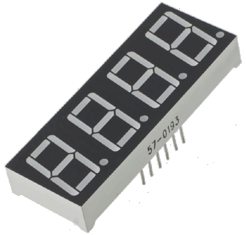


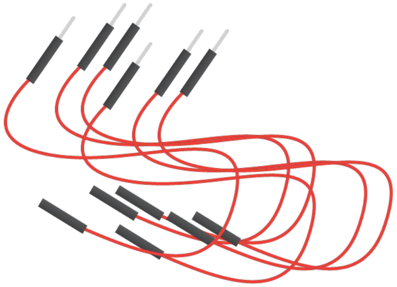
Suhap SAHIN

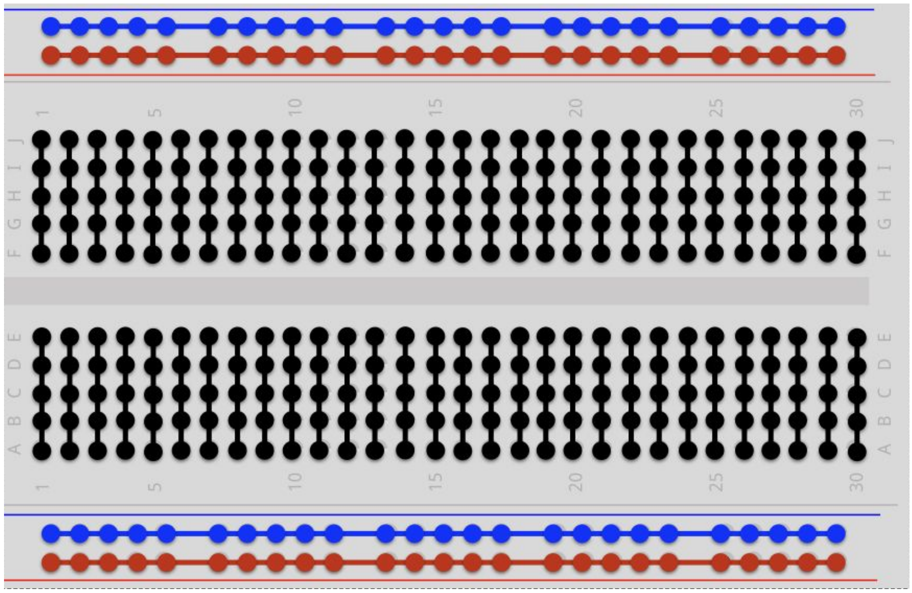
7 Parçalı Gösterge



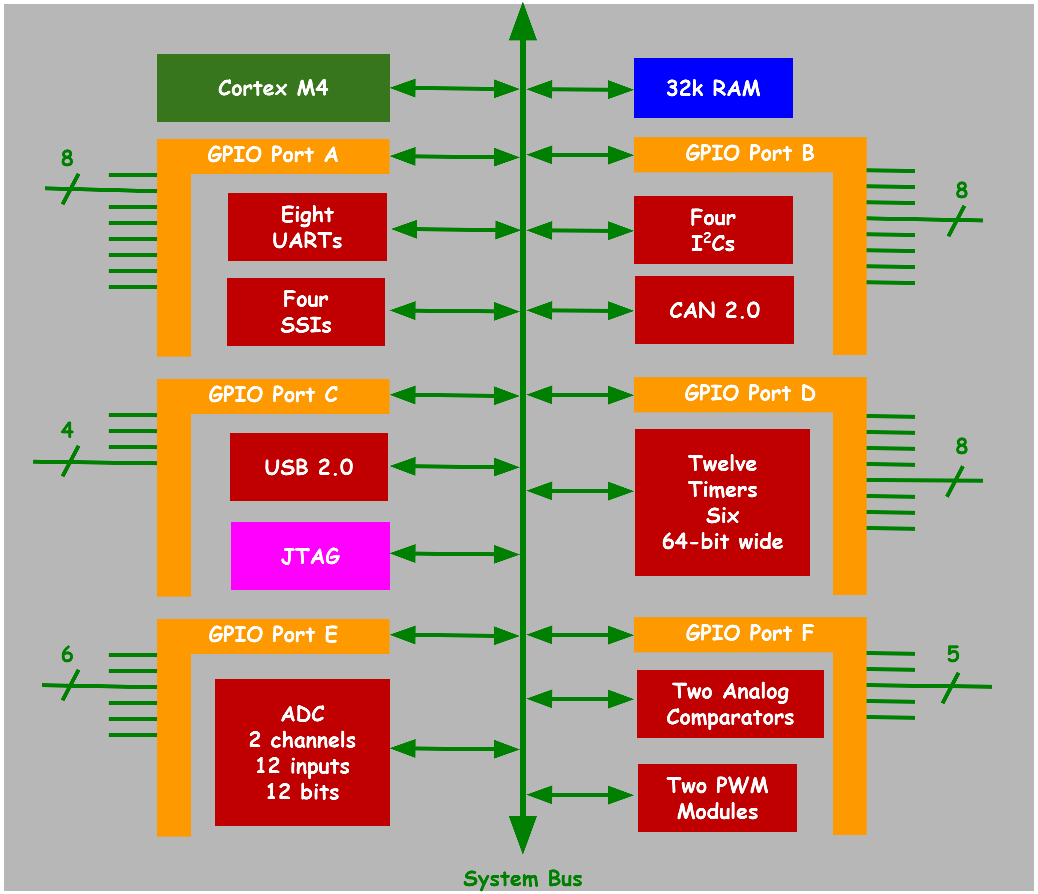




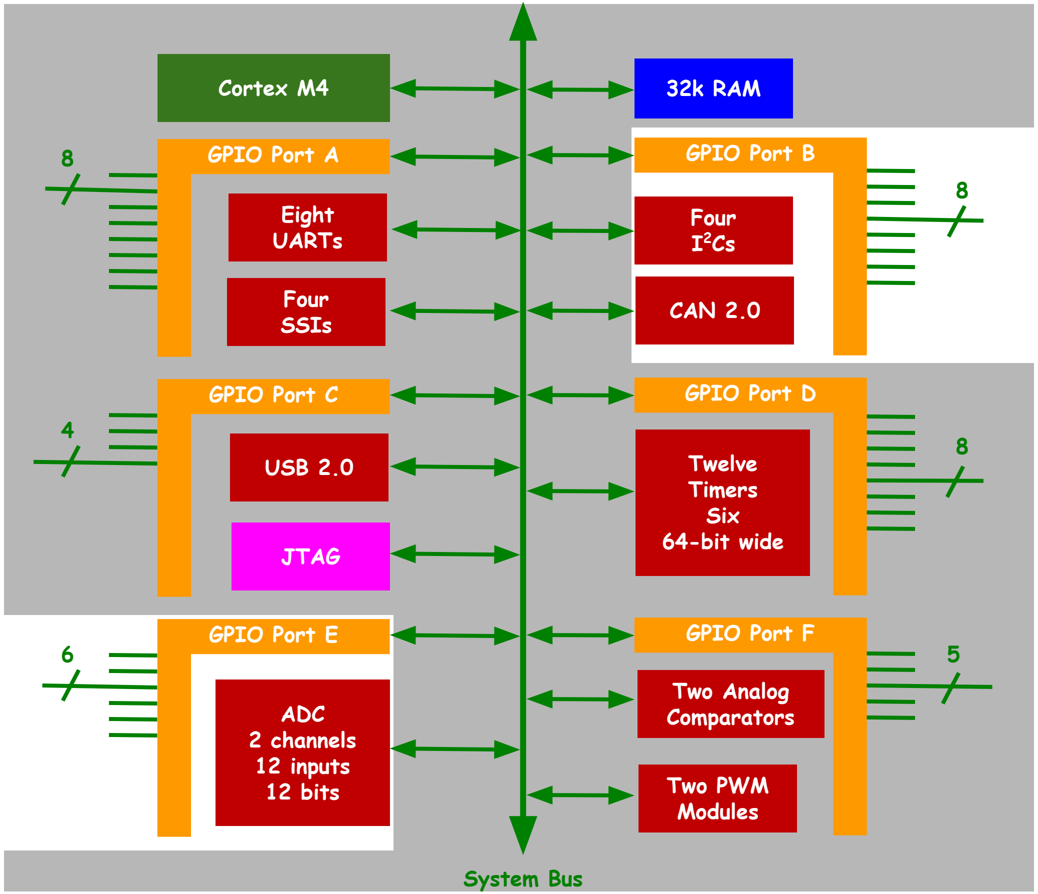




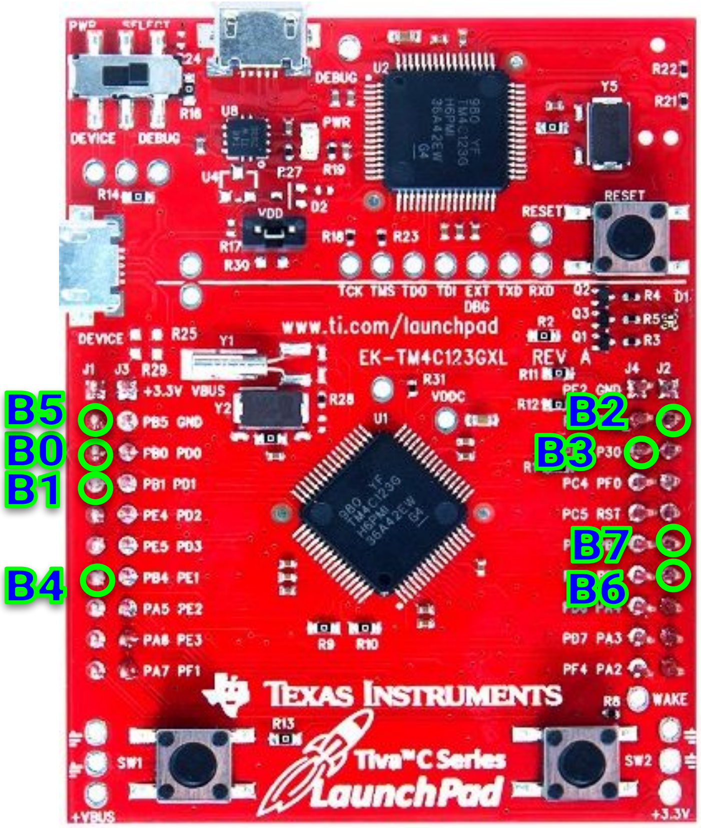
7 Parçalı Gösterge

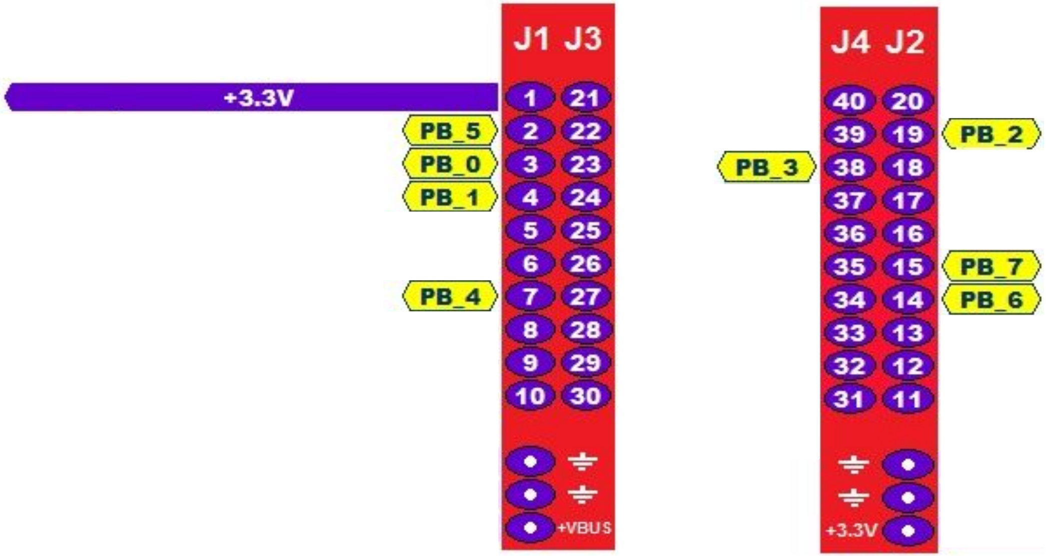


7 Parçalı Gösterge

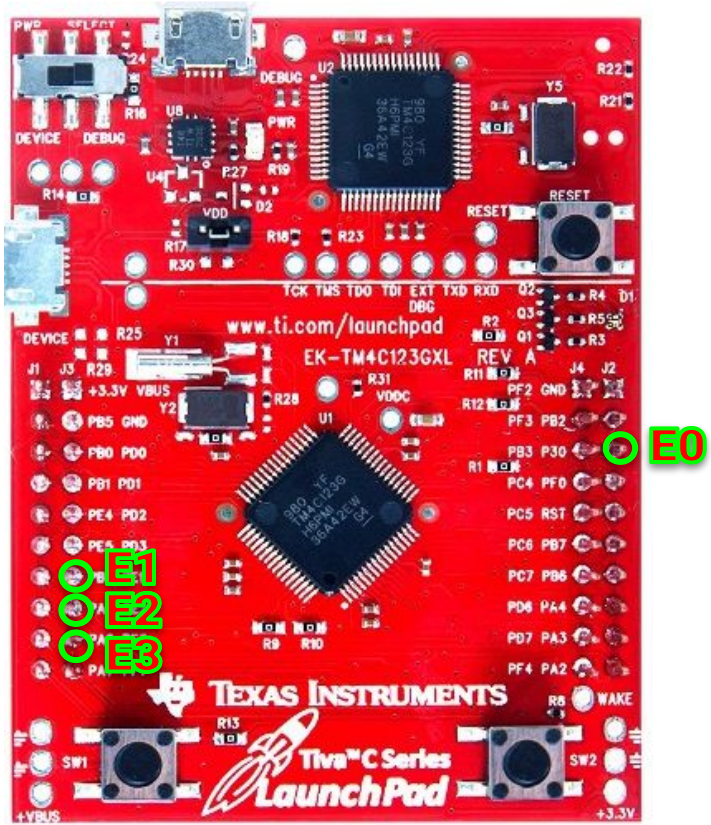


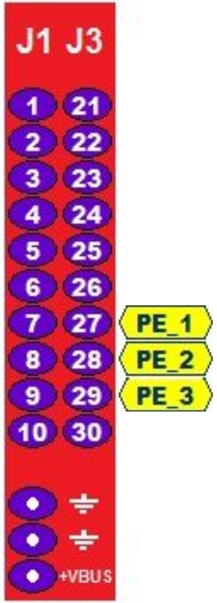
Tiva & Stellaris Port Baglantıları

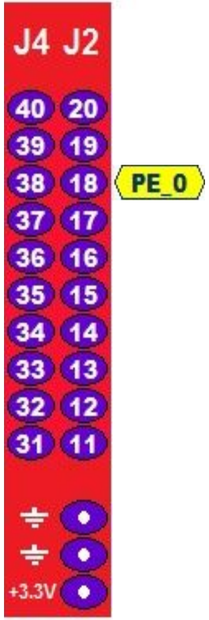




Tiva & Stellaris Port Baglantıları



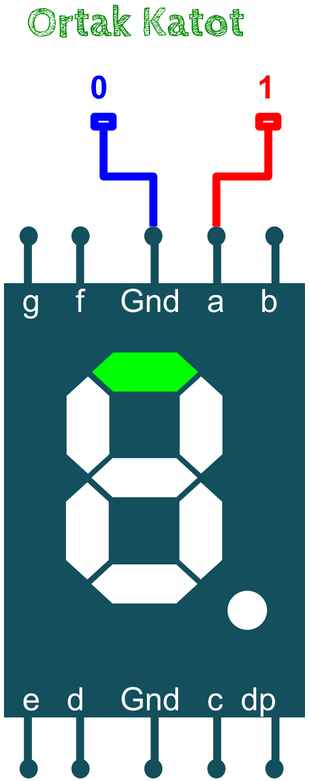


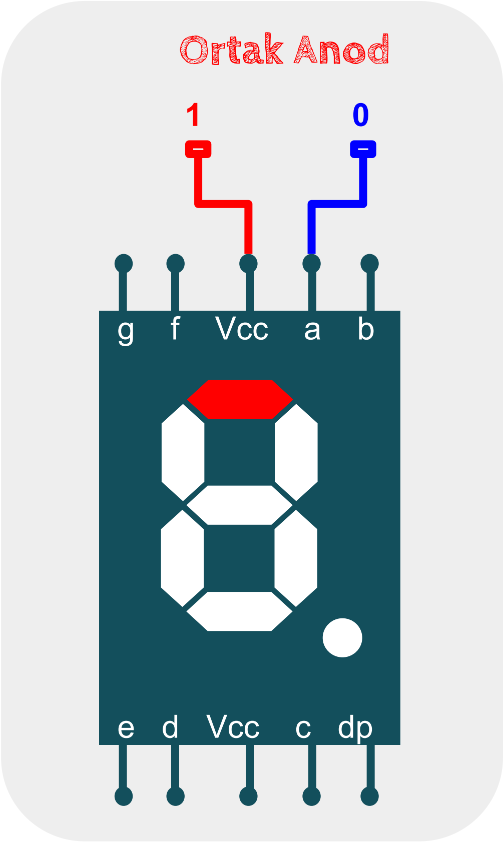


7 Parçalı Gösterge

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

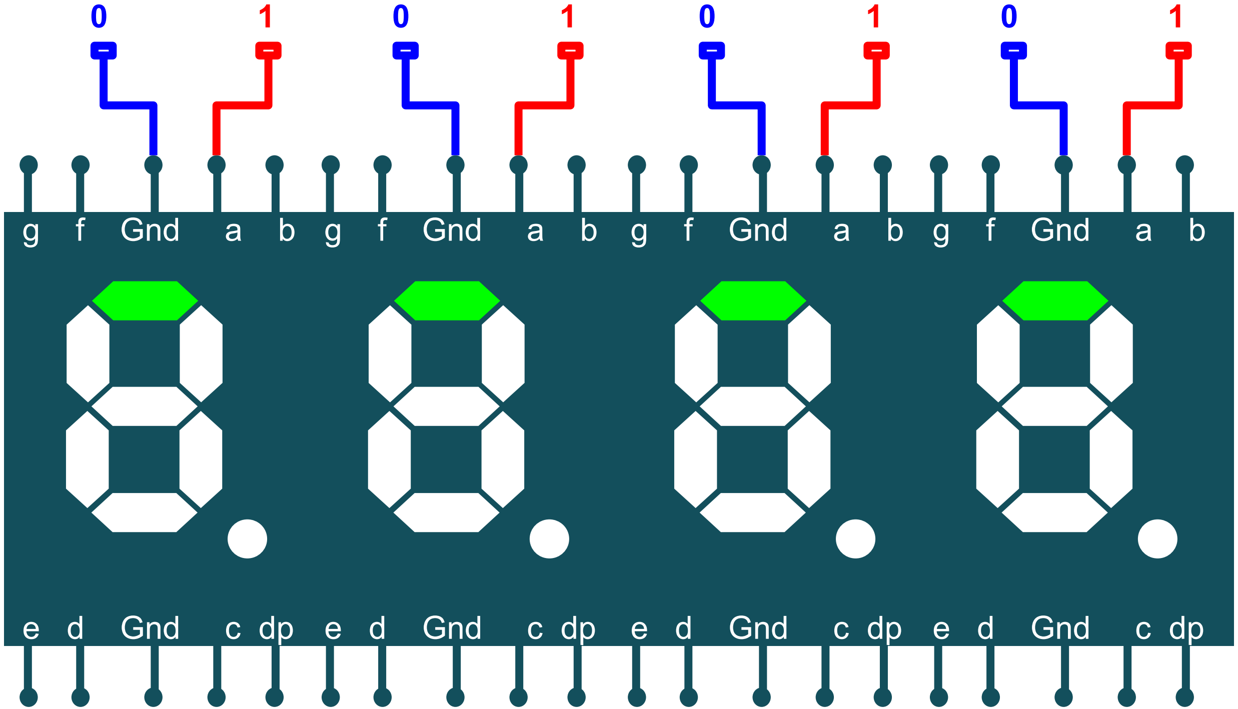
7 Parçalı Gösterge





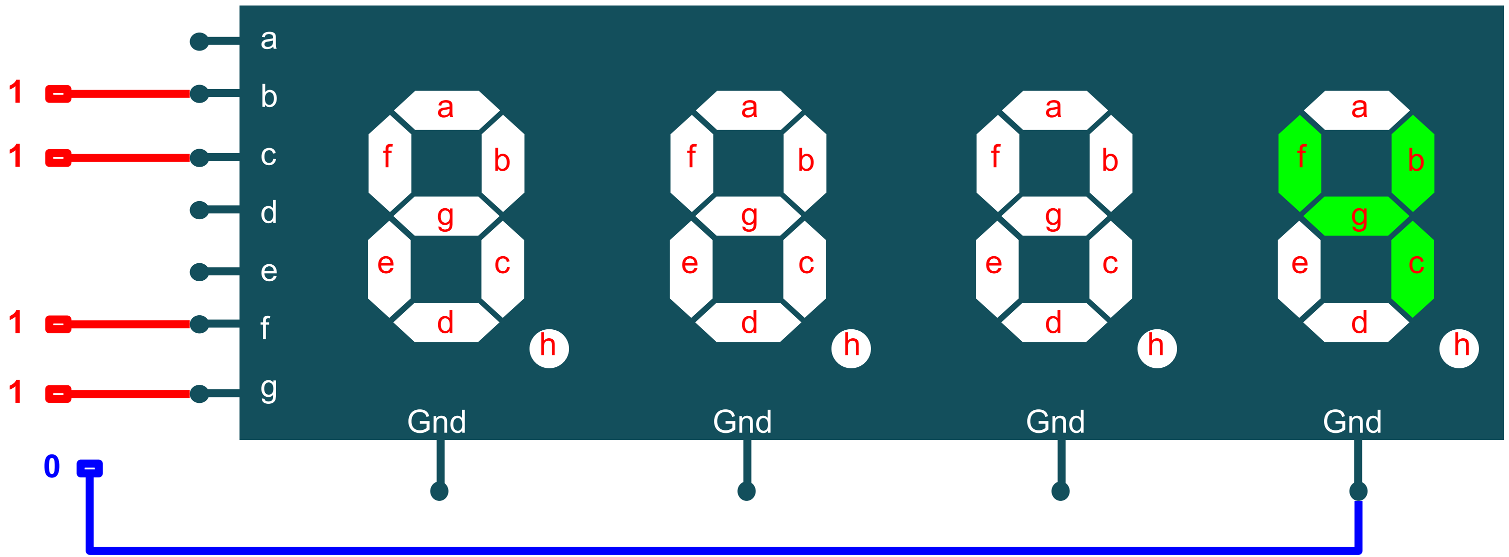
7 Parçalı Gösterge

Ortak Katot



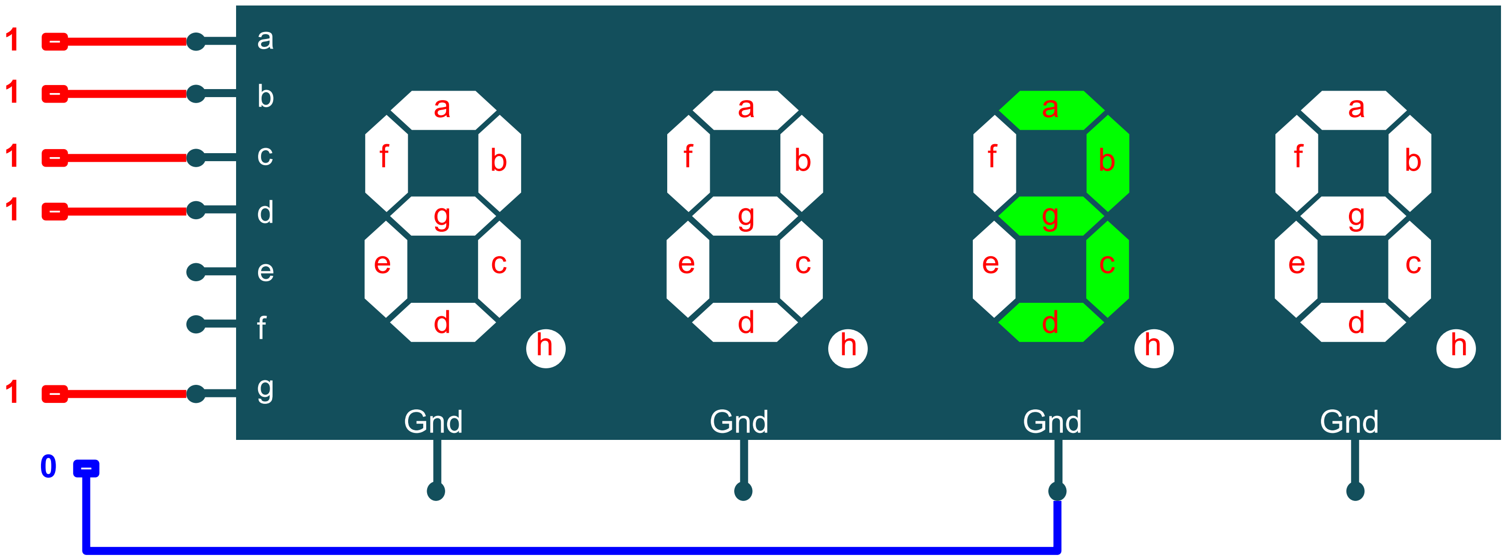
7 Parçalı Gösterge

Ortak Katot



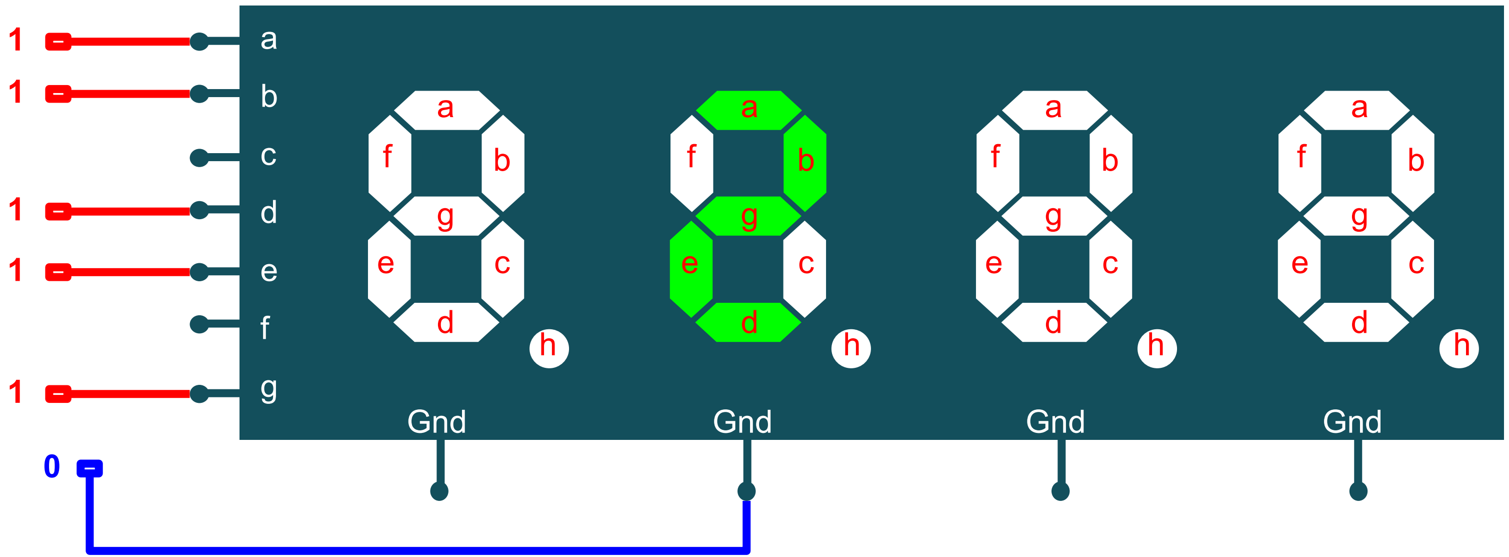
7 Parçalı Gösterge

Ortak Katot



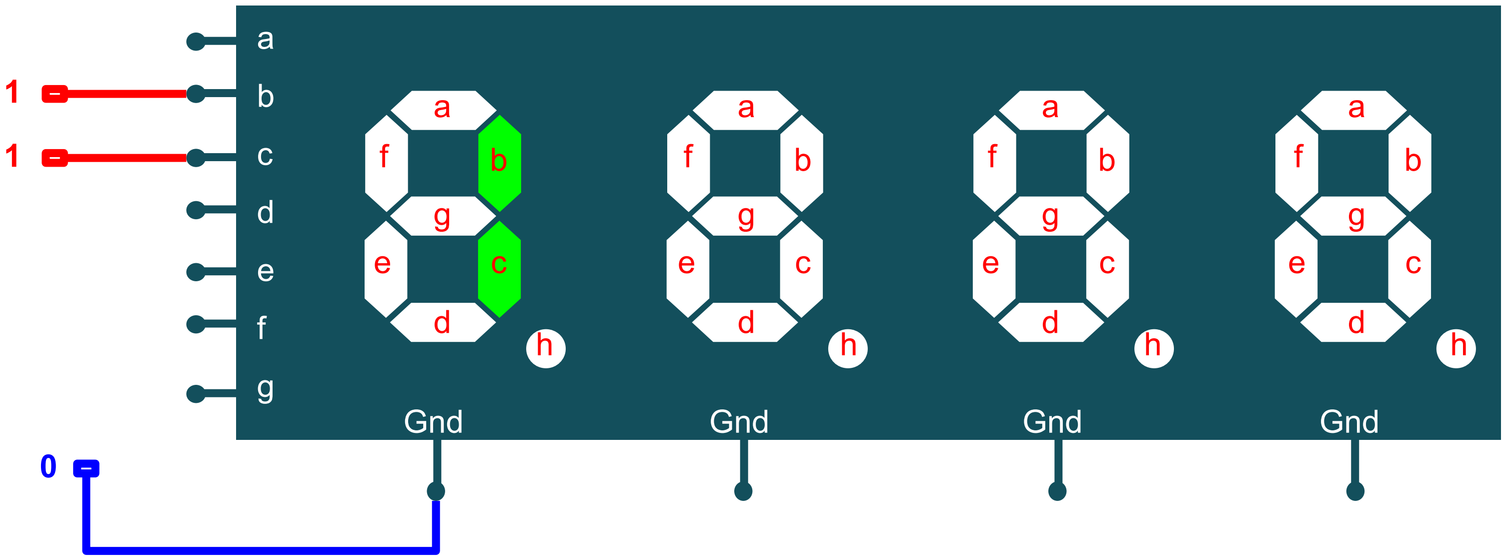
7 Parçalı Gösterge

Ortak Katot

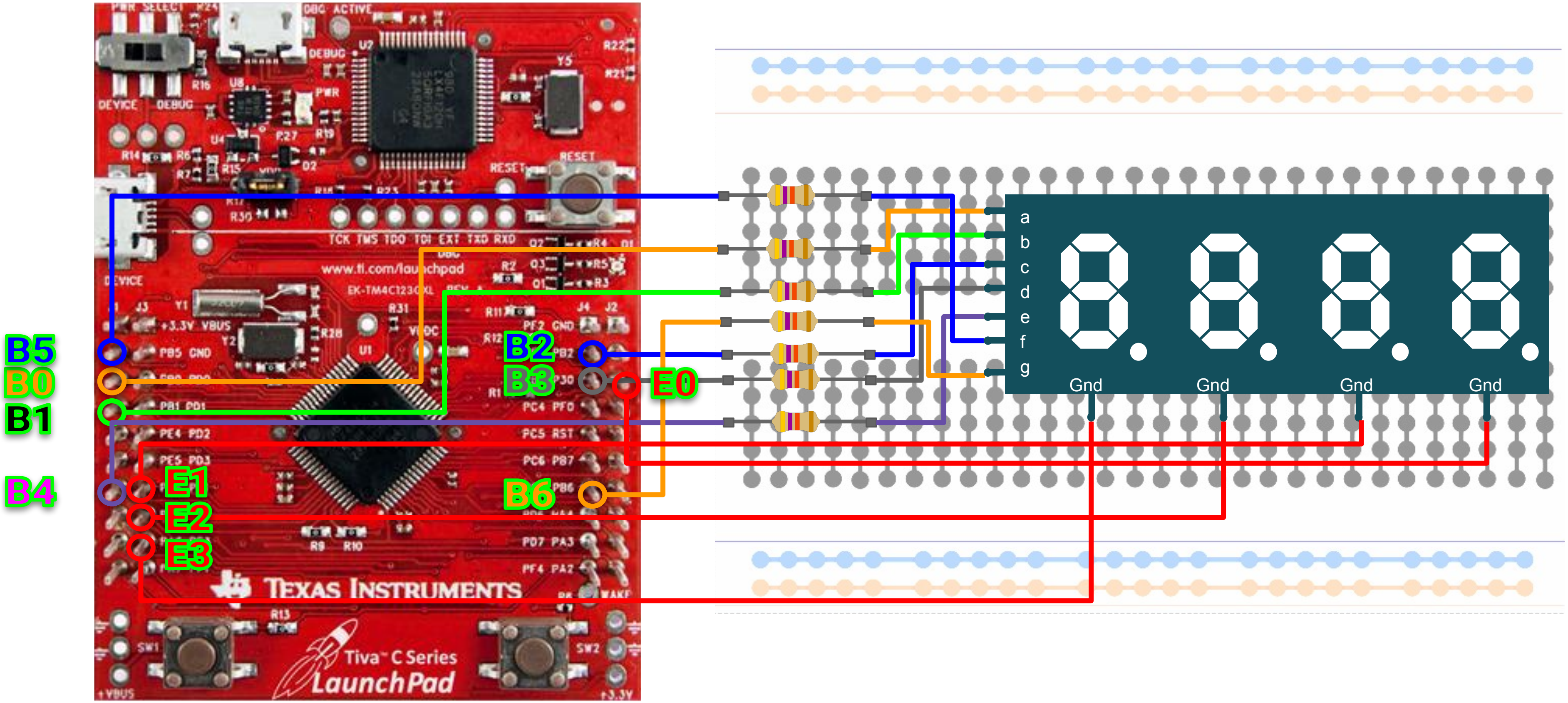


7 Parçalı Gösterge

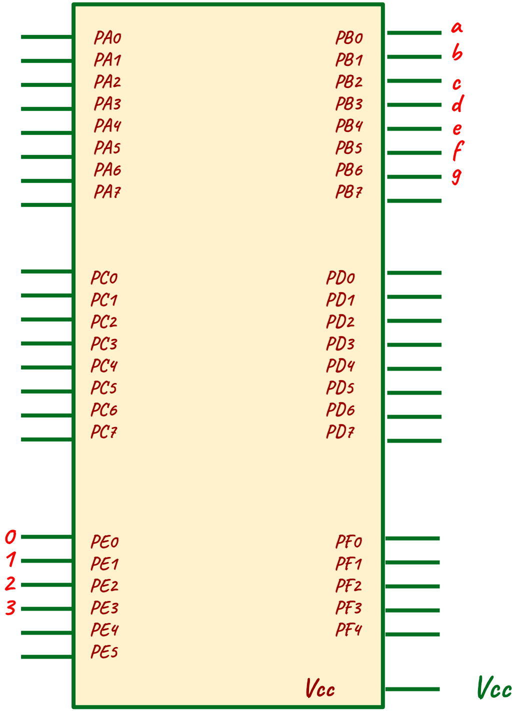
Ortak Katot

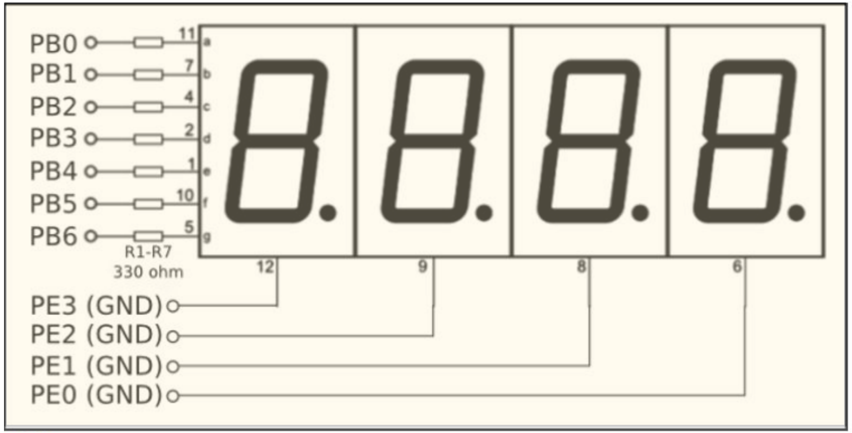


Elektronik Devre Kurulumu



Elektronik Devre Kurulumu





Port B 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



}
```

Port B 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


}
```

Port B Kurulumu

```c
#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

}
```

Port B 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
}
```

Port E Kurulumu

```
void init_port_E() {
        volatile unsigned long delay;
        SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOE;          // Port E’nin saati aktifleştir
        delay = SYSCTL_RCGC2_R;                        // Gecikme
        GPIO_PORTE_DIR_R |= 0x0F;                      // Port E’nin tüm bitleri cikis olarak ayarla
        GPIO_PORTE_AFSEL_R &= ~0x0F;                   // Alternatif fonksiyonlar kapat
        GPIO_PORTE_DEN_R |= 0x0F;                      // Tüm Port E için Digital çalışmayı aktifleştir
}
```

Seven Segment Tablosu

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

4 Digit Seven Segment

```
int main() {
   volatile unsigned long delay;
   // Seven Segment Tablosu buraya yazılacak










}
```

4 Digit Seven Segment

```
int main() {
  volatile unsigned long delay;
  // Seven Segment Tablosu buraya yazılacak
  init_port_B();
  init_port_E();


}
```

4 Digit Seven Segment

```
int main() {
  volatile unsigned long delay;
  // Seven Segment Tablosu buraya yazılacak
  init_port_B();
  init_port_E();
  int sayi = 1234;
  const int BEKLEME_LIMIT = 10000;








}
```

4 Digit Seven Segment

```
int main() {
  volatile unsigned long delay;
  // Seven Segment Tablosu buraya yazılacak
  init_port_B();
  init_port_E();
  int sayi = 1234;
  const int BEKLEME_LIMIT = 10000;
  while (1) {

  }
}
```

4 Digit Seven Segment

```
int main() {
  volatile unsigned long delay;
  // Seven Segment Tablosu buraya yazılacak
  init_port_B();
  init_port_E();
  int sayi = 1234;
  const int BEKLEME_LIMIT = 10000;
  while (1) {
     int birler = sayi % 10;

  }
}
```

4 Digit Seven Segment

```
int main() {
    volatile unsigned long delay;
    // Seven Segment Tablosu buraya yazılacak
    init_port_B();
