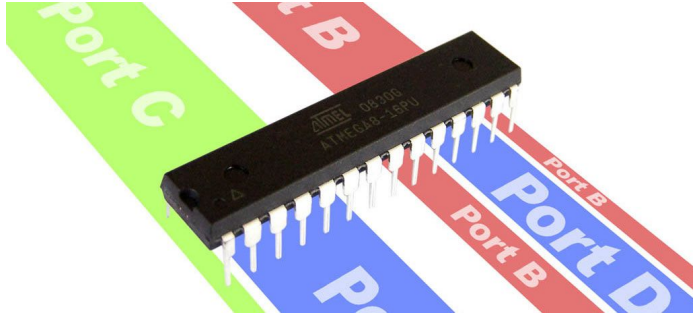


Mikrokontrolcu Portları



Suhap SAHIN

Gelistirme Ortamının Kurulumu

Sanal Makine Programı(VirtualBox)

Sanal Makine Dosyasının
Kurulumu(xubuntu-kouembedded-v20160217.ova)



VirtualBox

Virtualbox kurulumu

<https://www.virtualbox.org/>



The screenshot shows the Oracle VM VirtualBox website. The browser window has a single tab titled 'Oracle VM VirtualBo x'. The address bar shows 'Secure https://www.virtualbox.org/'. Below the address bar, there are social media links for Apps, OLASILIK NEDİR?, CS109-1, Vector intro for li, I3: I3 User's Guide, Data Structure an, and Source code bea. The main content area features the VirtualBox logo on the left, a large 'VirtualBox' title, and a 'Welcome to VirtualBox.org!' message. To the right of the welcome message is a 'News Flash' section with three items: 'New January 15th, 2018 VirtualBox 5.2.6 released!', 'New January 15th, 2018 VirtualBox 5.1.32 released!', and 'New October 18th, 2017 VirtualBox 5.2 released!'. At the bottom of the page, there is a large blue button with the text 'Download VirtualBox 5.2' and a red border. Below the button, the text 'Hot picks:' is visible.

Oracle VM VirtualBo x

Secure https://www.virtualbox.org/

Apps OLASILIK NEDİR? CS109-1 Vector intro for li I3: I3 User's Guide Data Structure an Source code bea

VirtualBox

Welcome to VirtualBox.org!

VirtualBox is a powerful x86 and AMD64/Intel64 virtualization product for enterprise as well as home use. Not only is VirtualBox an extremely feature rich, high performance product for enterprise customers, it is also the only professional solution that is freely available as Open Source Software under the terms of the GNU General Public License (GPL) version 2. See "About VirtualBox" for an introduction.

Presently, VirtualBox runs on Windows, Linux, Macintosh, and Solaris hosts and supports a large number of guest operating systems including but not limited to Windows (NT 4.0, 2000, XP, Server 2003, Vista, Windows 7, Windows 8, Windows 10), DOS/Windows 3.x, Linux (2.4, 2.6, 3.x and 4.x), Solaris and OpenSolaris, OS/2, and OpenBSD.

VirtualBox is being actively developed with frequent releases and has an ever growing list of features, supported guest operating systems and platforms it runs on. VirtualBox is a community effort backed by a dedicated company: everyone is encouraged to contribute while Oracle ensures the product always meets professional quality criteria.

Download VirtualBox 5.2

Hot picks:

News Flash

- New January 15th, 2018 VirtualBox 5.2.6 released!**
Oracle today released a 5.2 maintenance release which improves stability and fixes regressions. See the [Changelog](#) for details.
- New January 15th, 2018 VirtualBox 5.1.32 released!**
Oracle today released a 5.1 maintenance release which improves stability and fixes regressions. See the [Changelog](#) for details.
- New October 18th, 2017 VirtualBox 5.2 released!**
Oracle today shipped a new minor release, VirtualBox 5.2. See the [announcement](#) for details.

[More information...](#)

Virtualbox kurulumu

<https://www.virtualbox.org/wiki/Downloads>

A screenshot of a web browser displaying the VirtualBox Downloads page. The browser's address bar shows the URL 'https://www.virtualbox.org/wiki/Downloads'. The page features the VirtualBox logo on the left and a large 'VirtualBox' title in the center. Below the title, there is a section titled 'Download VirtualBox' with a sub-section 'VirtualBox binaries'. This section contains two main bullet points: 'VirtualBox 5.2.8 platform packages' and 'VirtualBox 5.2.8 Oracle VM VirtualBox Extension Pack'. The first bullet point is highlighted with a red border, and the second is highlighted with an orange border. The page also includes a sidebar with links to 'About', 'Screenshots', 'Downloads', 'Documentation', 'End-user docs', 'Technical docs', 'Contribute', and 'Community'.

Downloads - Oracle x

Secure | <https://www.virtualbox.org/wiki/Downloads>

Apps OLASILIK NEDİR? CS109-1 Vector intro for I3: i3 User's Guide Data Structure an

 **VirtualBox** [Login](#) [Preferences](#)

Download VirtualBox

Here, you will find links to VirtualBox binaries and its source code.

VirtualBox binaries

By downloading, you agree to the terms and conditions of the respective license.

If you're looking for the VirtualBox 5.1.34 packages, see [VirtualBox 5.1 builds](#). Consider upgrading.

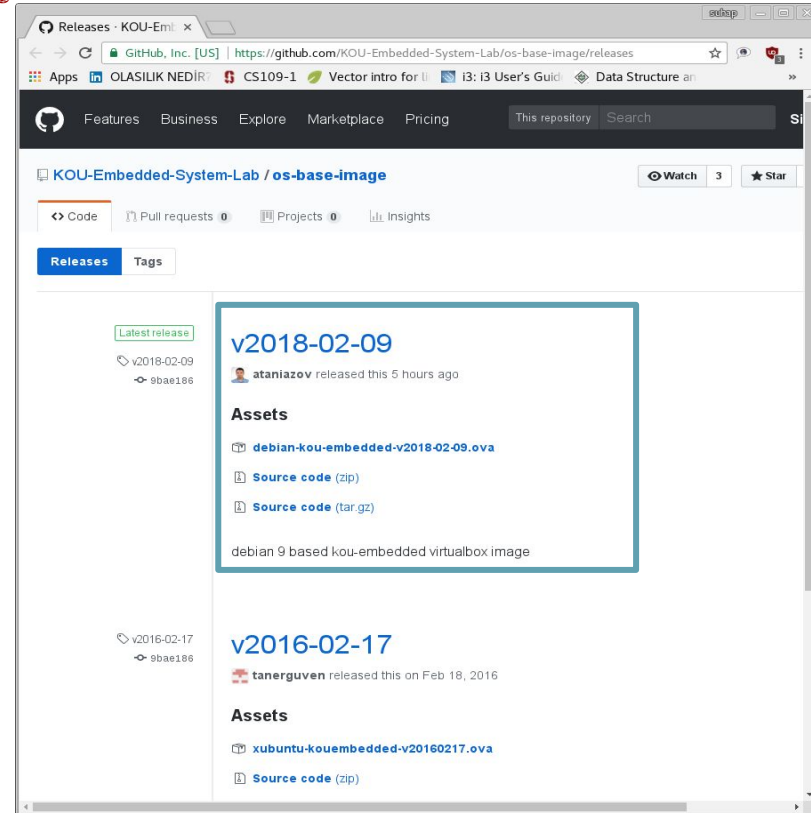
- **VirtualBox 5.2.8 platform packages.** The binaries are released under the terms of the GPL version 2.
 - [Windows hosts](#)
 - [OS X hosts](#)
 - [Linux distributions](#)
 - [Solaris hosts](#)
- **VirtualBox 5.2.8 Oracle VM VirtualBox Extension Pack** [All supported platforms](#)
Support for USB 2.0 and USB 3.0 devices, VirtualBox RDP, disk encryption, NVMe and PXE boot for Intel cards. See [this chapter from the User Manual](#) for an introduction to this Extension Pack.
The Extension Pack binaries are released under the [VirtualBox Personal Use and Evaluation License \(PUEL\)](#).
Please install the extension pack with the same version as your installed version of VirtualBox.

Sanal Makina Dosyası

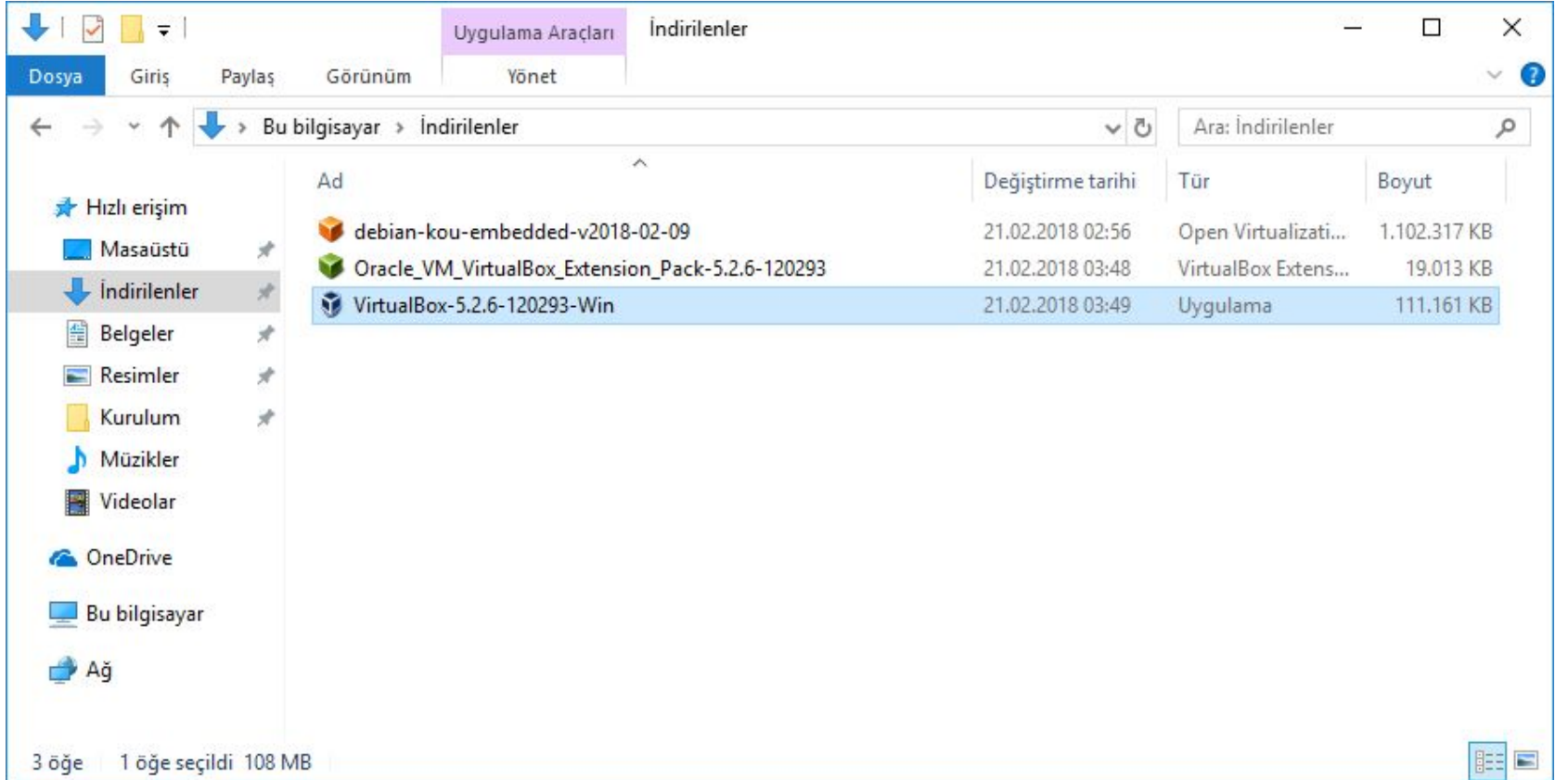
<https://github.com/KOU-Embedded-System-Lab/os-base-image/releases/tag/v2018-02-09>

<https://github.com/KOU-Embedded-System-Lab/os-base-image/releases>

debian-kou-embedded-v2018-02-09.ova



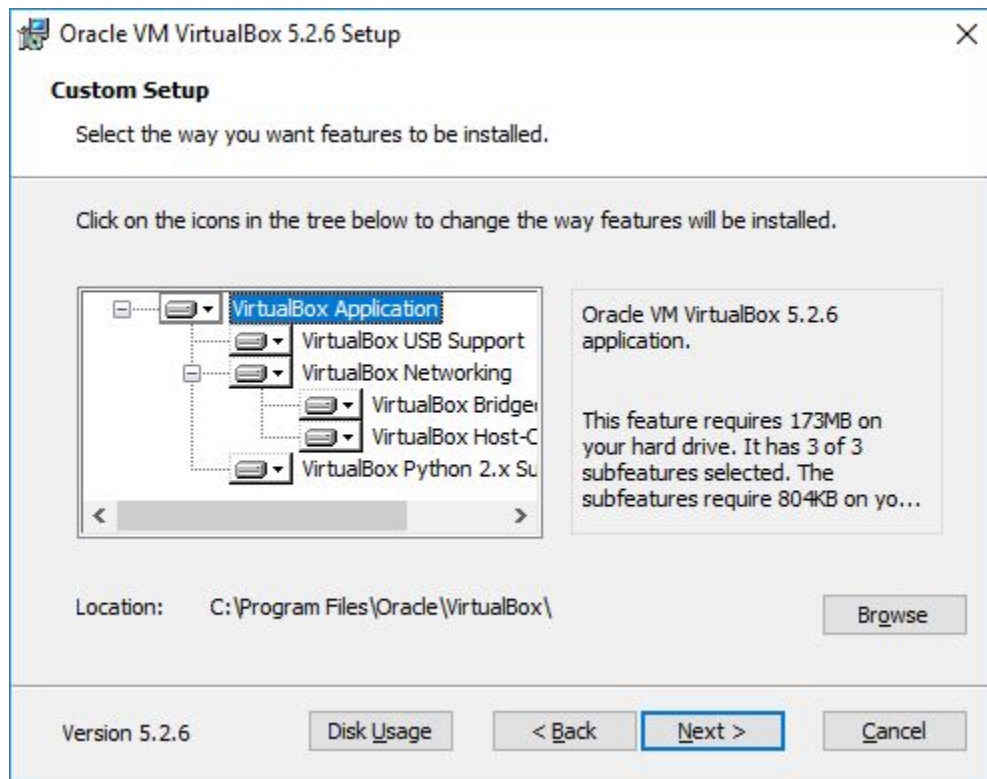
Sanal Makina Kurulumu



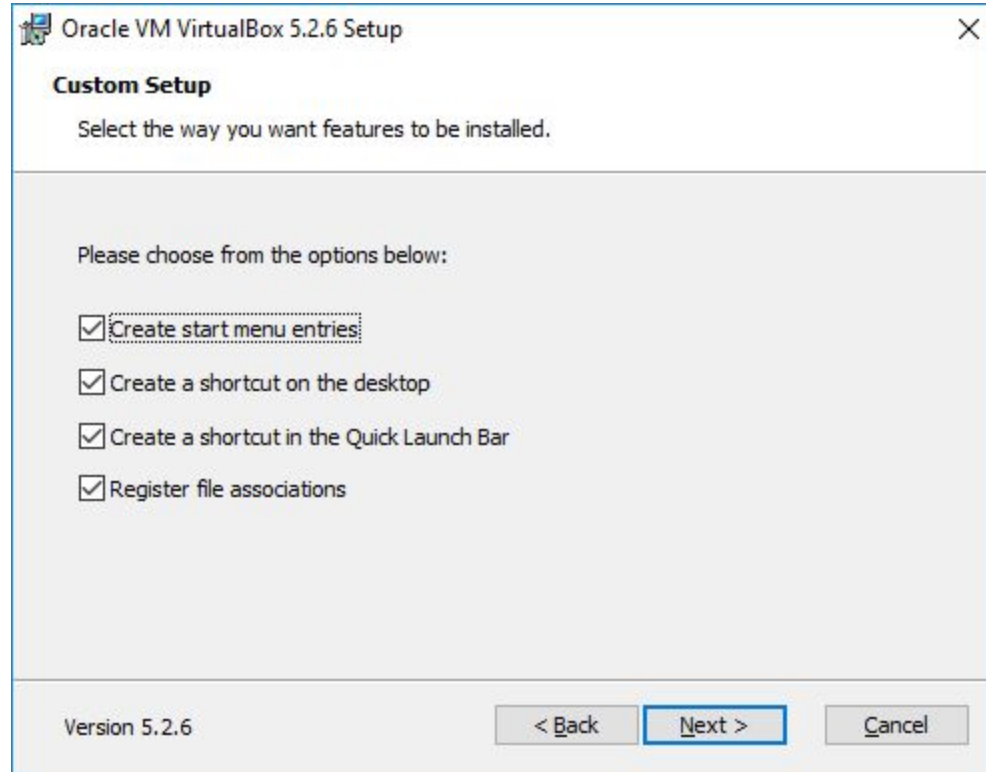
Sanal Makina Kurulumu



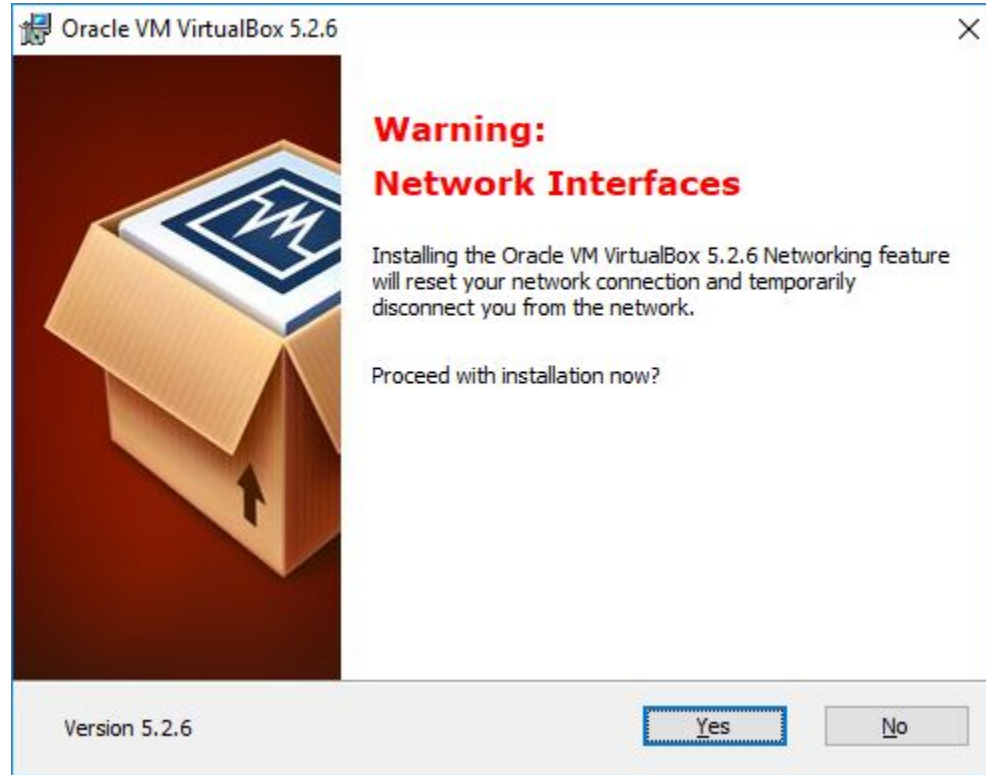
Sanal Makina Kurulumu



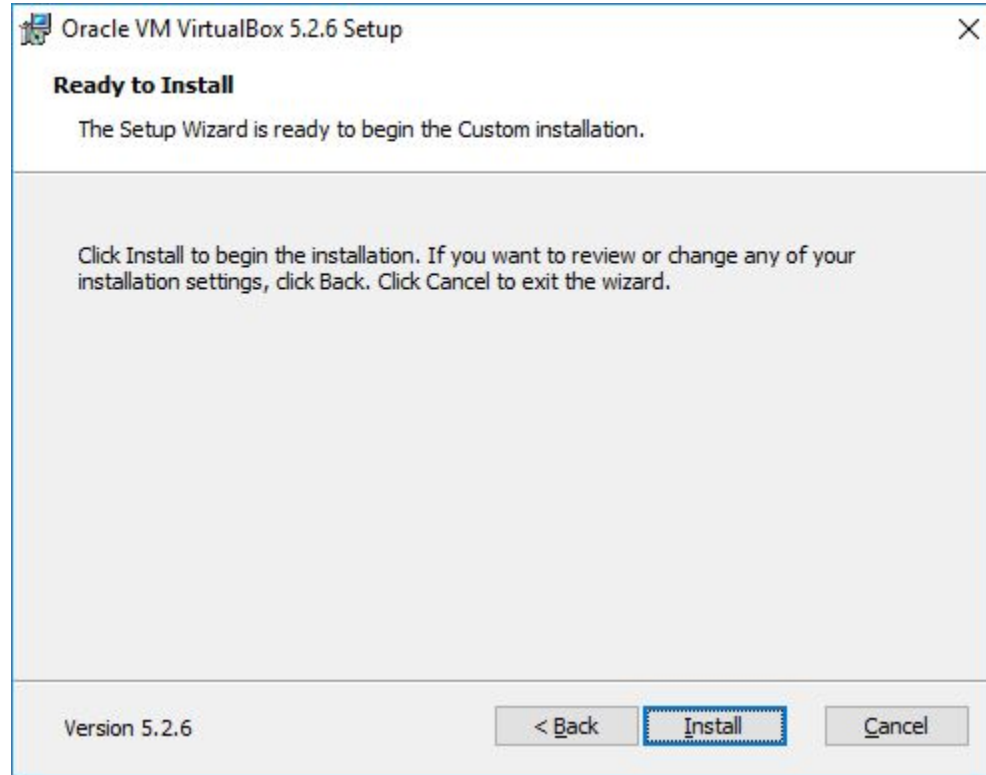
Sanal Makina Kurulumu



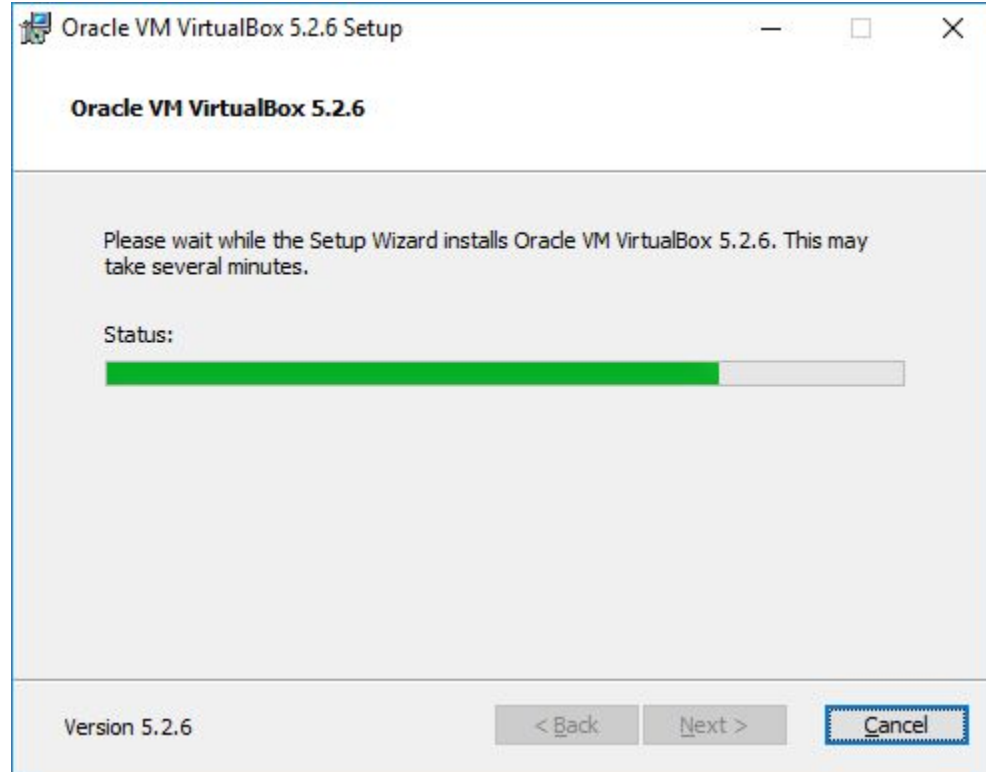
Sanal Makina Kurulumu



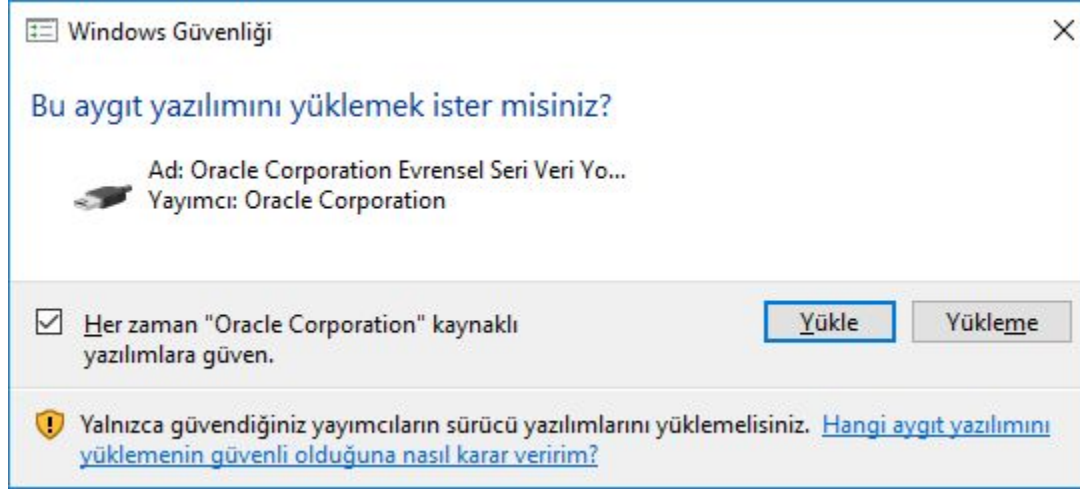
Sanal Makina Kurulumu



Sanal Makina Kurulumu



Sanal Makina Kurulumu