    init_port_E();
    int sayi = 1234;
    const int BEKLEME_LIMIT = 10000;
    while (1) {
        int birler = sayi % 10;
        GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat




















    }
}
```

4 Digit Seven Segment

```
int main() {
  volatile unsigned long delay;
  // Seven Segment Tablosu buraya yazılacak
  init_port_B();
  init_port_E();
  int sayi = 1234;
  const int BEKLEME_LIMIT = 10000;
  while (1) {
    int birler = sayi % 10;
    GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat
    GPIO_PORTB_DATA_R = kodlar[birler];

  }
}
```

4 Digit Seven Segment

```c
int main() {
    volatile unsigned long delay;
    // Seven Segment Tablosu buraya yazılacak
    init_port_B();
    init_port_E();
    int sayi = 1234;
    const int BEKLEME_LIMIT = 10000;
    while (1) {
        int birler = sayi % 10;
        GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat
        GPIO_PORTB_DATA_R = kodlar[birler];
        GPIO_PORTE_DATA_R &= ~0b0001; // birler basamagini aktiflestir














    }
}
```

4 Digit Seven Segment

```c
int main() {
  volatile unsigned long delay;
  // Seven Segment Tablosu buraya yazılacak
  init_port_B();
  init_port_E();
  int sayi = 1234;
  const int BEKLEME_LIMIT = 10000;
  while (1) {
    int birler = sayi % 10;
    GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat
    GPIO_PORTB_DATA_R = kodlar[birler];
    GPIO_PORTE_DATA_R &= ~0b0001; // birler basamagini aktiflestir
    for (delay = 0 ; delay < BEKLEME_LIMIT ; delay++)/* bekle */;







}
```

4 Digit Seven Segment

```c
int main() {
    volatile unsigned long delay;
    // Seven Segment Tablosu buraya yazılacak
    init_port_B();
    init_port_E();
    int sayi = 1234;
    const int BEKLEME_LIMIT = 10000;
    while (1) {
        int birler = sayi % 10;
        GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat
        GPIO_PORTB_DATA_R = kodlar[birler];
        GPIO_PORTE_DATA_R &= ~0b0001; // birler basamagini aktiflestir
        for (delay = 0 ; delay < BEKLEME_LIMIT ; delay++)/* bekle */;
        int onlar = (sayi / 10) % 10;

    }
}
```

4 Digit Seven Segment

```
int main() {
    volatile unsigned long delay;
    // Seven Segment Tablosu buraya yazılacak
    init_port_B();
    init_port_E();
    int sayi = 1234;
    const int BEKLEME_LIMIT = 10000;
    while (1) {
        int birler = sayi % 10;
        GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat
        GPIO_PORTB_DATA_R = kodlar[birler];
        GPIO_PORTE_DATA_R &= ~0b0001; // birler basamagini aktiflestir
        for (delay = 0 ; delay < BEKLEME_LIMIT ; delay++)/* bekle */;
        int onlar = (sayi / 10) % 10;
        GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat
        GPIO_PORTB_DATA_R = kodlar[onlar];
        GPIO_PORTE_DATA_R &= ~0b0010; // onlar basamagini aktiflestir
        for (delay = 0 ; delay < BEKLEME_LIMIT ; delay++)/* bekle */;









    }
}
```

4 Digit Seven Segment

```
int main() {
  volatile unsigned long delay;
  // Seven Segment Tablosu buraya yazılacak
  init_port_B();
  init_port_E();
  int sayi = 1234;
  const int BEKLEME_LIMIT = 10000;
  while (1) {
     int birler = sayi % 10;
     GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat
     GPIO_PORTB_DATA_R = kodlar[birler];
     GPIO_PORTE_DATA_R &= ~0b0001; // birler basamagini aktiflestir