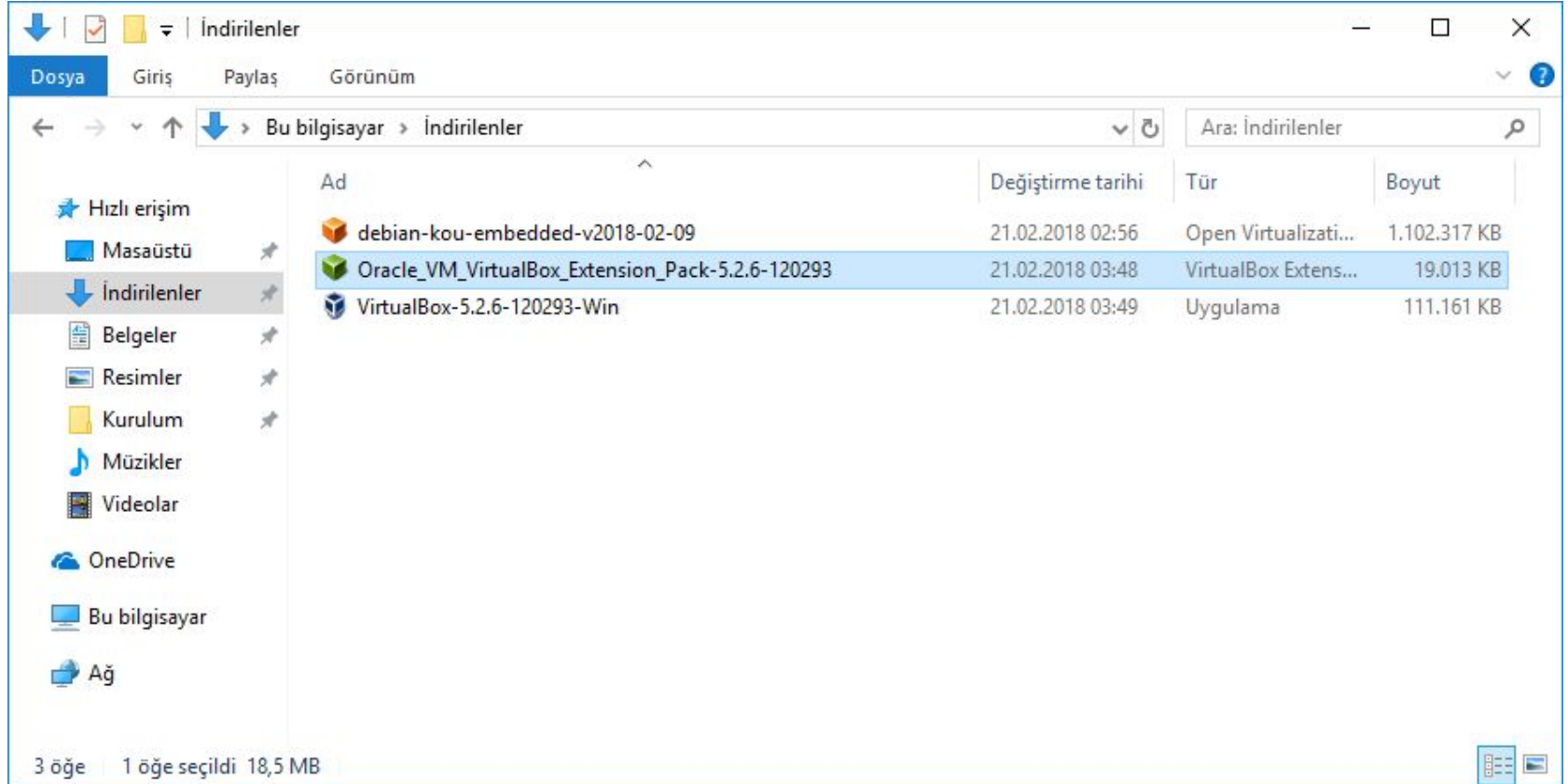
Sanal Makina Kurulumu



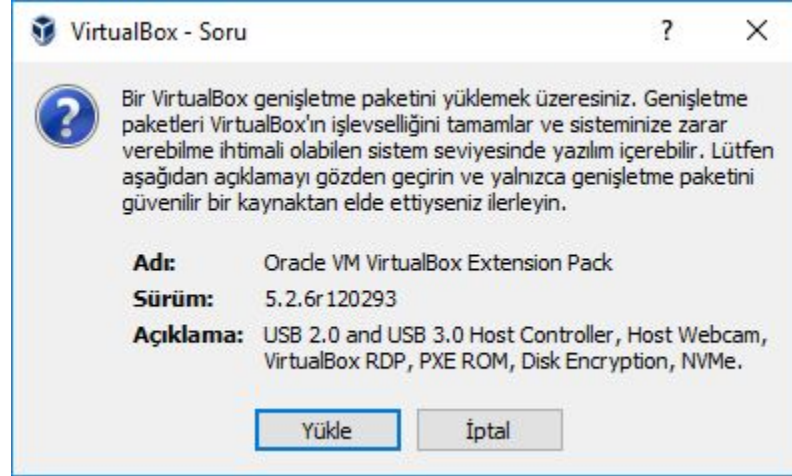
Sanal Makina Kurulumu



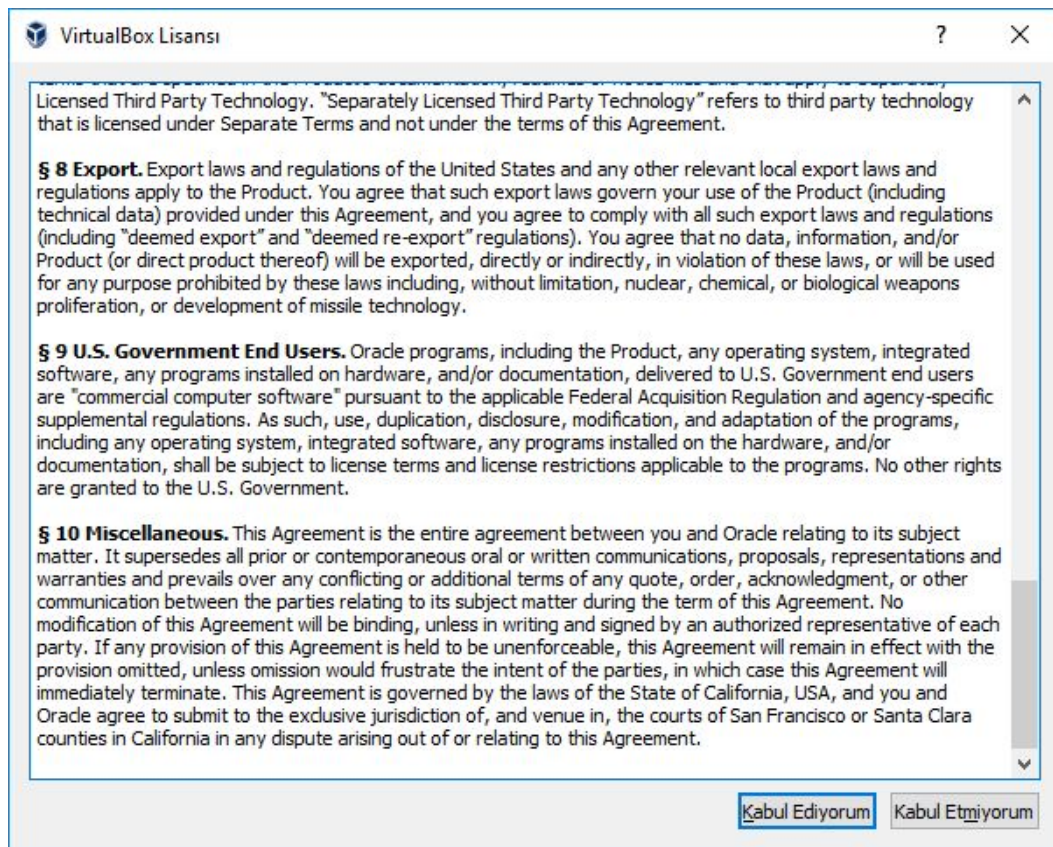
Sanal Makina Kurulumu



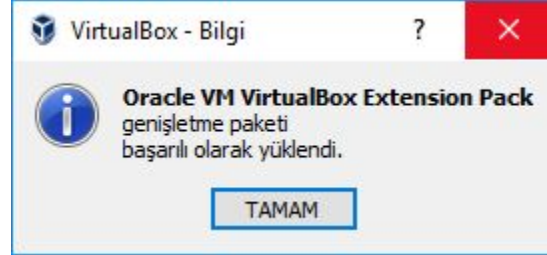
Sanal Makina Kurulumu



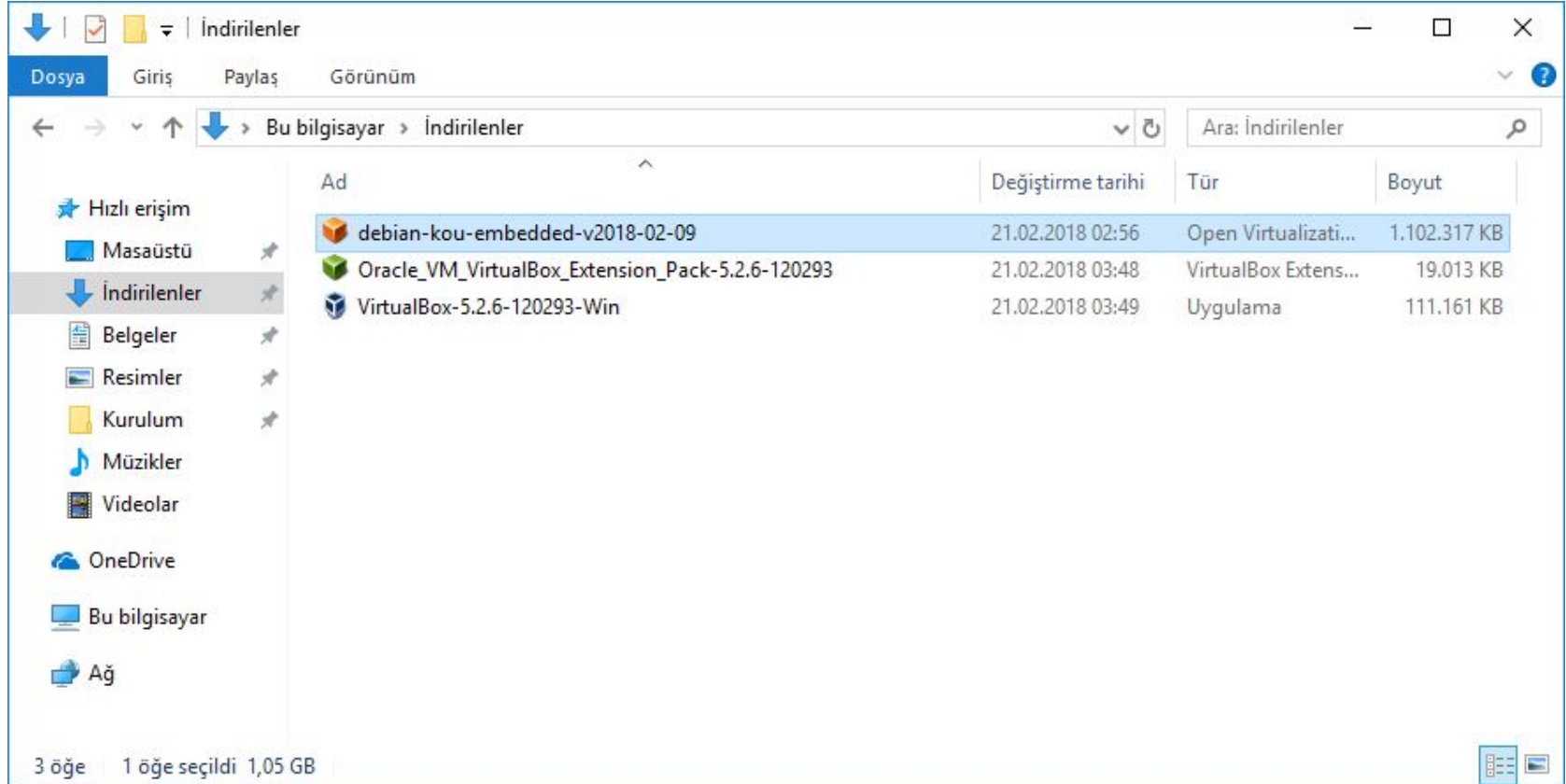
Sanal Makina Kurulumu



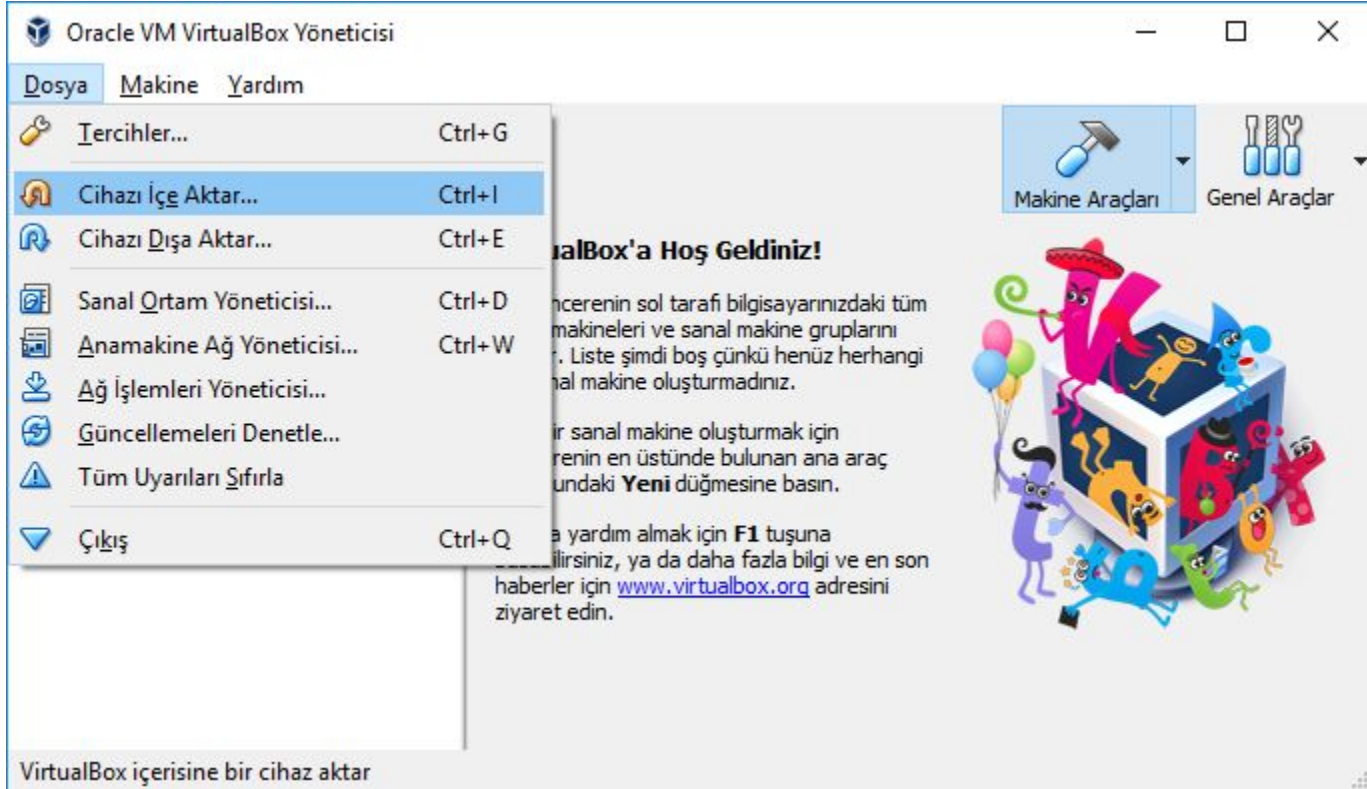
Sanal Makina Kurulumu



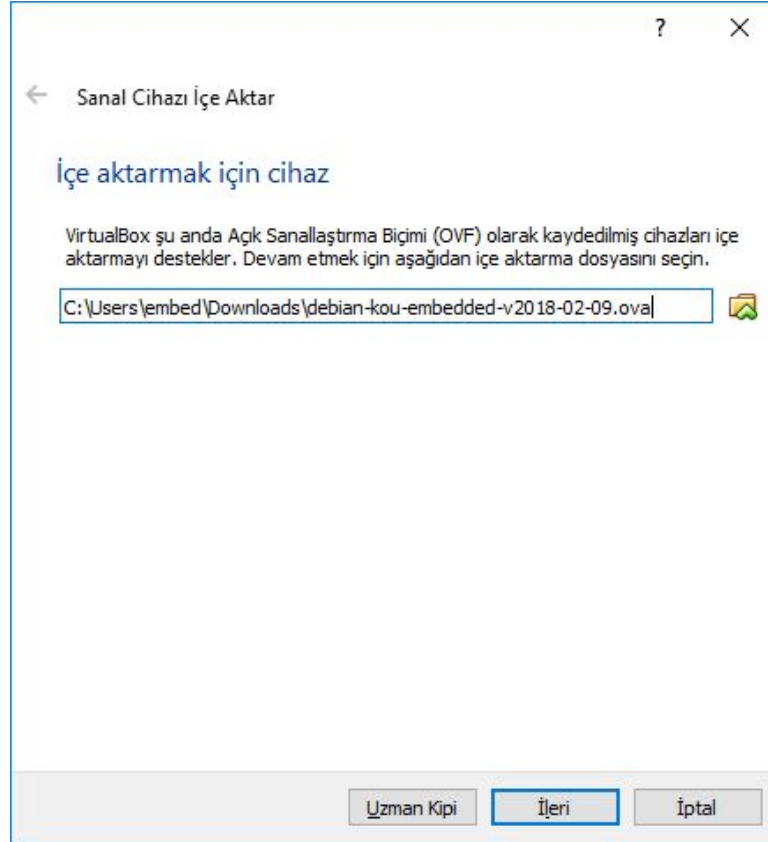
Sanal Makina Kurulumu



Sanal Makina Kurulumu



Sanal Makina Kurulumu











Sanal Makina Kurulumu

? ×

← Sanal Cihazı İçe Aktar

Cihaz ayarları

Bunlar cihaz içinde bulunan sanal makineler ve içe aktarılmış VirtualBox makinelerinin önerilen ayarlarıdır. Ögelere çift tıkladığında gösterilen çoğu özellikleri değiştirebilirsiniz ve aşağıdan işaretleme kutularını kullanarak diğerlerini etkisizleştirebilirsiniz.

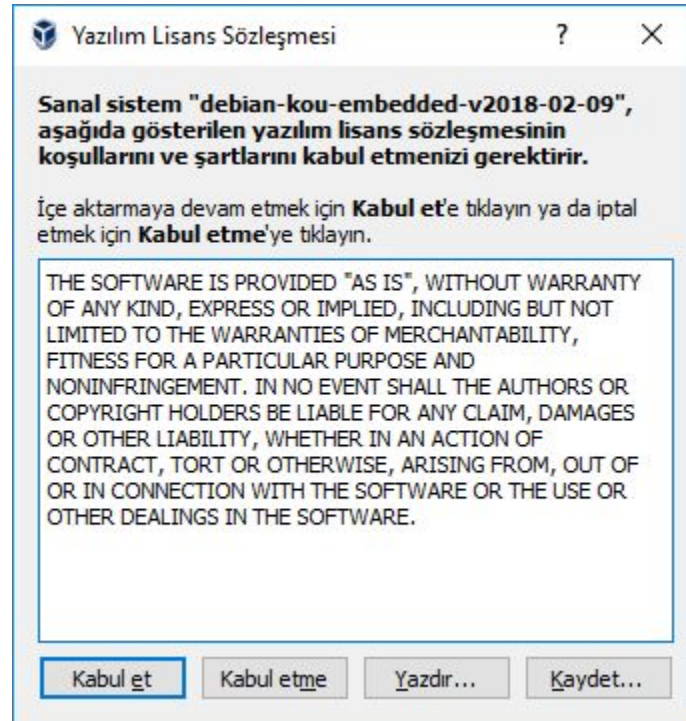
Sanal Sistem 1	
 Adı	debian-kou-embedded-...
 Sürüm	v2018-02-09
 Açıklama	debian-kou-embedded-...
 Misafir İS Türü	 Debian (64-bit)
 İşlemci	1
 Bellek	1536 MB
 DVD	<input checked="" type="checkbox"/>