     for (delay = 0 ; delay < BEKLEME_LIMIT ; delay++)/* bekle */;
     int onlar = (sayi / 10) % 10;
     GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat
     GPIO_PORTB_DATA_R = kodlar[onlar];
     GPIO_PORTE_DATA_R &= ~0b0010; // onlar basamagini aktiflestir
     for (delay = 0 ; delay < BEKLEME_LIMIT ; delay++)/* bekle */;
     int yuzler = (sayi / 100) % 10;
     GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat
     GPIO_PORTB_DATA_R = kodlar[yuzler];
     GPIO_PORTE_DATA_R &= ~0b0100; // yuzler basamagini aktiflestir
     for (delay = 0 ; delay < BEKLEME_LIMIT ; delay++) /* bekle */;




  }
}
```

4 Digit Seven Segment

```
int main() {
    volatile unsigned long delay;
    // Seven Segment Tablosu buraya yazılacak
    init_port_B();
    init_port_E();
    int sayi = 1234;
    const int BEKLEME_LIMIT = 10000;
    while (1) {
        int birler = sayi % 10;
        GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat
        GPIO_PORTB_DATA_R = kodlar[birler];
        GPIO_PORTE_DATA_R &= ~0b0001; // birler basamagini aktiflestir
        for (delay = 0 ; delay < BEKLEME_LIMIT ; delay++)/* bekle */;
        int onlar = (sayi / 10) % 10;
        GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat
        GPIO_PORTB_DATA_R = kodlar[onlar];
        GPIO_PORTE_DATA_R &= ~0b0010; // onlar basamagini aktiflestir
        for (delay = 0 ; delay < BEKLEME_LIMIT ; delay++)/* bekle */;
        int yuzler = (sayi / 100) % 10;
        GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat
        GPIO_PORTB_DATA_R = kodlar[yuzler];
        GPIO_PORTE_DATA_R &= ~0b0100; // yuzler basamagini aktiflestir
        for (delay = 0 ; delay < BEKLEME_LIMIT ; delay++) /* bekle */;
        int binler = (sayi / 1000) % 10;
        GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat
        GPIO_PORTB_DATA_R = kodlar[binler];
        GPIO_PORTE_DATA_R &= ~0b1000; // binler basamagini aktiflestir
        for (delay = 0 ; delay < BEKLEME_LIMIT ; delay++)/* bekle */;
    }
}
```

4 Digit Seven Segment

```
int main() {
    volatile unsigned long delay;
    // Seven Segment Tablosu buraya yazılacak
    init_port_B();
    init_port_E();
    int sayi = 1234;
    const int BEKLEME_LIMIT = 10000;
    while (1) {
        int birler = sayi % 10;
        GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat
        GPIO_PORTB_DATA_R = kodlar[birler];
        GPIO_PORTE_DATA_R &= ~0b0001; // birler basamagini aktiflestir
        for (delay = 0 ; delay < BEKLEME_LIMIT ; delay++)/* bekle */;
        int onlar = (sayi / 10) % 10;
        GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat
        GPIO_PORTB_DATA_R = kodlar[onlar];
        GPIO_PORTE_DATA_R &= ~0b0010; // onlar basamagini aktiflestir
        for (delay = 0 ; delay < BEKLEME_LIMIT ; delay++)/* bekle */;
        int yuzler = (sayi / 100) % 10;
        GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat
        GPIO_PORTB_DATA_R = kodlar[yuzler];
        GPIO_PORTE_DATA_R &= ~0b0100; // yuzler basamagini aktiflestir
        for (delay = 0 ; delay < BEKLEME_LIMIT ; delay++) /* bekle */;
        int binler = (sayi / 1000) % 10;
        GPIO_PORTE_DATA_R |= 0b1111; // hepsini kapat
        GPIO_PORTB_DATA_R = kodlar[binler];
        GPIO_PORTE_DATA_R &= ~0b1000; // binler basamagini aktiflestir
        for (delay = 0 ; delay < BEKLEME_LIMIT ; delay++)/* bekle */;
    }
}
```