☐ Tüm ağ kartlarının MAC adresini yeniden başlat

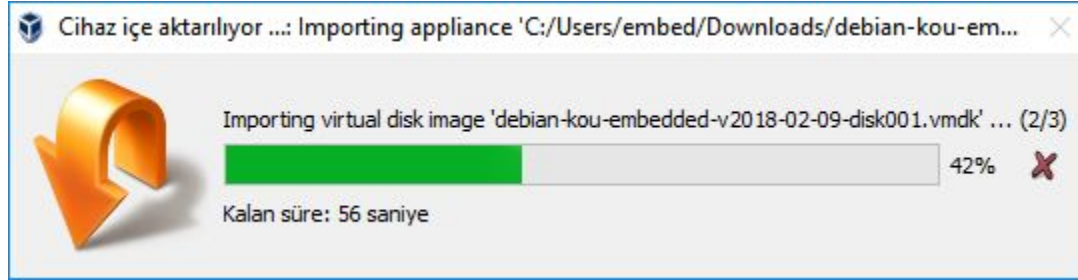
Cihaz imzalı değil

Varsayılanları Geri Yükle İçe Aktar İptal

Sanal Makina Kurulumu



Sanal Makina Kurulumu



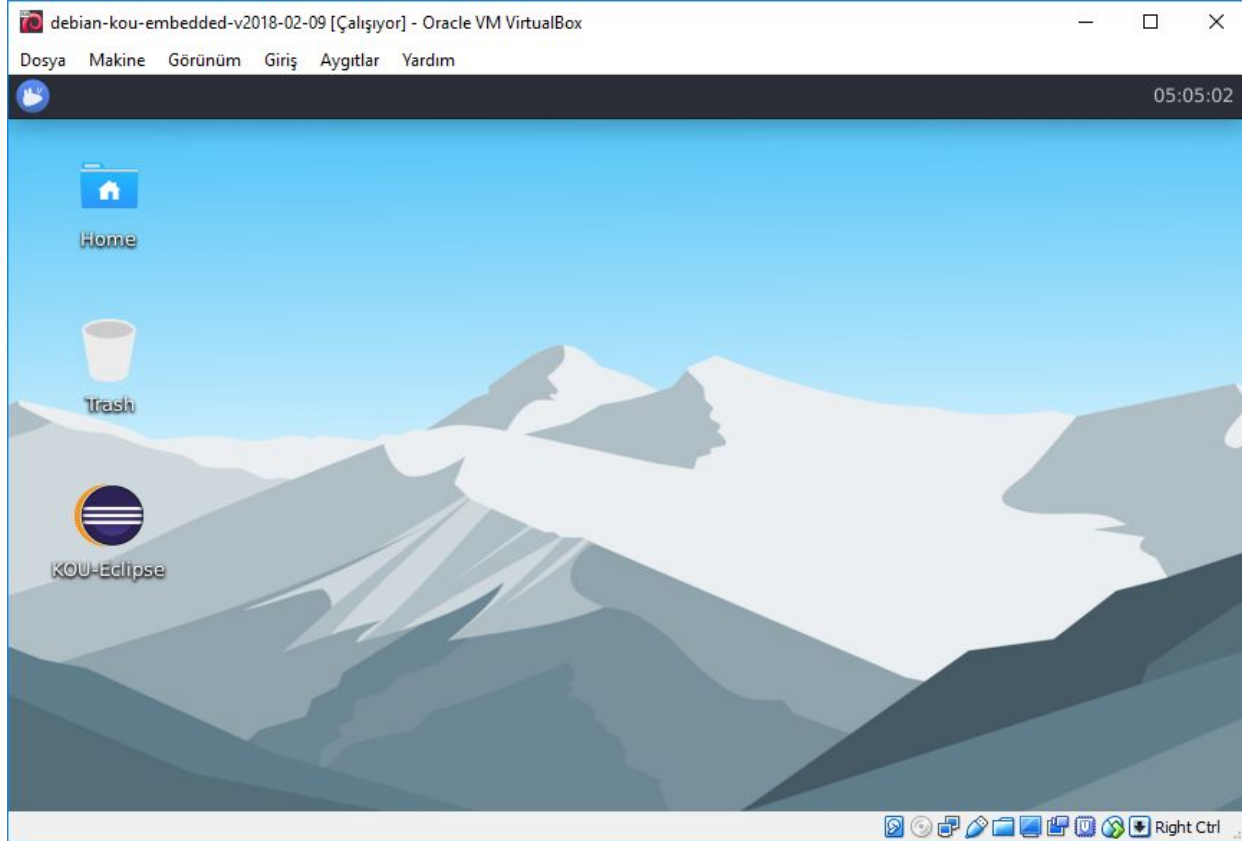
Sanal Makina Kurulumu



Sanal Makina Baslatma



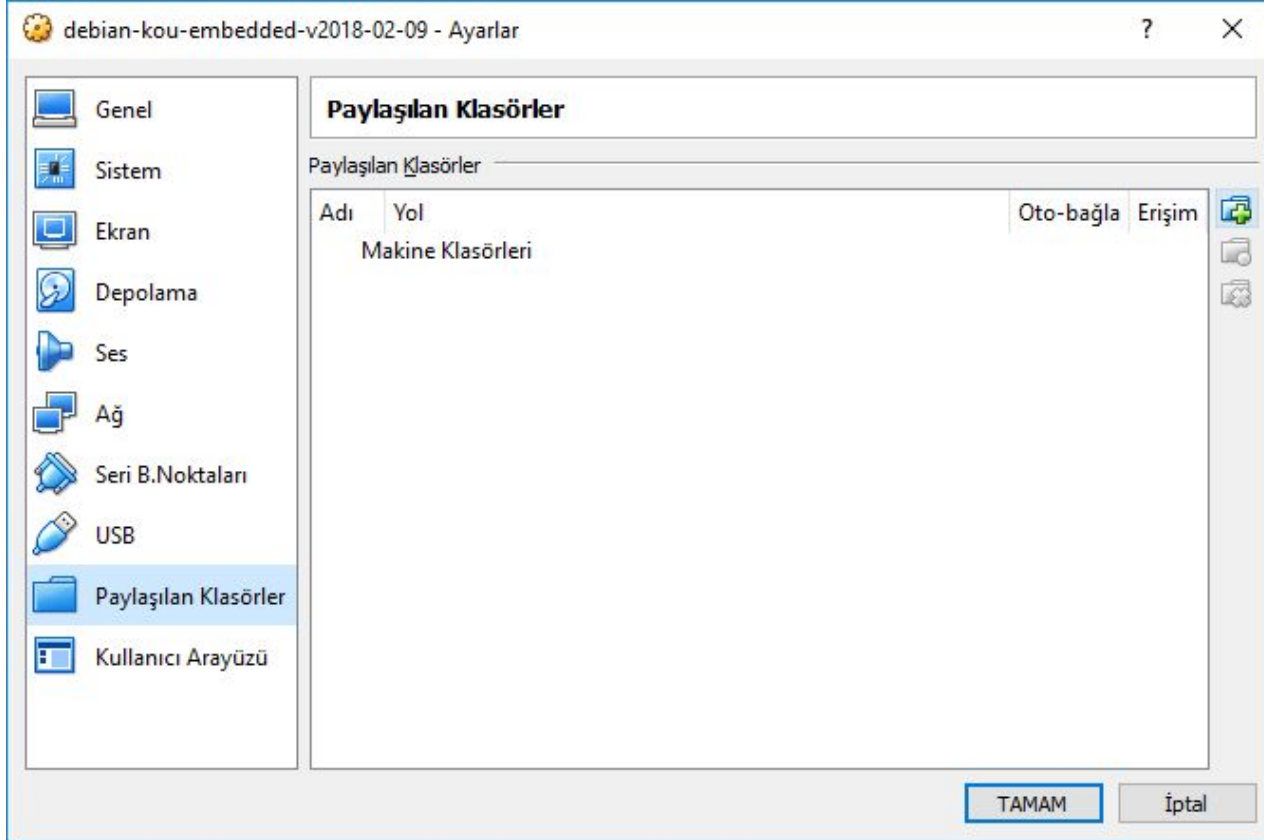