```
sayi =sayi+1;
```

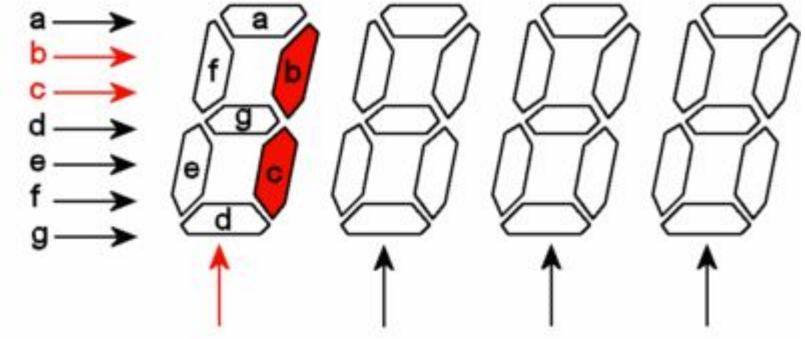
Es Zamanlı islem yapma

islem_2

islem_1

Es Zamanlı islem yapma

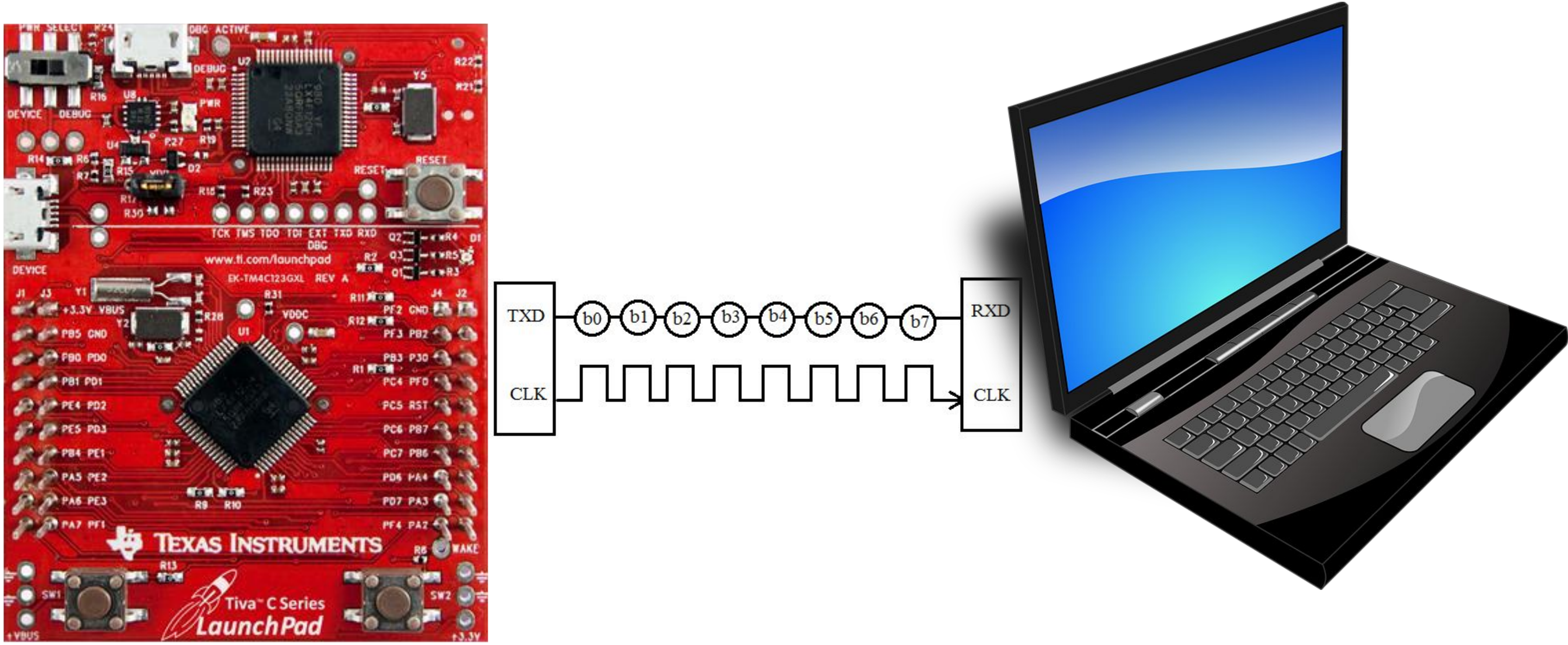
islem_1



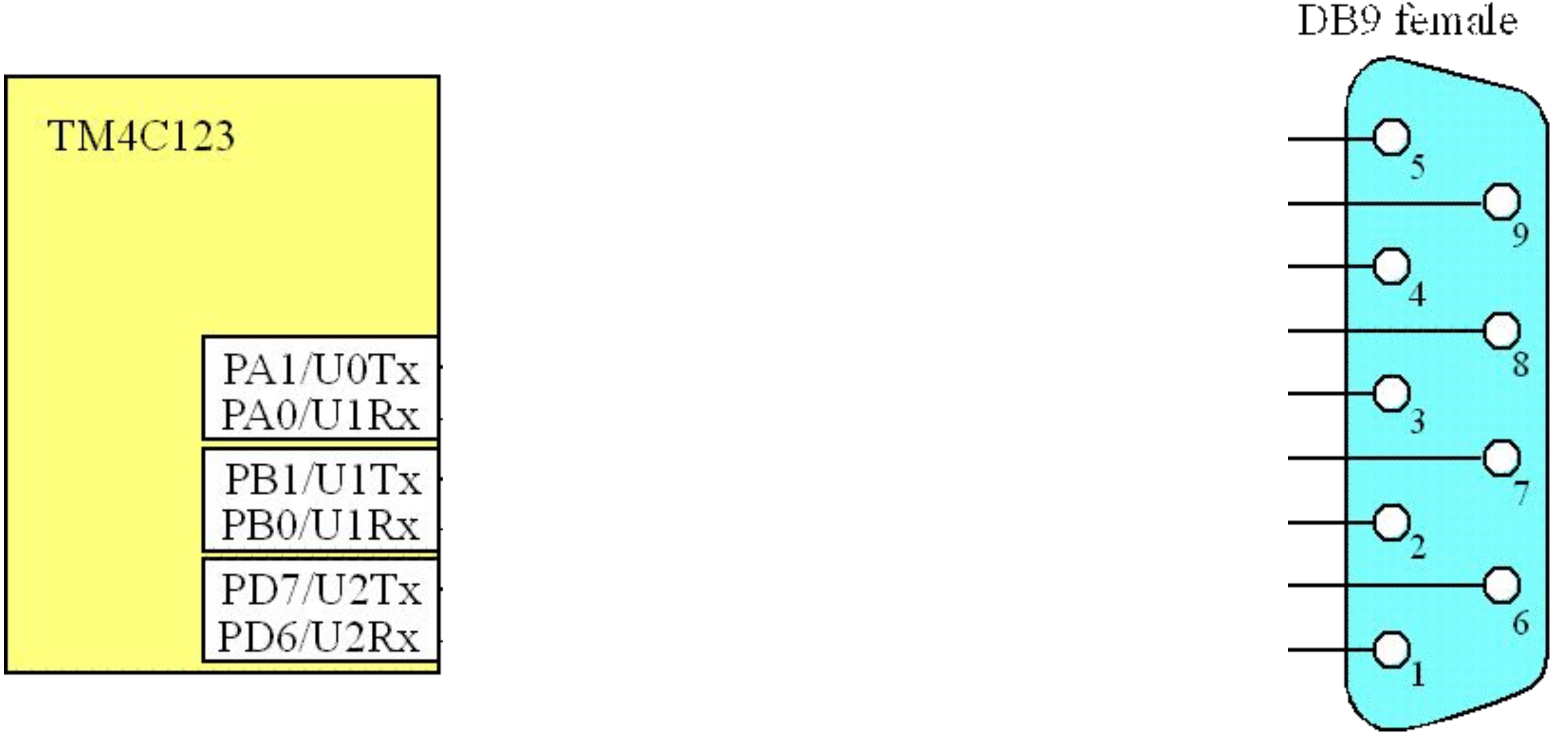
islem_2

```
sayi = sayi+1;
```

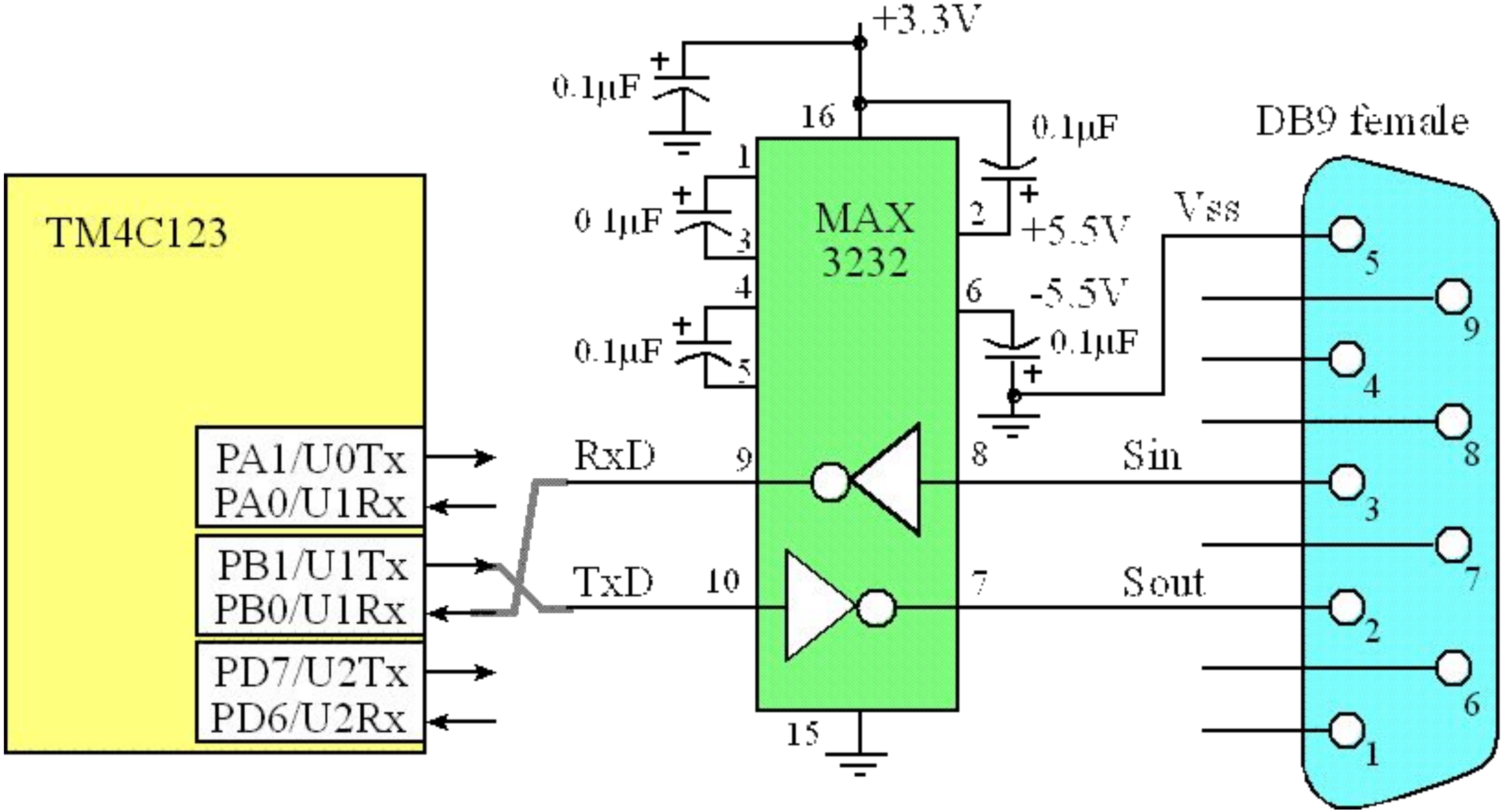
Seri Porttan iletisim



Tiva & Stellaris Port Baglantıları

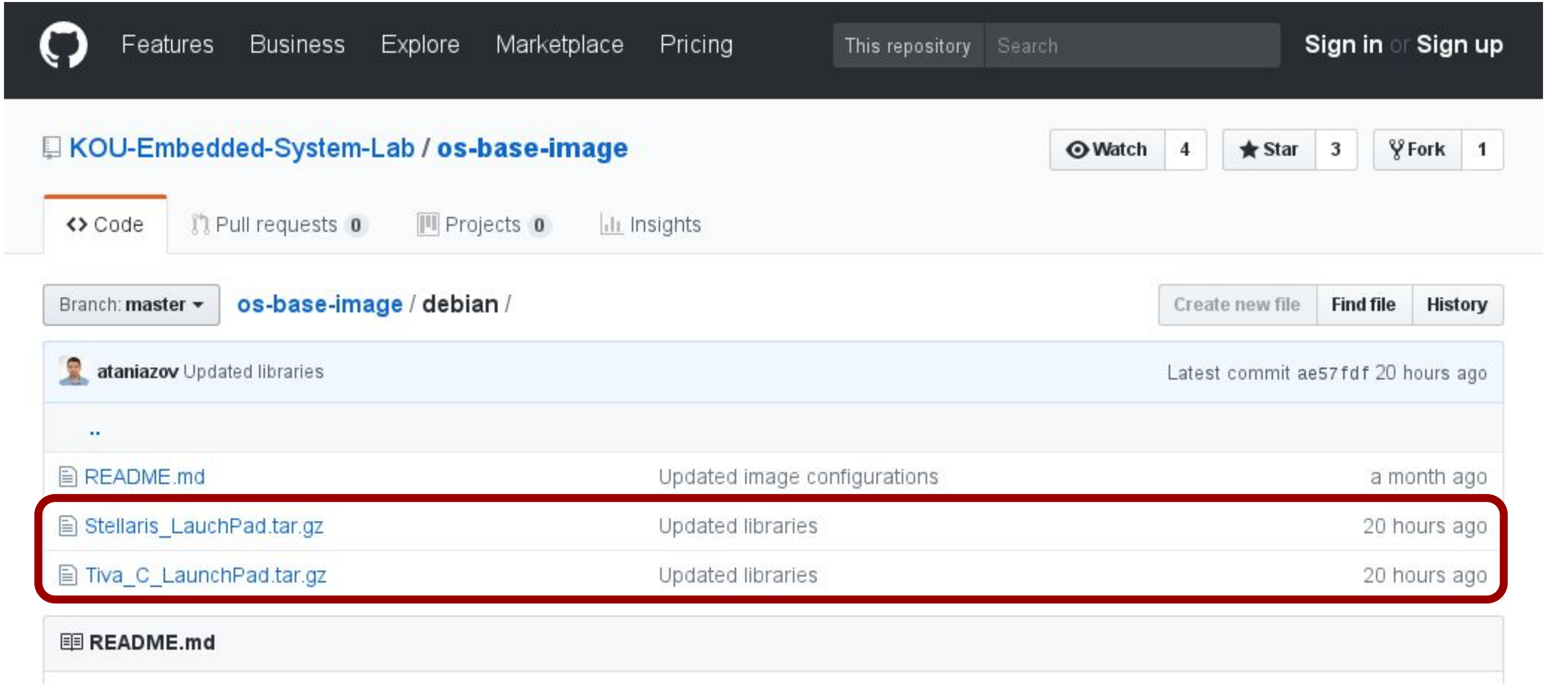


Tiva & Stellaris Port Baglantıları



Seri Port Proje Ayarları

https://github.com/KOU-Embedded-System-Lab/os-base-image/tree/master/debian



KOU-Embedded-System-Lab / os-base-image

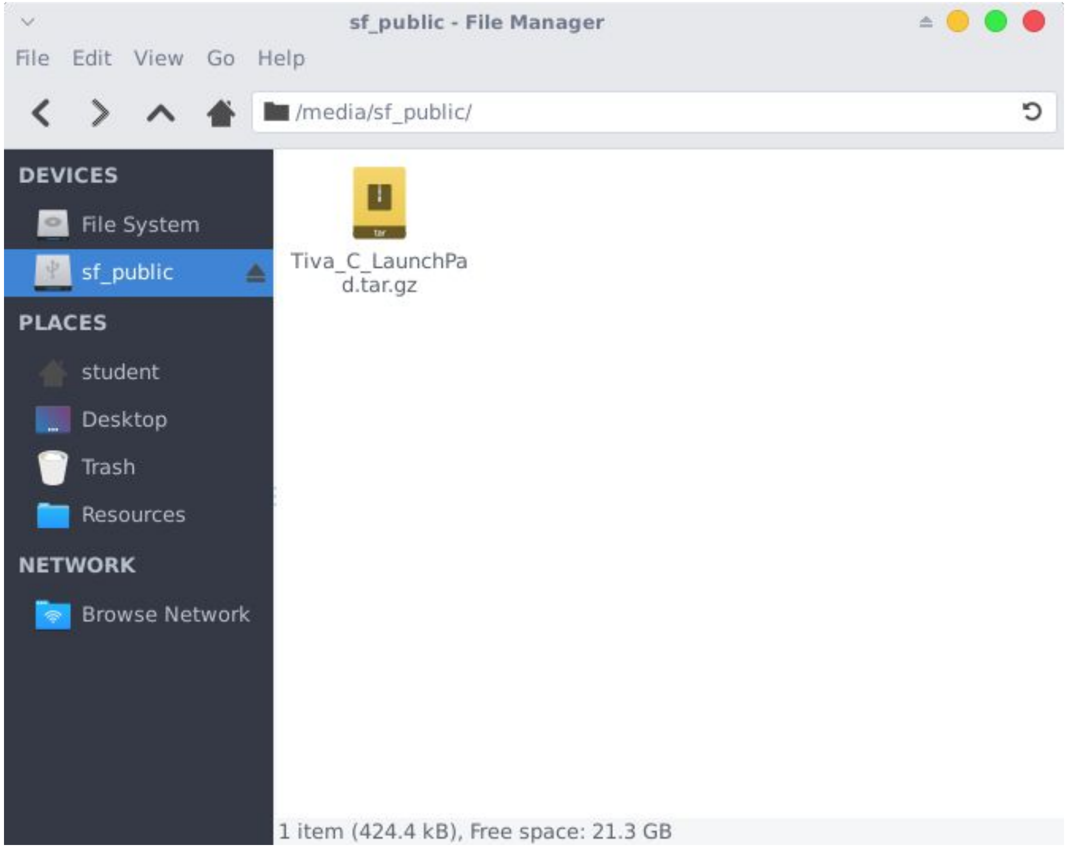
os-base-image / debian /

ataniazov Updated libraries Latest commit ae57fdf 20 hours ago

..		
README.md	Updated image configurations	a month ago
Stellaris_LauchPad.tar.gz	Updated libraries	20 hours ago
Tiva_C_LaunchPad.tar.gz	Updated libraries	20 hours ago

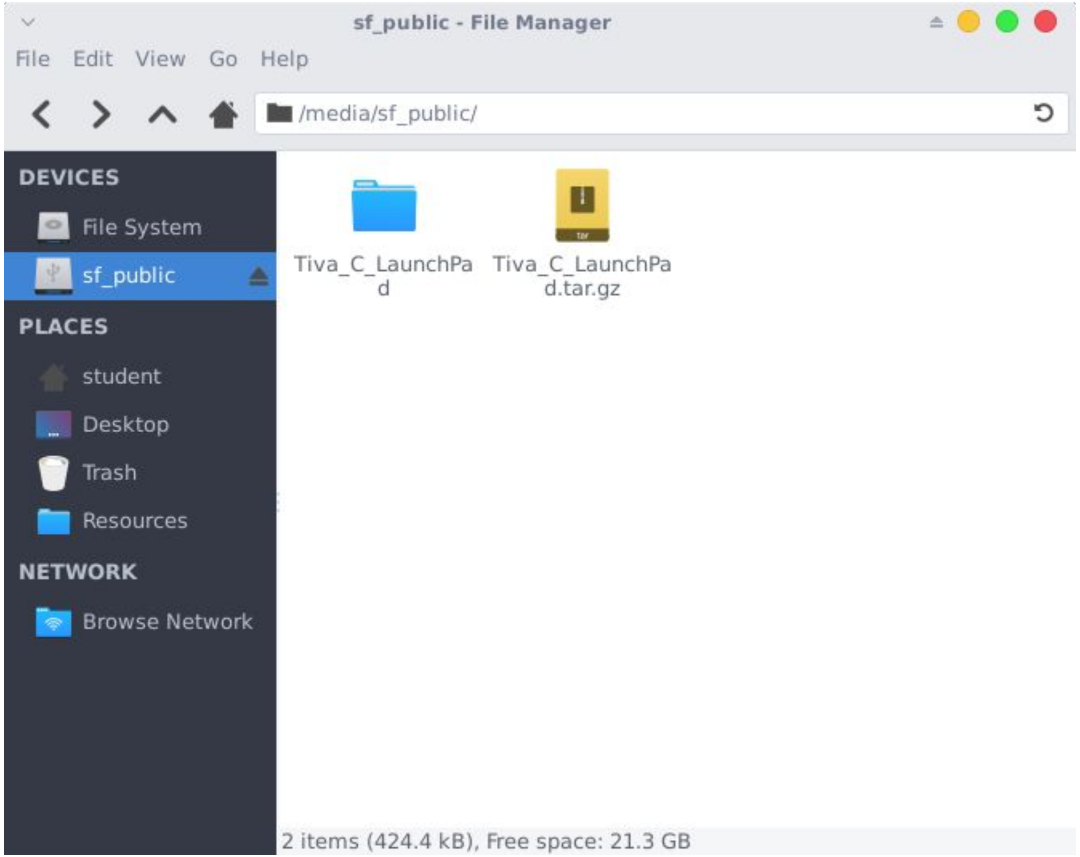
README.md

Seri Port Proje Ayarları

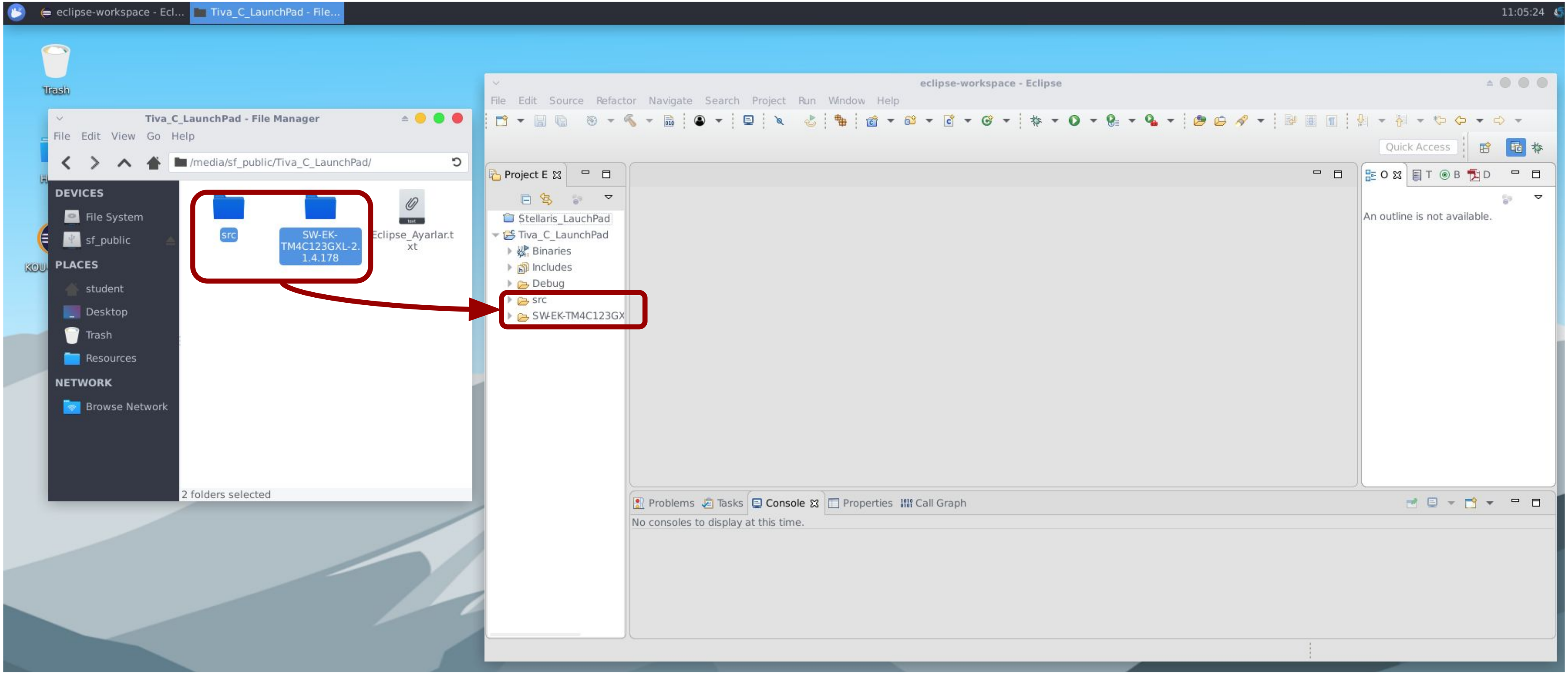


1 item (424.4 kB), Free space: 21.3 GB

Seri Port Proje Ayarları

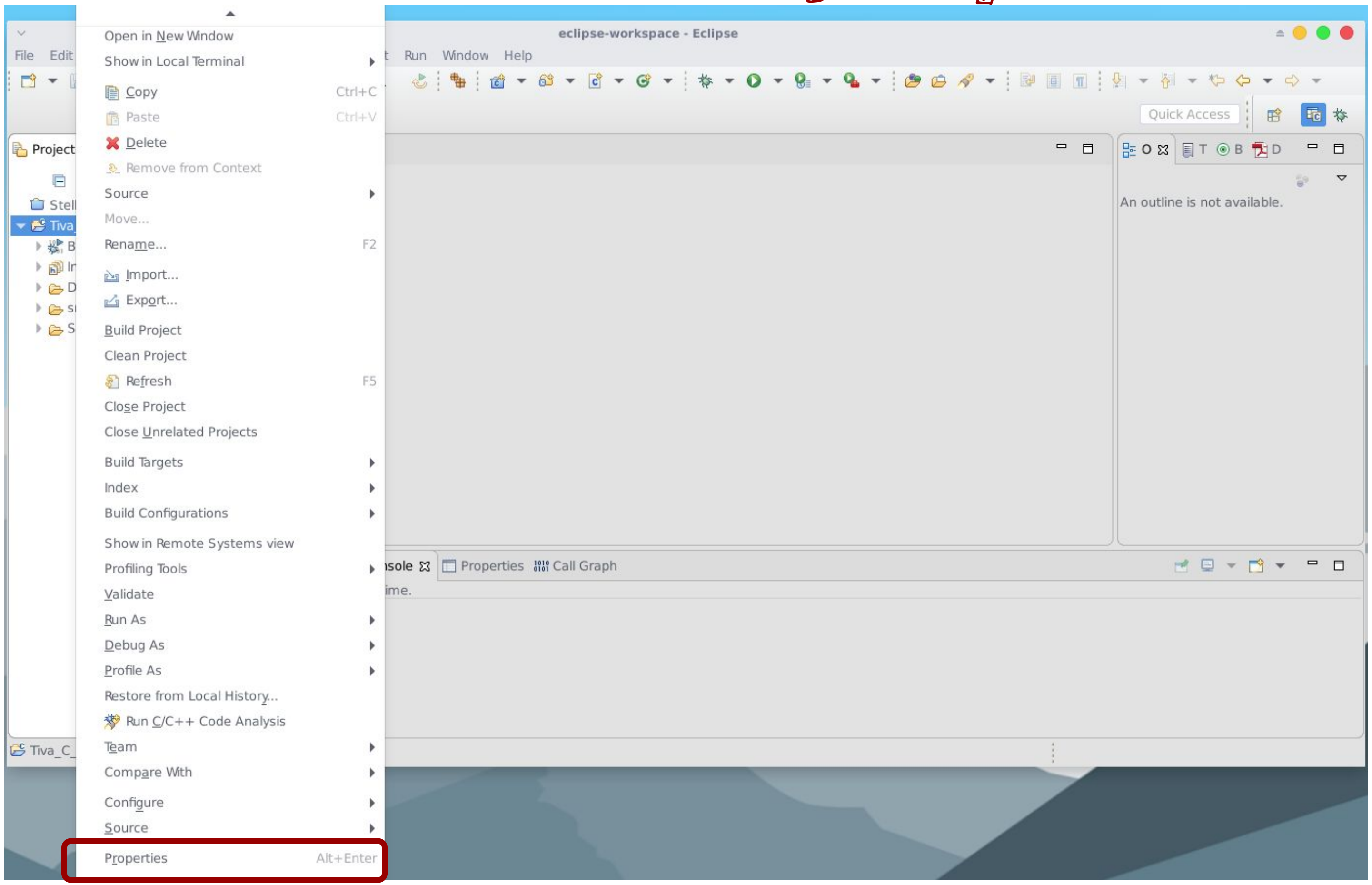


Seri Port Proje Ayarları

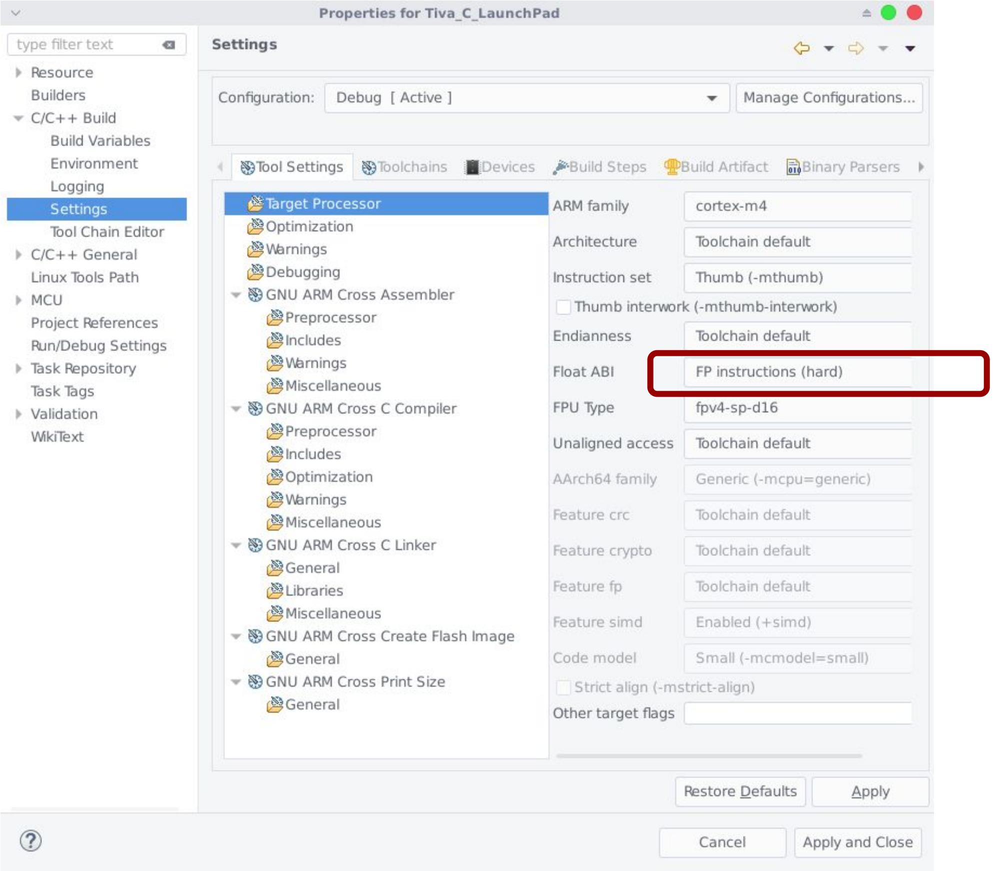


eclipse-workspace - Ecl...
Tiva_C_LaunchPad - File...
11:05:24
Trash
Tiva_C_LaunchPad - File Manager
File Edit View Go Help
/media/sf_public/Tiva_C_LaunchPad/
DEVICES
File System
sf_public
PLACES
student
Desktop
Trash
Resources
NETWORK
Browse Network
src
SW-EK-TM4C123GXL-2.1.4.178
Eclipse_Ayarlar.txt
2 folders selected
eclipse-workspace - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Project E
Stellaris_LauchPad
Tiva_C_LaunchPad
Binaries
Includes
Debug
src
SW-EK-TM4C123GX
An outline is not available.
Problems Tasks Console Properties Call Graph
No consoles to display at this time.

Seri Port Proje Ayarları



Seri Port Proje Ayarları



Configuration: Debug [Active]

ARM family	cortex-m4
Architecture	Toolchain default
Instruction set	Thumb (-mthumb)
☐ Thumb interwork (-mthumb-interwork)	
Endianness	Toolchain default
Float ABI	FP instructions (hard)
FPU Type	fpv4-sp-d16
Unaligned access	Toolchain default
AArch64 family	Generic (-mcpu=generic)
Feature crc	Toolchain default
Feature crypto	Toolchain default
Feature fp	Toolchain default
Feature simd	Enabled (+simd)
Code model	Small (-mcmodel=small)
☐ Strict align (-mstrict-align)	
Other target flags	

Seri Port Kurulumu