Sanal Makina Baslatma



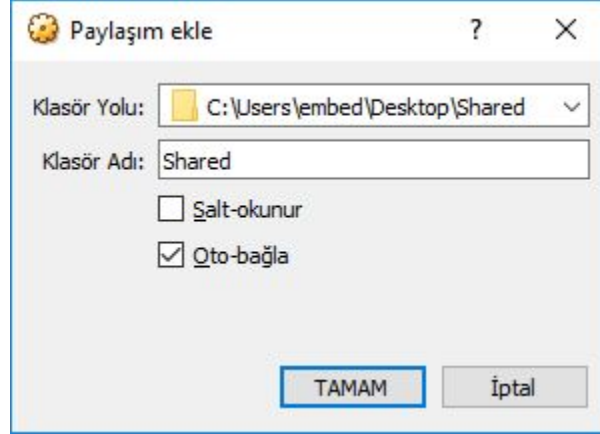
Sanal Makina Ayarlanması



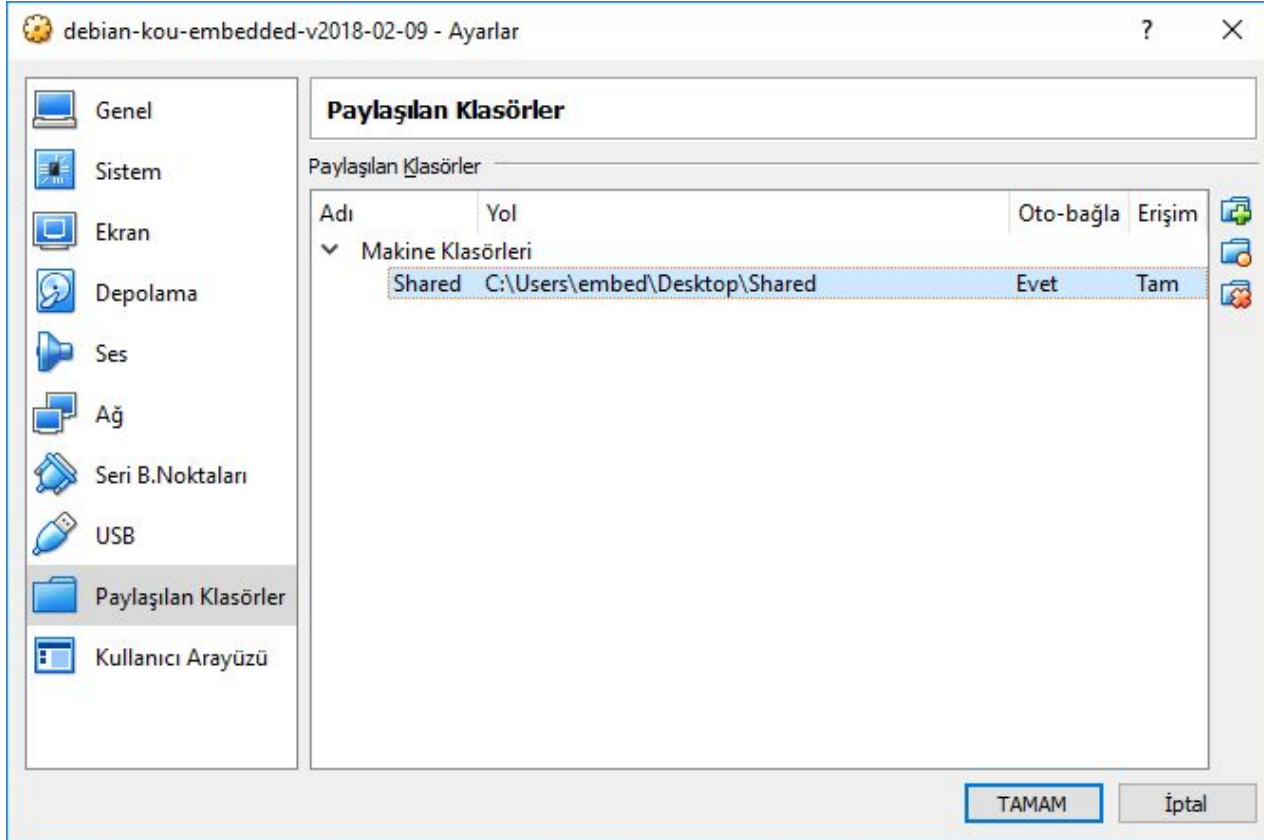
Sanal Makina Ayarlanması



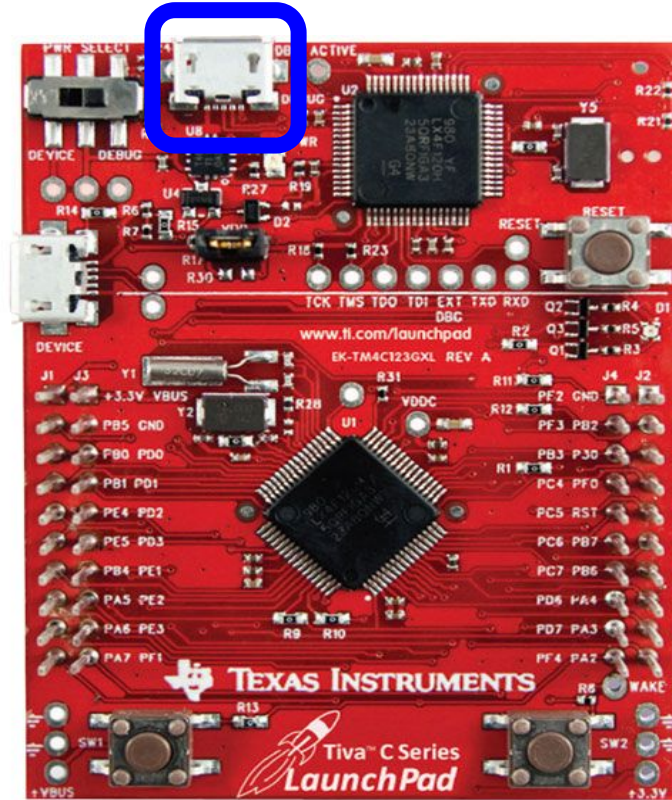
Sanal Makina Ayarlanması



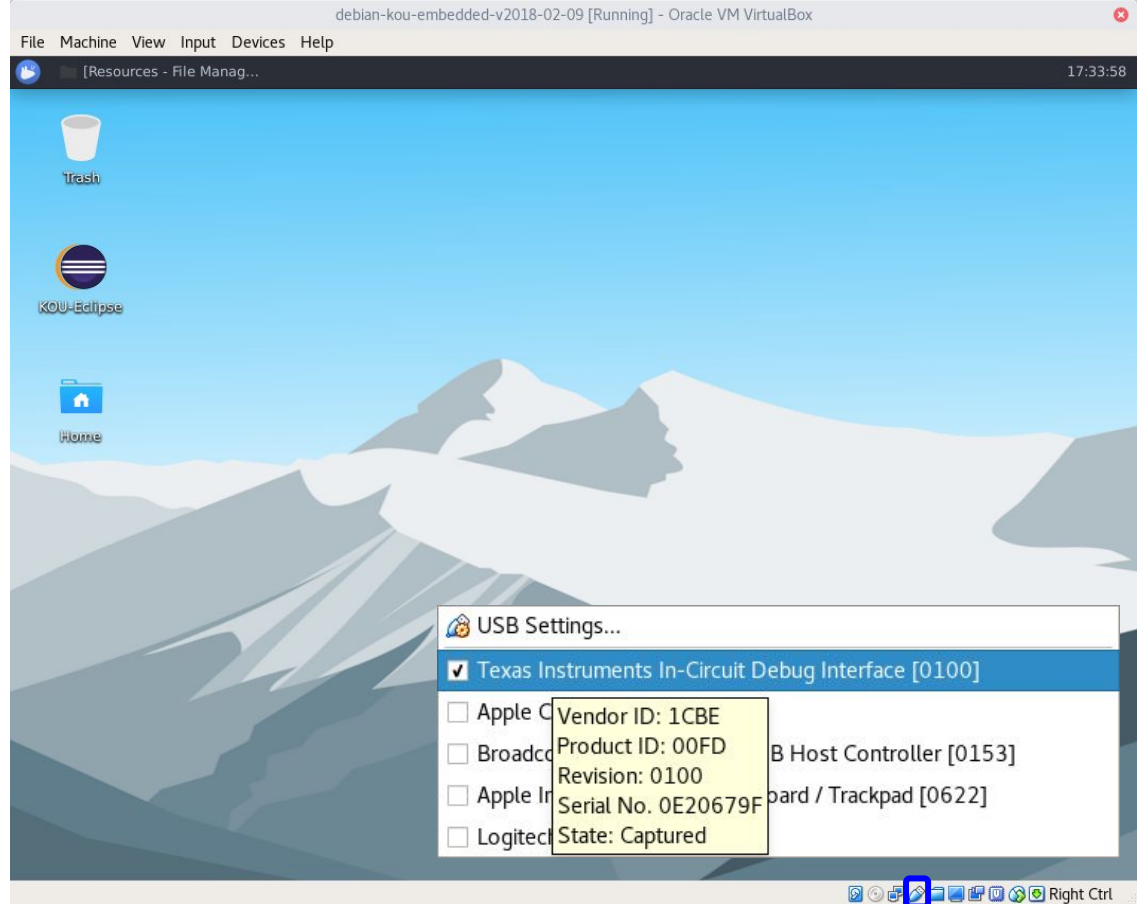
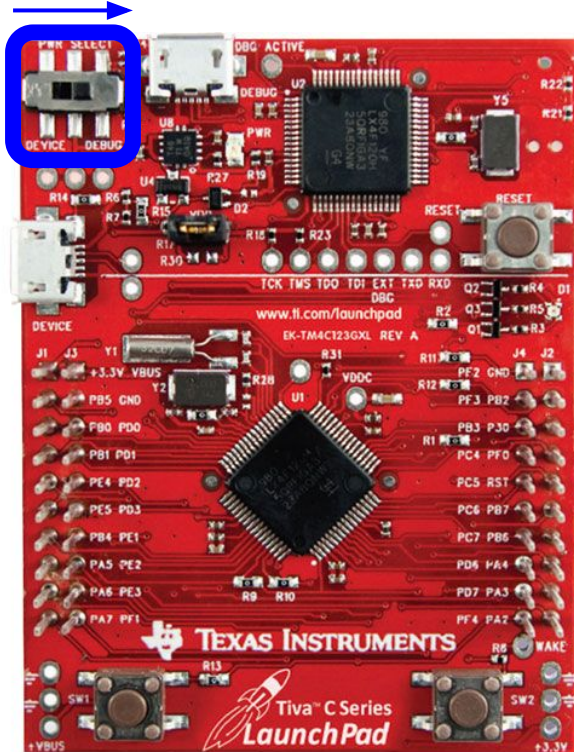
Sanal Makina Ayarlanması