```
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
// stellaris #include "inc/lm4f120h5qr.h"
#include <stdbool.h>
#include "inc/hw_types.h"
#include "driverlib/gpio.h"
#include "driverlib/sysctl.h"
#include "driverlib/uart.h"
#include "utils/uartstdio.h"
```

Seri Port Kurulumu

```
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
// stellaris #include "inc/lm4f120h5qr.h"
#include <stdbool.h>
#include "inc/hw_types.h"
#include "driverlib/gpio.h"
#include "driverlib/sysctl.h"
#include "driverlib/uart.h"
#include "utils/uartstdio.h"
// secenekler: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200
#define BAUDRATE 600
```

Seri Port Kurulumu

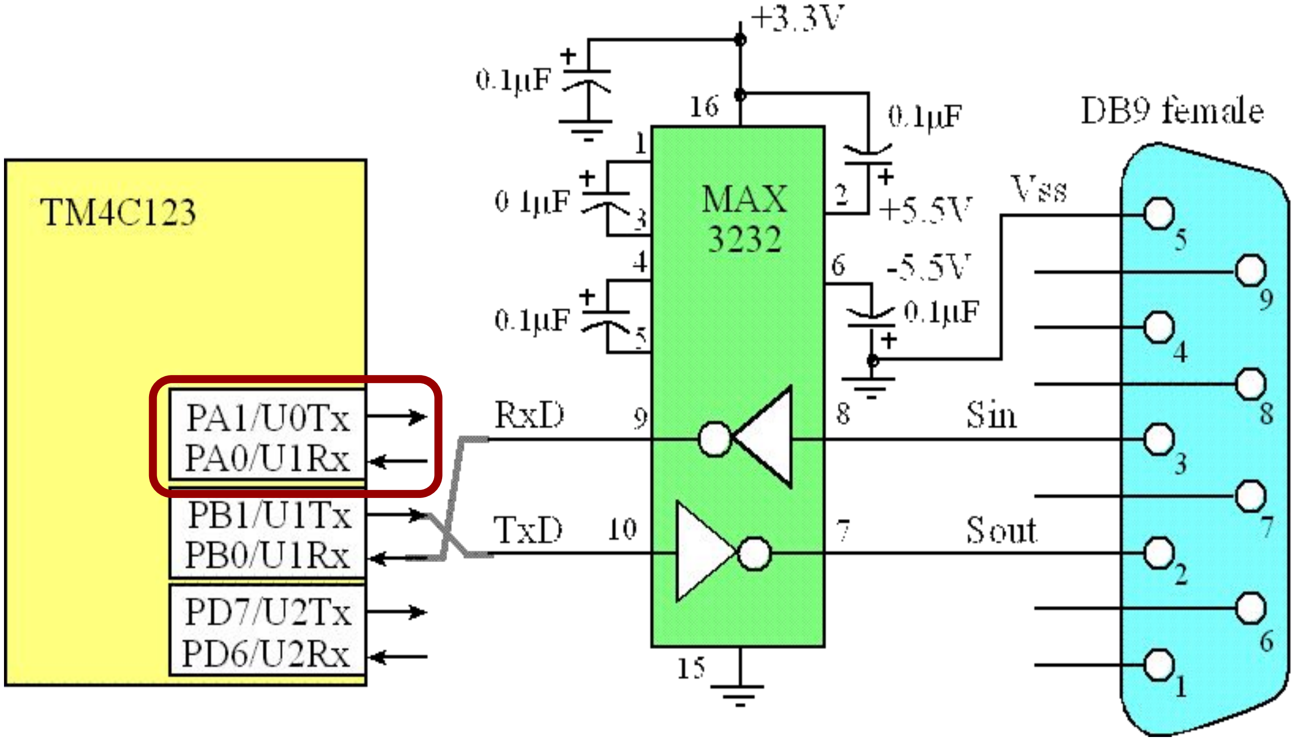
```c
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
// stellaris #include "inc/lm4f120h5qr.h"
#include <stdbool.h>
#include "inc/hw_types.h"
#include "driverlib/gpio.h"
#include "driverlib/sysctl.h"
#include "driverlib/uart.h"
#include "utils/uartstdio.h"
// secenekler: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200
#define BAUDRATE 600
/** UART (seri port) ayarini yapan fonksiyon */
void init_UARTstdio() {



}
```

Seri Port Kurulumu

```
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
// stellaris #include "inc/lm4f120h5qr.h"
#include <stdbool.h>
#include "inc/hw_types.h"
#include "driverlib/gpio.h"
#include "driverlib/sysctl.h"
#include "driverlib/uart.h"
#include "utils/uartstdio.h"
// secenekler: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200
#define BAUDRATE 600
/** UART (seri port) ayarini yapan fonksiyon */
void init_UARTstdio() {
	SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOA);






}
```

Seri Porttan iletisim

```
SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOA);
```



Seri Port Kurulumu

```
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
// stellaris #include "inc/lm4f120h5qr.h"
#include <stdbool.h>
#include "inc/hw_types.h"
#include "driverlib/gpio.h"
#include "driverlib/sysctl.h"
#include "driverlib/uart.h"
#include "utils/uartstdio.h"
// secenekler: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200
#define BAUDRATE 600
/** UART (seri port) ayarini yapan fonksiyon */
void init_UARTstdio() {
        SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOA);  // A portu SSI0 çevre birimi kullanım için etkinleştirilir.
        GPIOPinConfigure(0x00000001);





}
```

Seri Port Kurulumu

```
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
// stellaris #include "inc/lm4f120h5qr.h"
#include <stdbool.h>
#include "inc/hw_types.h"
#include "driverlib/gpio.h"
#include "driverlib/sysctl.h"
#include "driverlib/uart.h"
#include "utils/uartstdio.h"
// secenekler: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200
#define BAUDRATE 600
/** UART (seri port) ayarini yapan fonksiyon */
void init_UARTstdio() {
        SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOA);
        GPIOPinConfigure(0x00000001);
        GPIOPinConfigure(0x00000401);




}
```

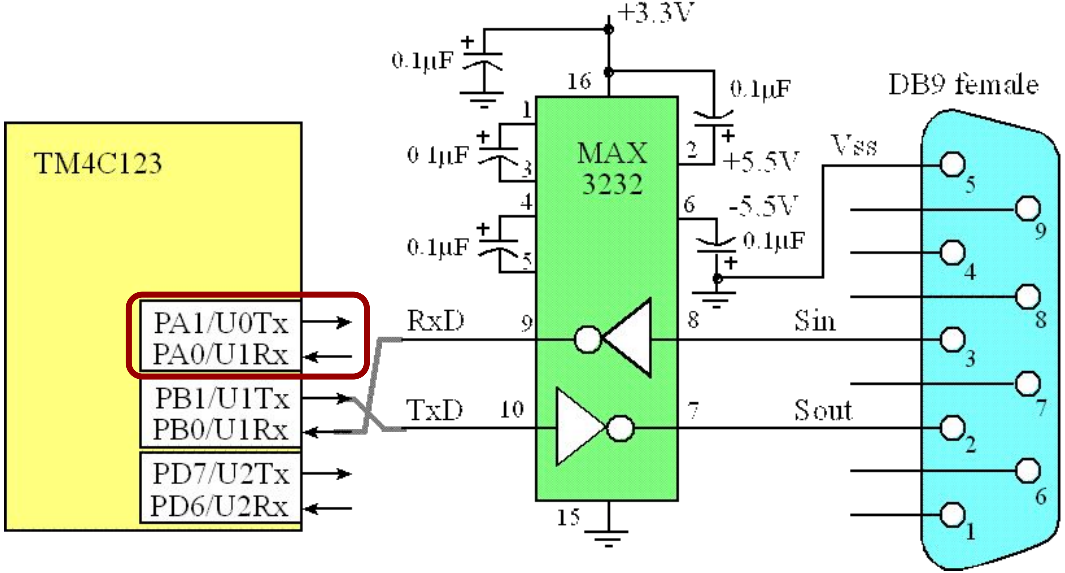
Seri Port Kurulumu

```c
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
// stellaris #include "inc/lm4f120h5qr.h"
#include <stdbool.h>
#include "inc/hw_types.h"
#include "driverlib/gpio.h"
#include "driverlib/sysctl.h"
#include "driverlib/uart.h"
#include "utils/uartstdio.h"
// secenekler: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200
#define BAUDRATE 600
/** UART (seri port) ayarini yapan fonksiyon */
void init_UARTstdio() {
        SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOA);
        GPIOPinConfigure(0x00000001);
        GPIOPinConfigure(0x00000401);
        GPIOPinTypeUART(0x40004000, 0x00000001 | 0x00000002);



}
```

Seri Porttan iletisim

```
GPIOPinConfigure(0x00000001);
GPIOPinConfigure(0x00000401);
GPIOPinTypeUART(0x40004000, 0x00000001 | 0x00000002);
```

// RX:PA0 TX:PA1



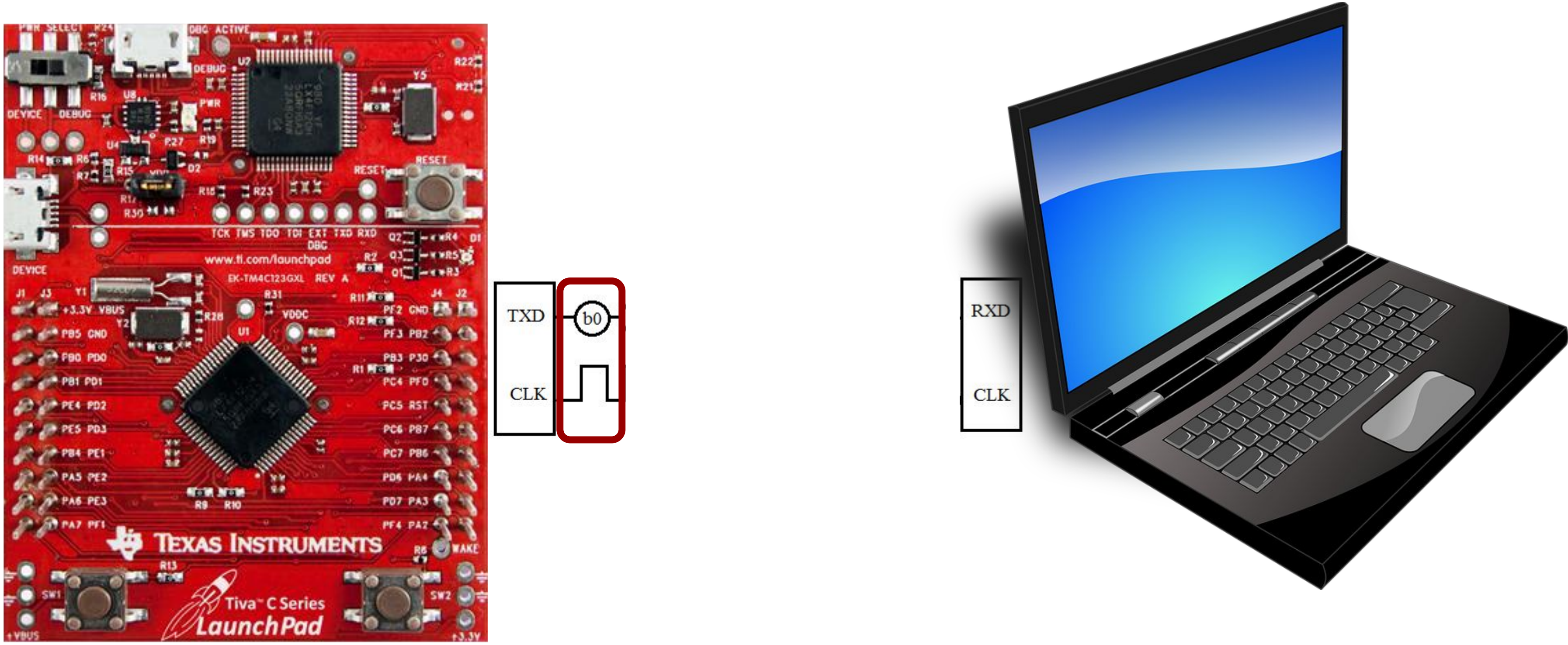
Seri Port Kurulumu

```
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
// stellaris #include "inc/lm4f120h5qr.h"
#include <stdbool.h>
#include "inc/hw_types.h"
#include "driverlib/gpio.h"
#include "driverlib/sysctl.h"
#include "driverlib/uart.h"
#include "utils/uartstdio.h"
// secenekler: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200
#define BAUDRATE 600
/** UART (seri port) ayarini yapan fonksiyon */
void init_UARTstdio() {
        SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOA);
        GPIOPinConfigure(0x00000001);
        GPIOPinConfigure(0x00000401);
        GPIOPinTypeUART(0x40004000, 0x00000001 | 0x00000002);
        UARTConfigSetExpClk(0x40004000, SysCtlClockGet(), BAUDRATE, (UART_CONFIG_WLEN_8 | UART_CONFIG_STOP_ONE |
                        UART_CONFIG_PAR_NONE));

}
```

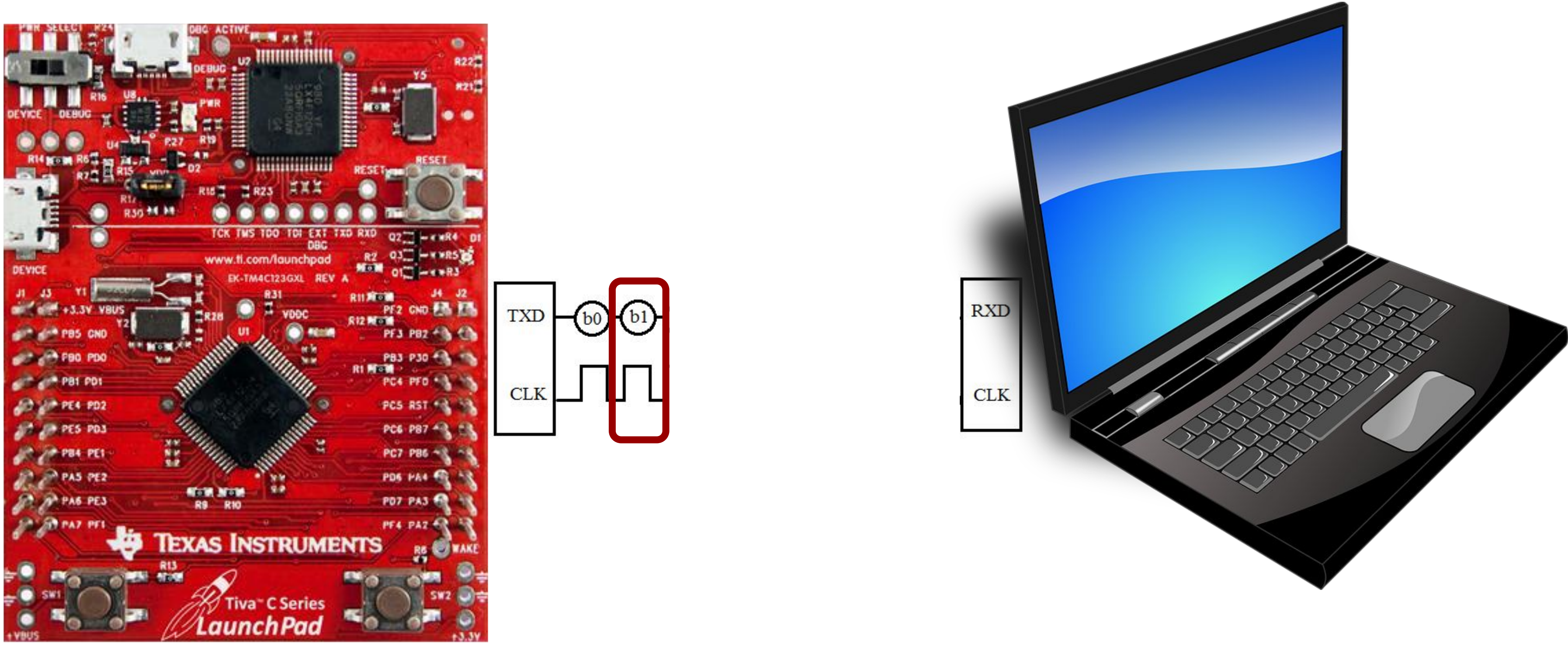
Seri Port Kurulumu

```
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
// stellaris #include "inc/lm4f120h5qr.h"
#include <stdbool.h>
#include "inc/hw_types.h"
#include "driverlib/gpio.h"
#include "driverlib/sysctl.h"
#include "driverlib/uart.h"
#include "utils/uartstdio.h"
// secenekler: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200
#define BAUDRATE 600
/** UART (seri port) ayarini yapan fonksiyon */
void init_UARTstdio() {
        SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOA);
        GPIOPinConfigure(0x00000001);
        GPIOPinConfigure(0x00000401);
        GPIOPinTypeUART(0x40004000, 0x00000001 | 0x00000002);
        UARTConfigSetExpClk(0x40004000, SysCtlClockGet(), BAUDRATE, (UART_CONFIG_WLEN_8 | UART_CONFIG_STOP_ONE |
                        UART_CONFIG_PAR_NONE));
        UARTStdioConfig(0, BAUDRATE, SysCtlClockGet());
}
```

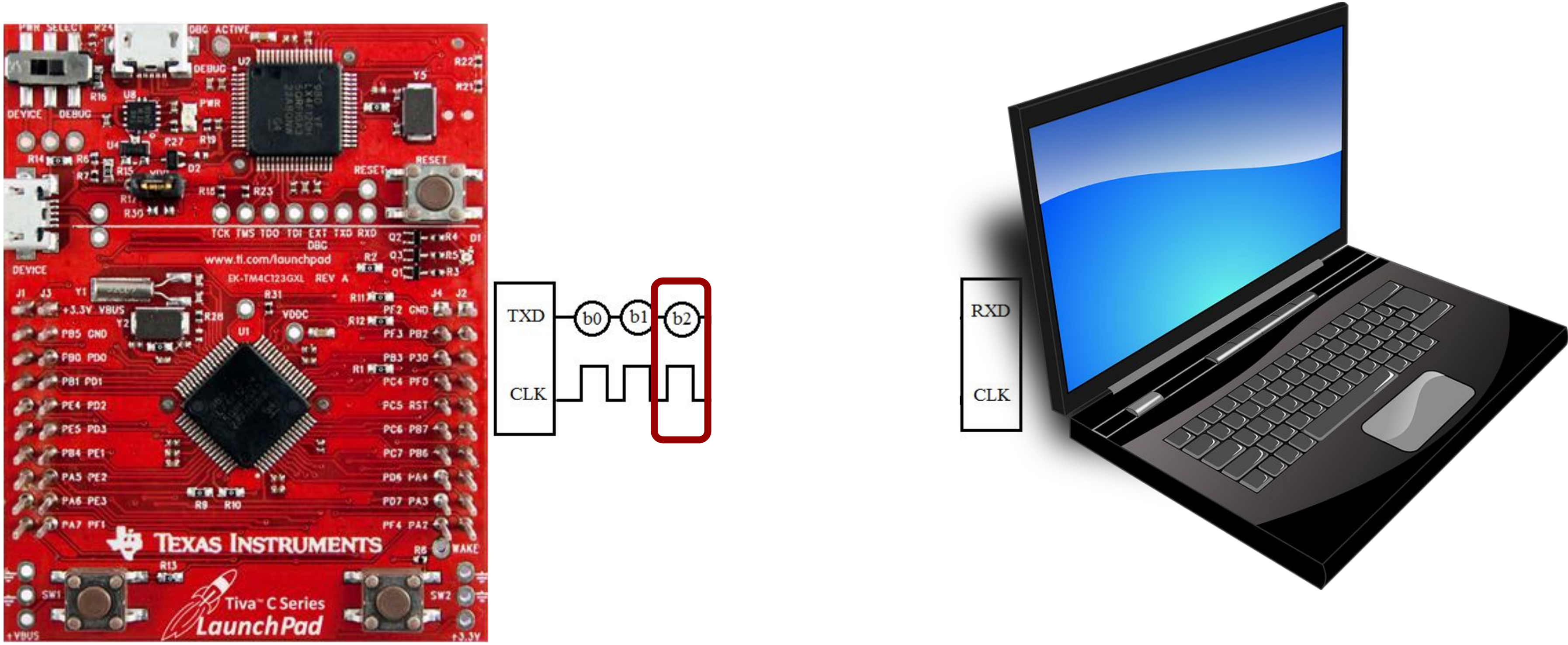
Seri Porttan iletisim



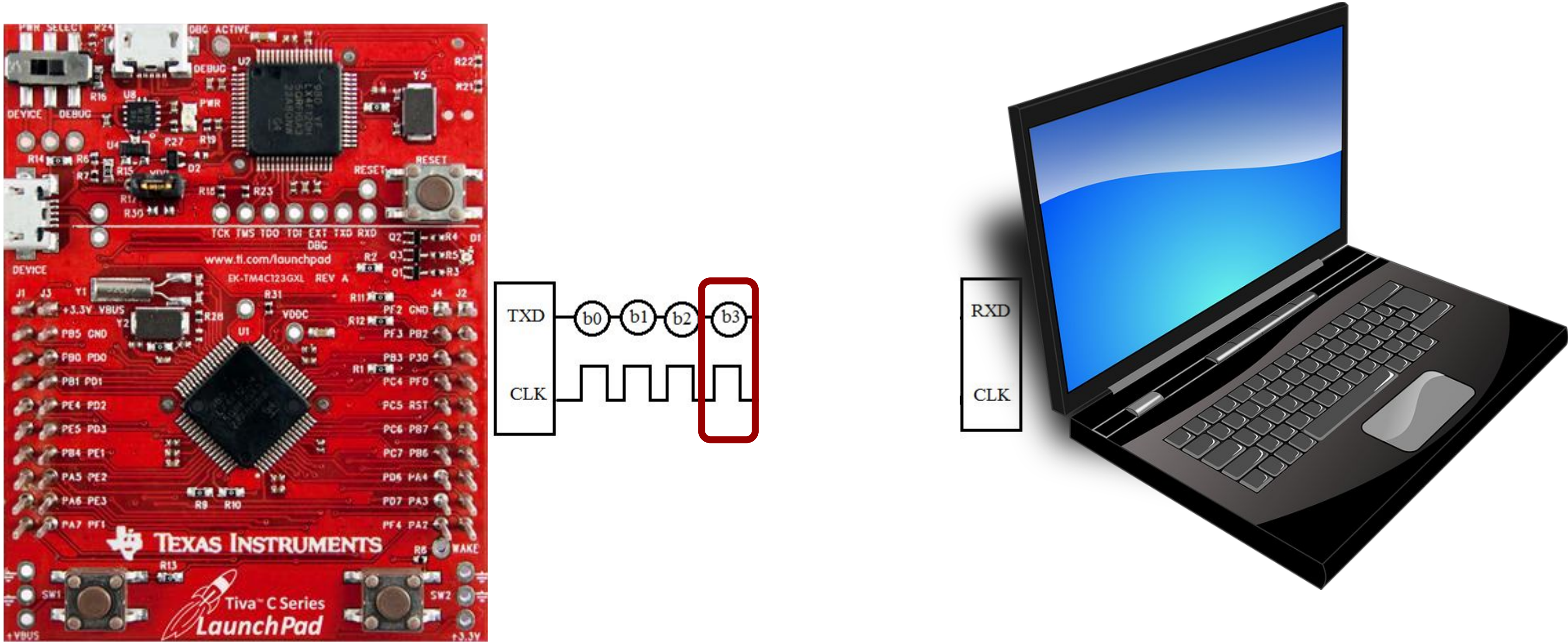
Seri Porttan iletisim



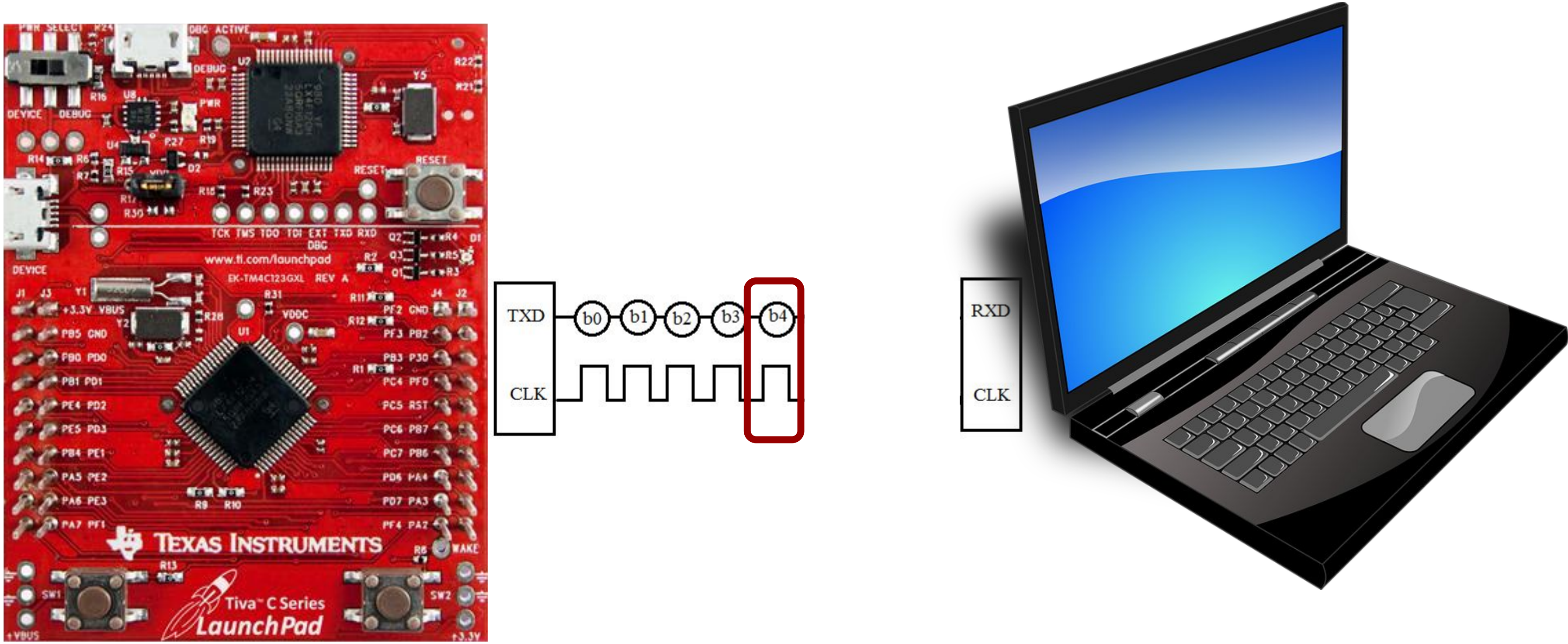
Seri Porttan iletisim



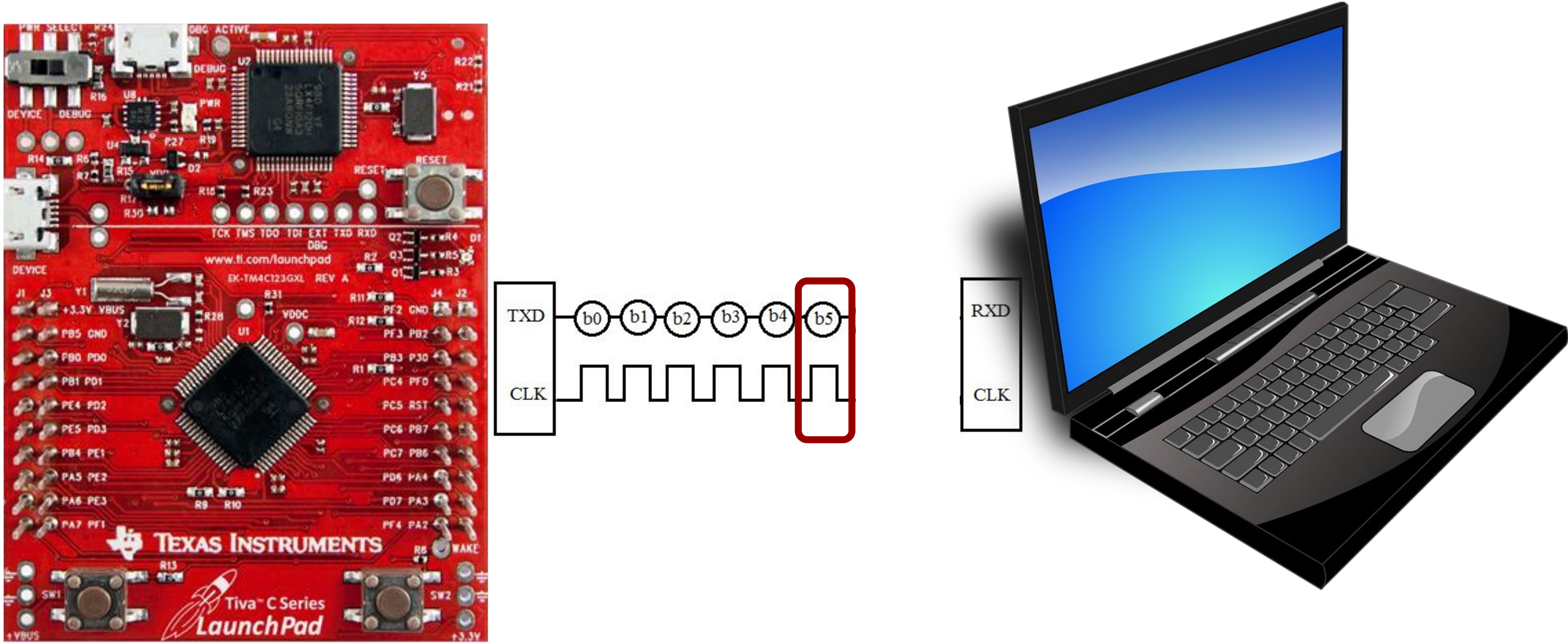
Seri Porttan iletisim



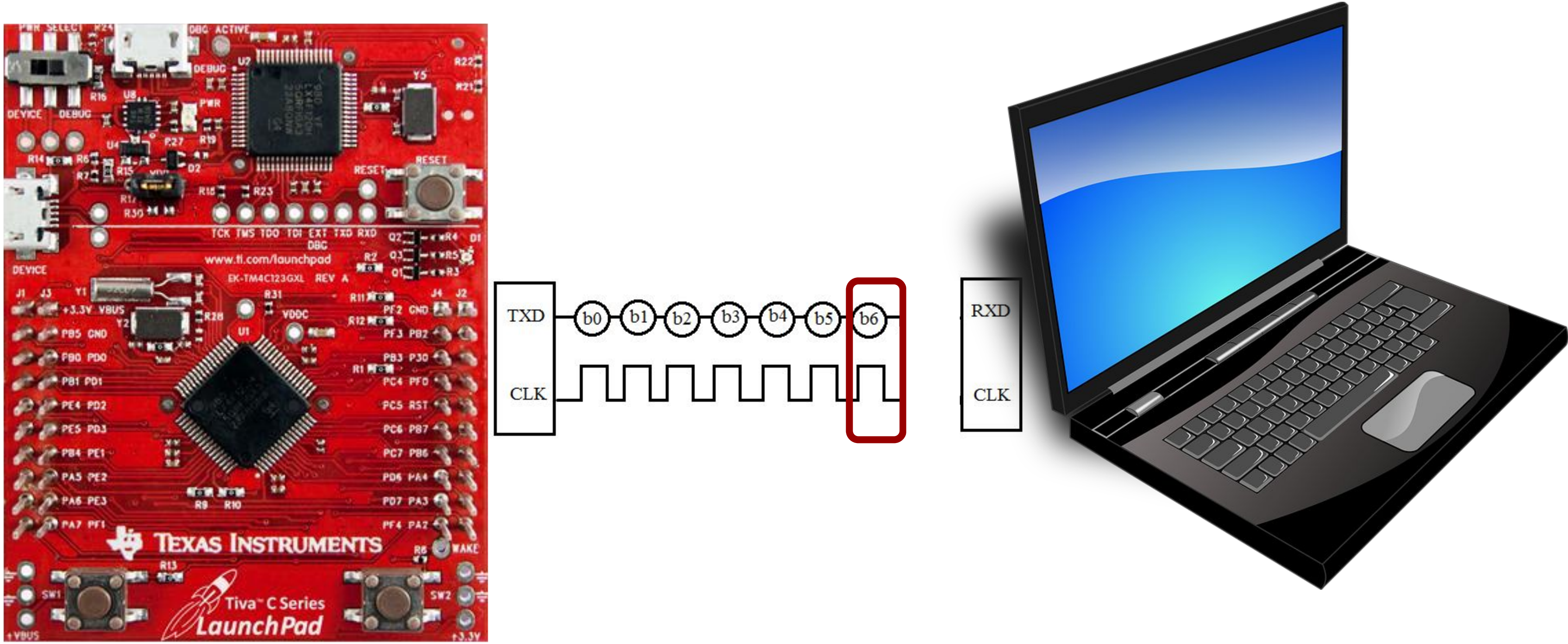
Seri Porttan iletisim



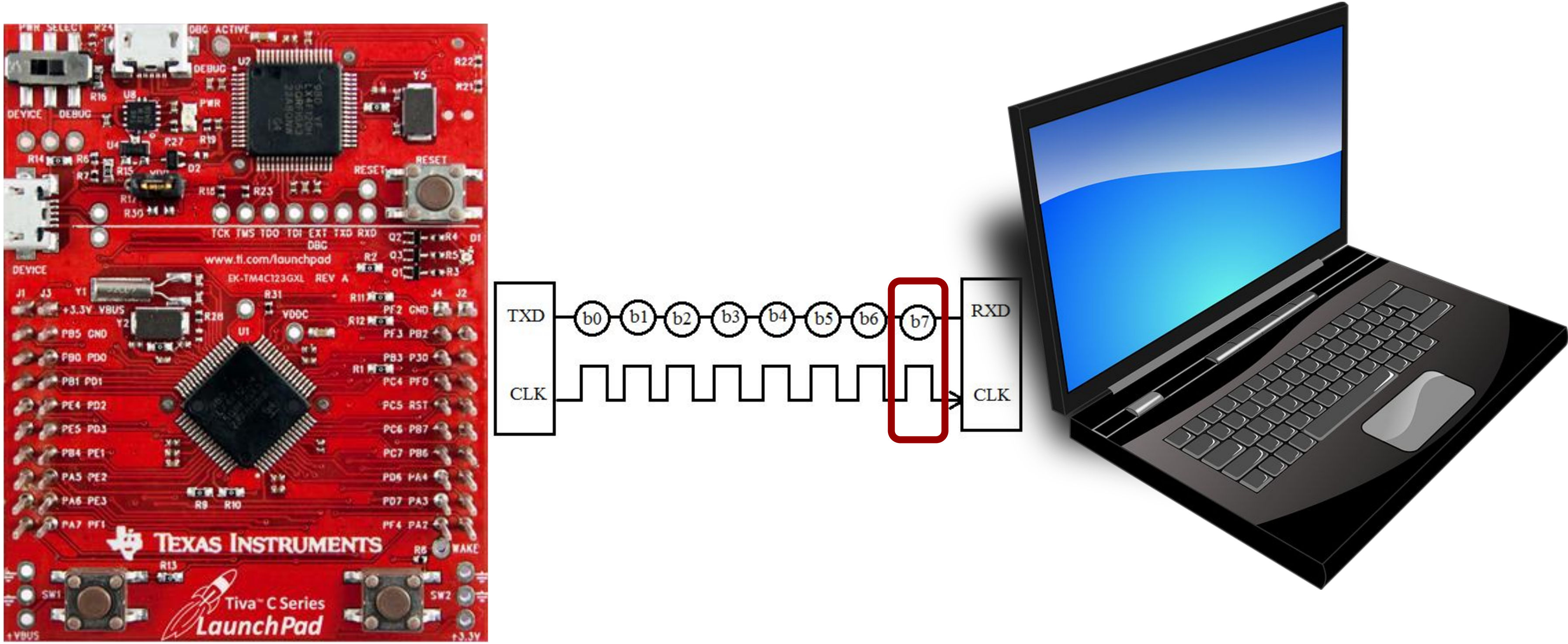
Seri Porttan iletisim



Seri Porttan iletisim



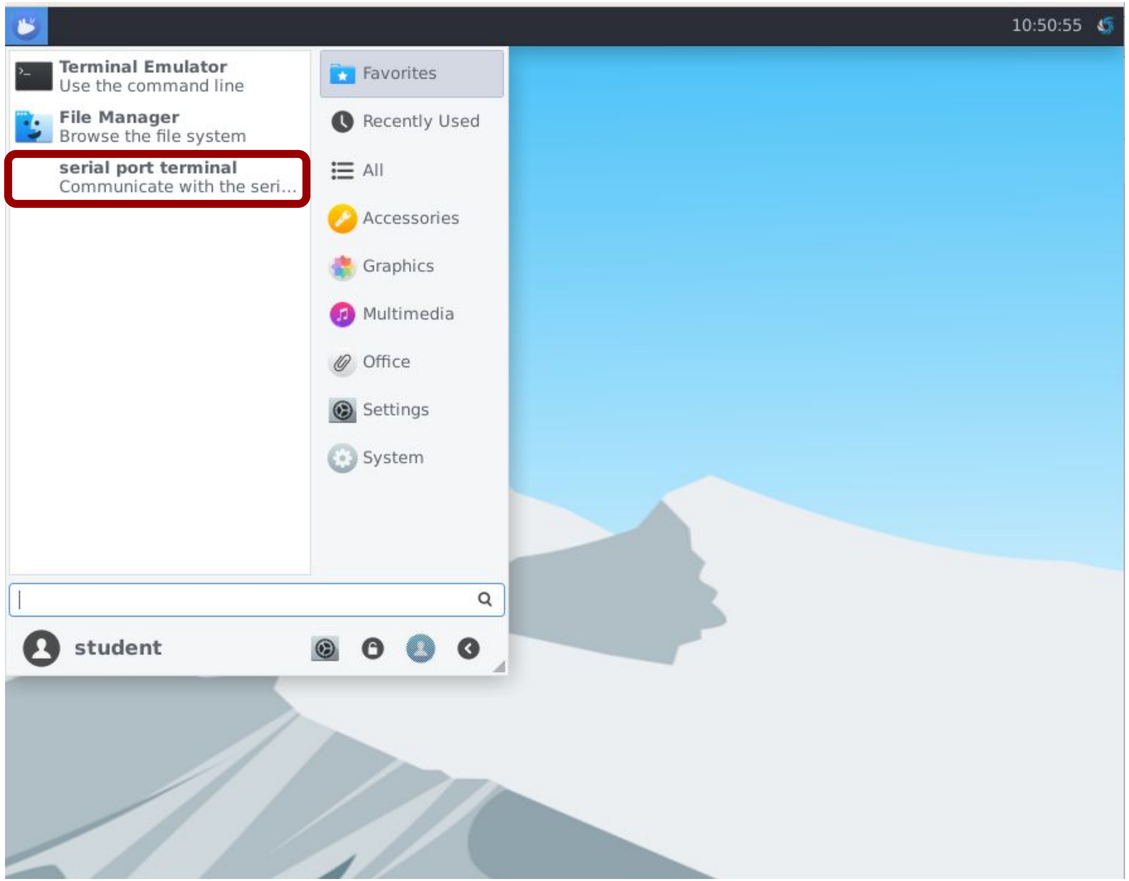
Seri Porttan iletisim



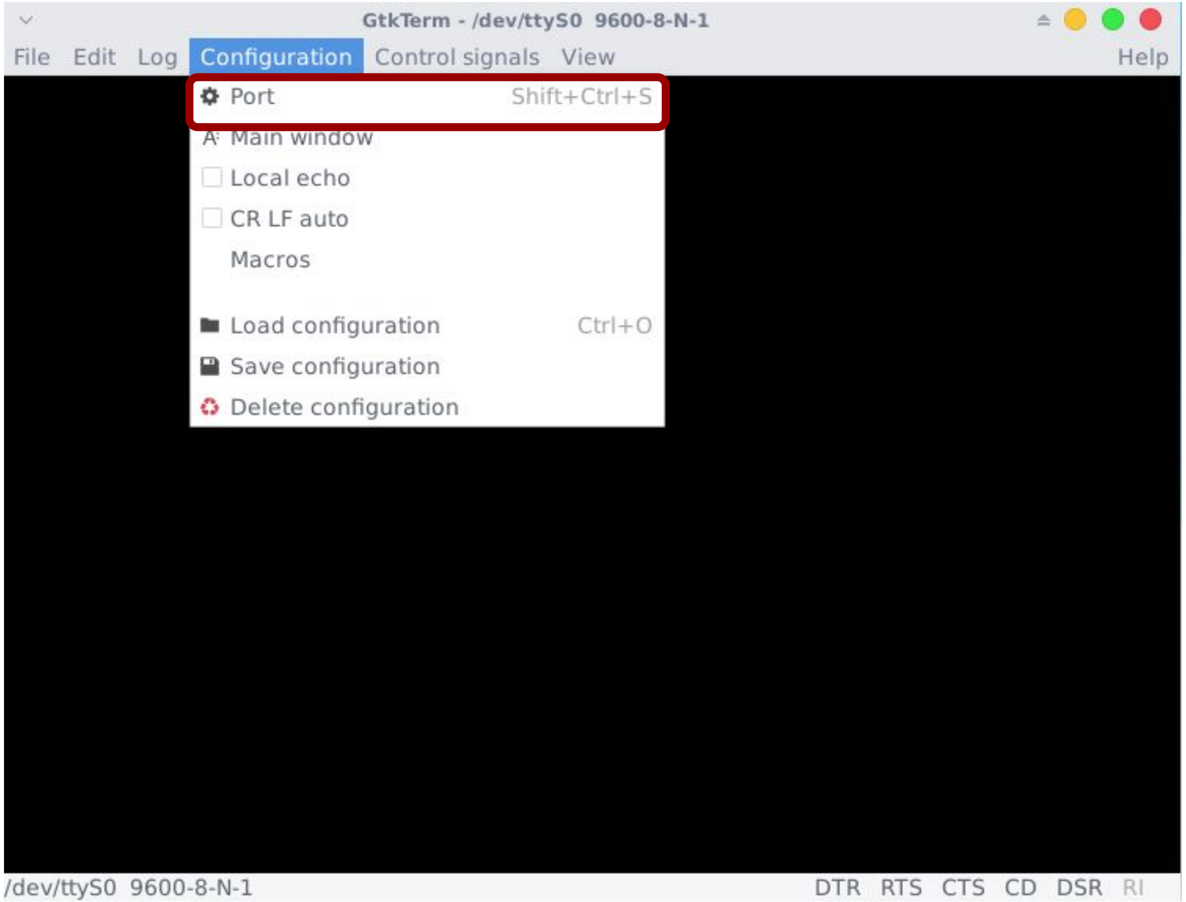
Seri Port Çalıstırma

```
int main(void) {
        int i;

        init_UARTstdio();

        for (i = 0 ; ; i++) {
         UARTprintf("test: %d\n", i);
        }
}
```

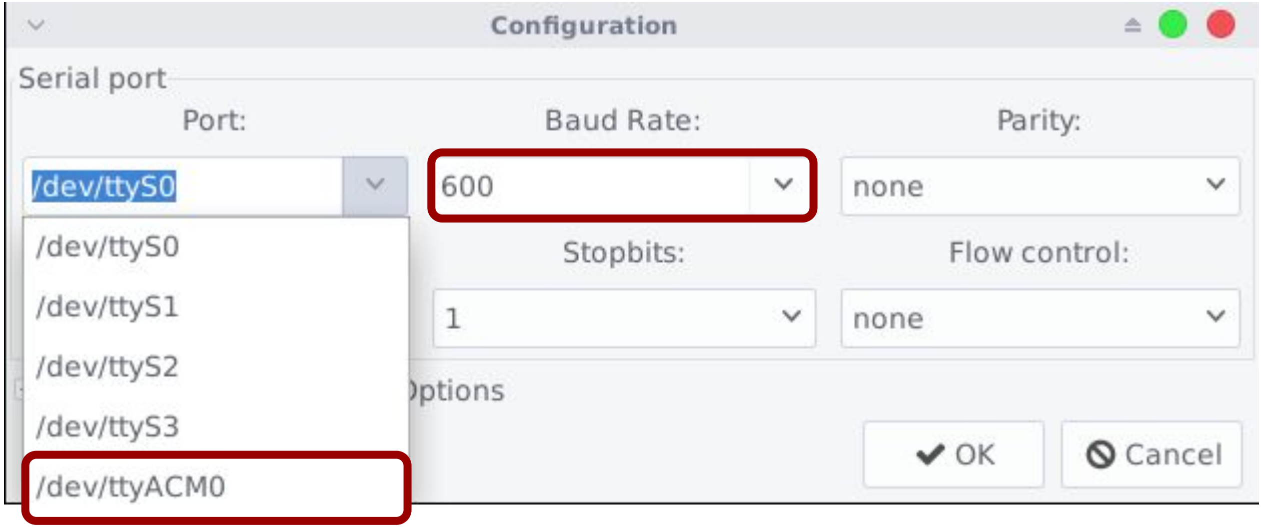
Seri Port Çalıstırma



Seri Port Çalıstırma



Seri Port Çalıstırma



Port: /dev/ttyS0

/dev/ttyS0
/dev/ttyS1
/dev/ttyS2
/dev/ttyS3
/dev/ttyACM0

Baud Rate: 600

Parity: none

Stopbits: 1

Flow control: none

Sorular