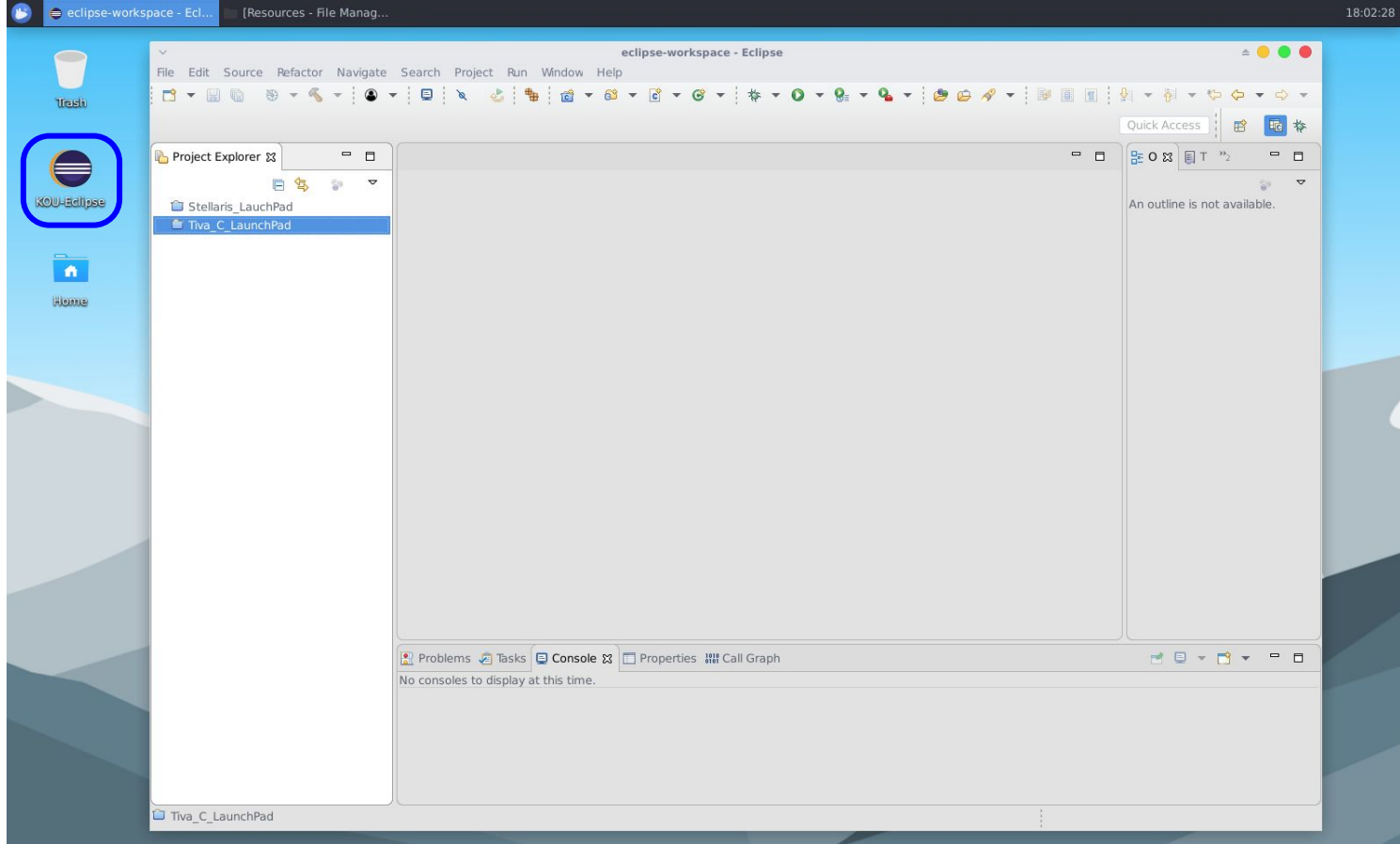
Kartın Sanal Makinaya Aktarılması



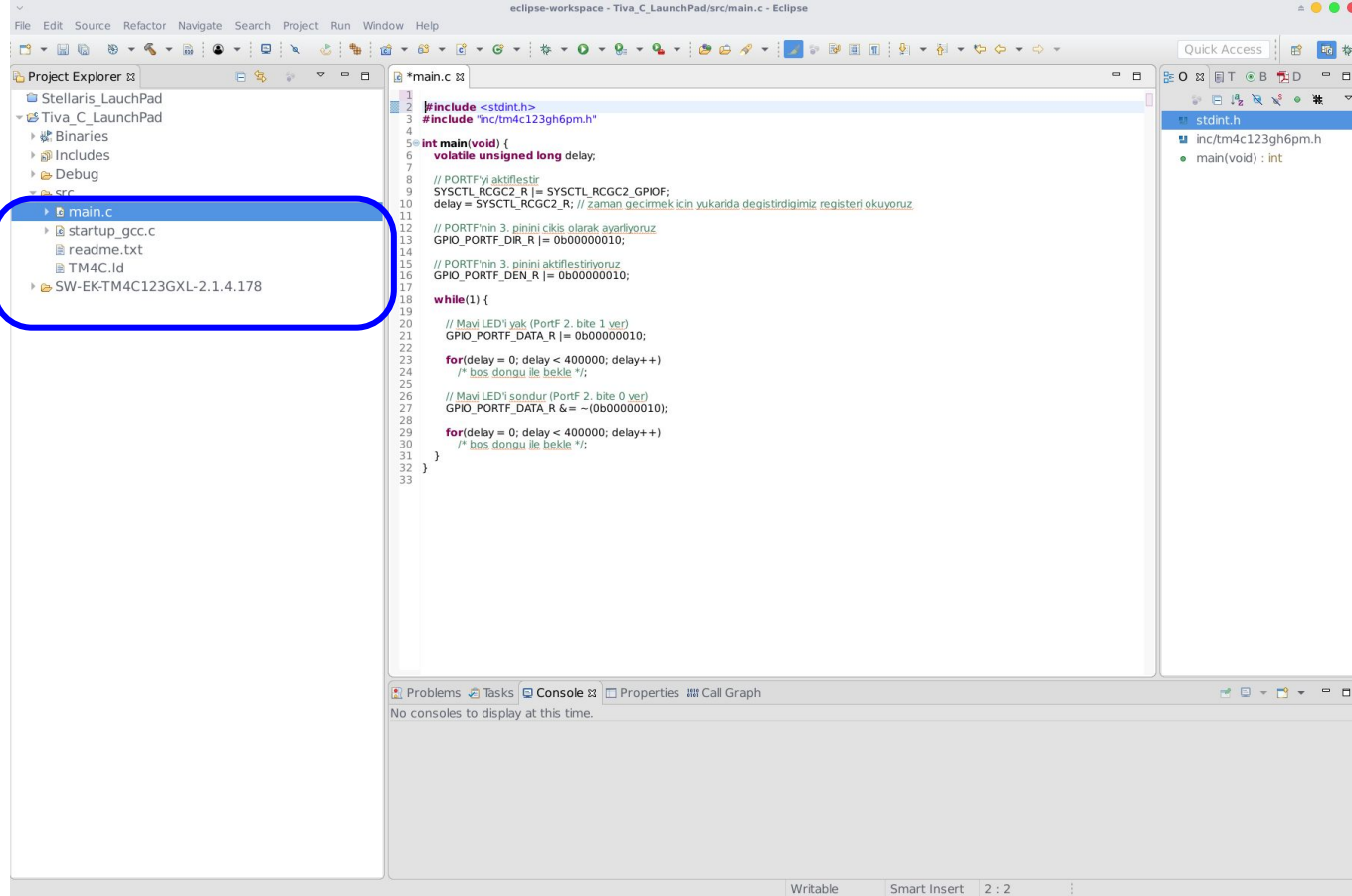
Kartın Sanal Makinaya Aktarılması



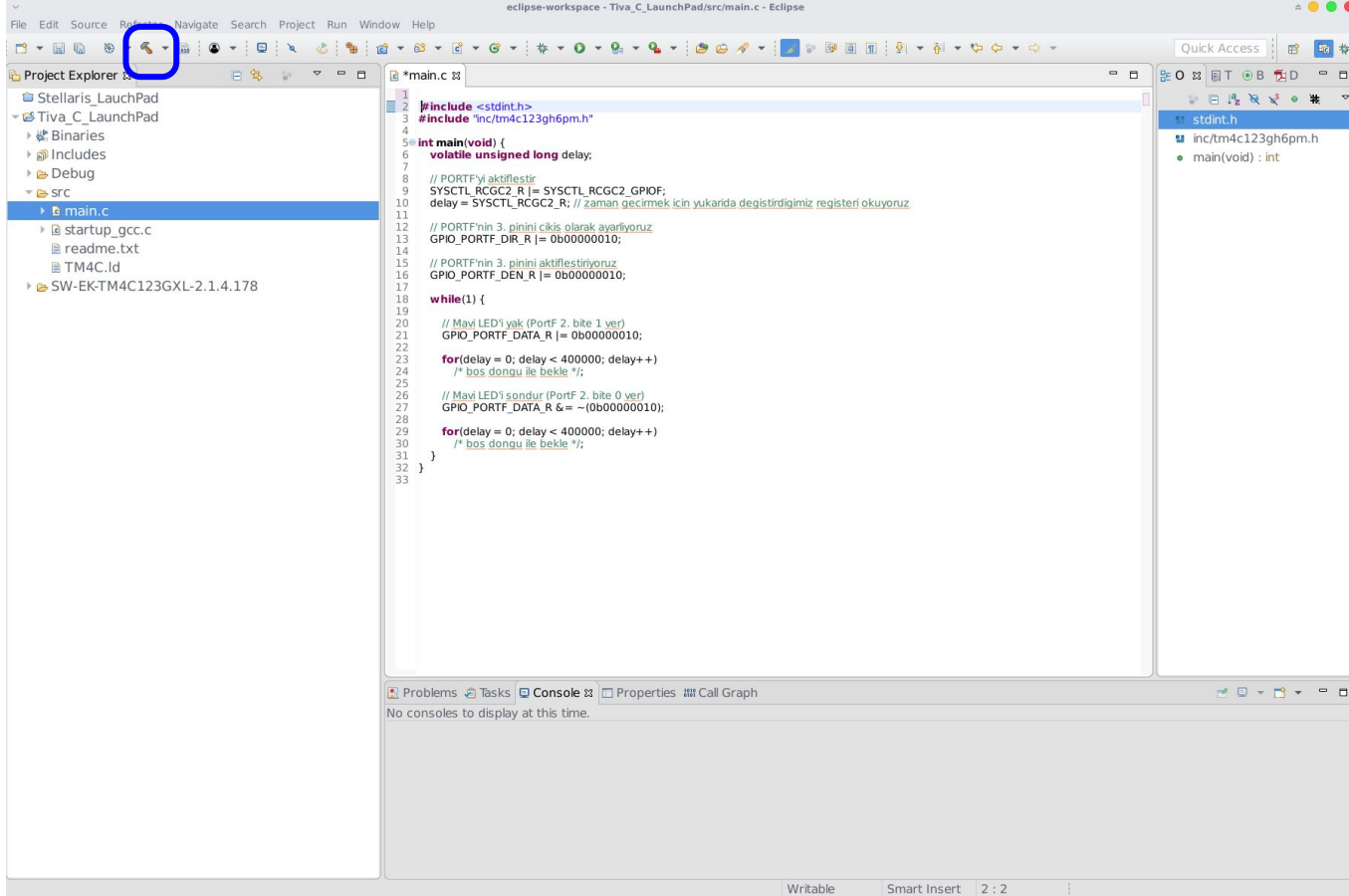
İlk Projenin Çalıştırılması



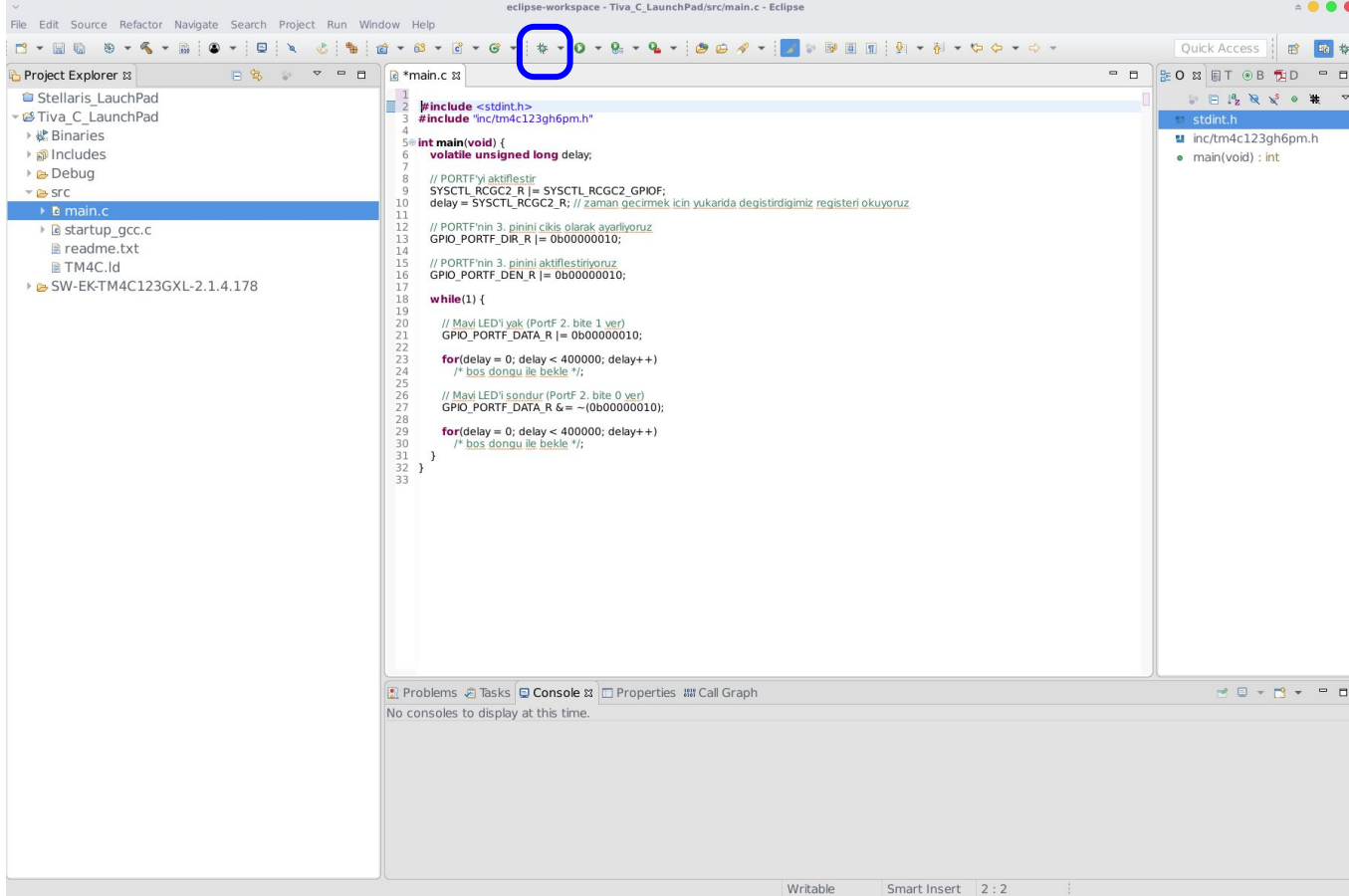
İlk Projenin Çalıştırılması



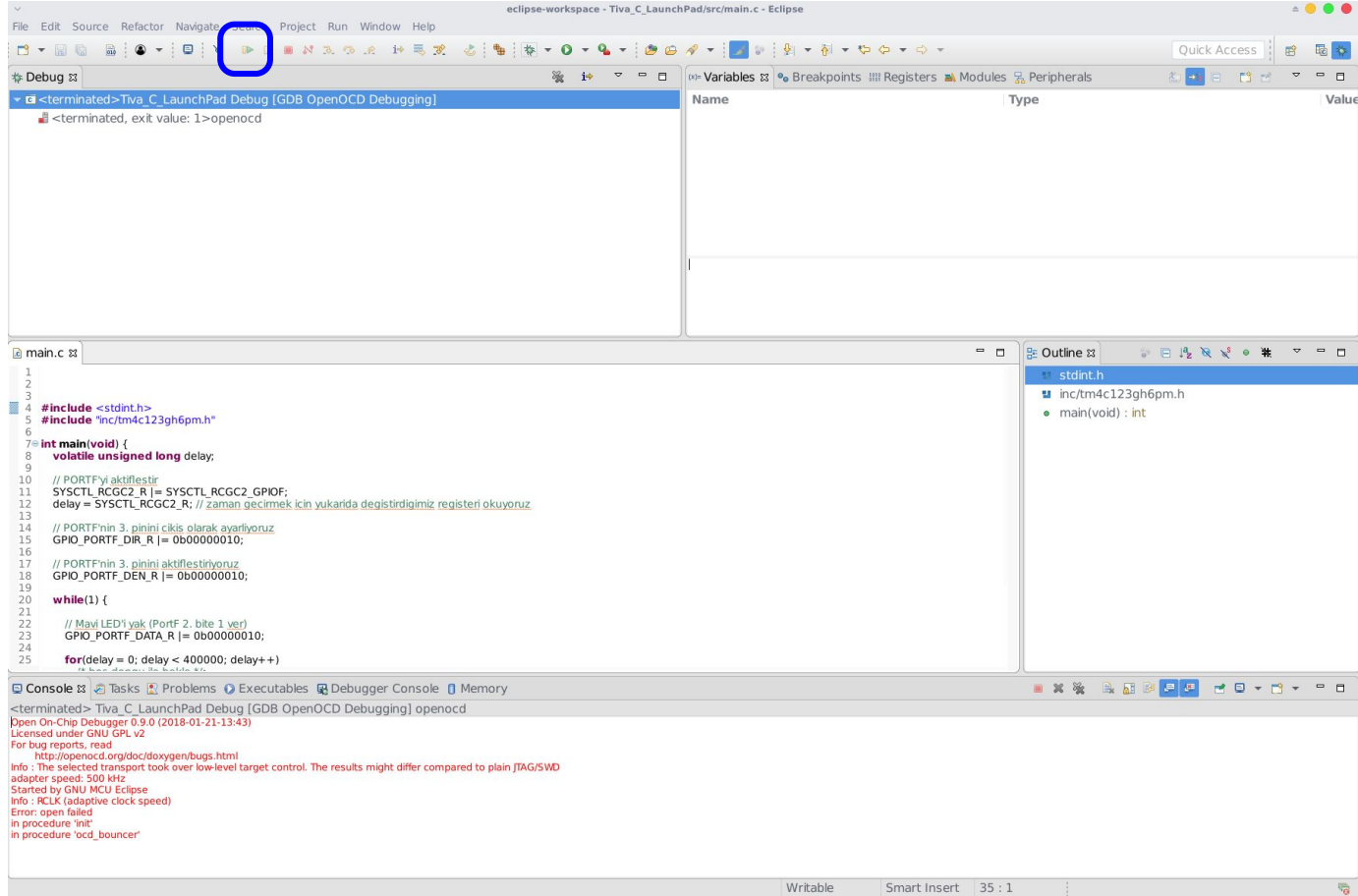
İlk Projenin Çalıştırılması



İlk Projenin Çalıştırılması



İlk Projenin Çalıştırılması



The screenshot shows the Eclipse IDE interface with the following components:

- Top Menu Bar:** File, Edit, Source, Refactor, Navigate, **Debug** (highlighted with a blue circle), Project, Run, Window, Help.
- Debug Console:** Shows the status "<terminated> Tiva_C_LaunchPad Debug [GDB OpenOCD Debugging]" and the message "<terminated, exit value: 1>openocd".
- Variables Panel:** Empty, showing Name, Type, and Value columns.
- Source Editor:** Displays the code in `main.c`. The code includes `<stdint.h>` and `inc/tm4c123gh6pm.h`. It defines a `main(void)` function that sets up GPIOs and turns on an LED. The code is as follows:

```
1
2
3
4 #include <stdint.h>
5 #include "inc/tm4c123gh6pm.h"
6
7 int main(void) {
8     volatile unsigned long delay;
9
10    // PORTF'yi aktifleştir
11    SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOF;
12    delay = SYSCTL_RCGC2_R; // zaman gecirmek için yukarıda degistirdigimiz registeri okuyoruz
13
14    // PORTF'nin 3. pinini cikis olarak ayarlıyoruz
15    GPIO_PORTF_DIR_R |= 0b000000010;
16
17    // PORTF'nin 3. pinini aktifleştiriyoruz
18    GPIO_PORTF_DEN_R |= 0b000000010;
19
20    while(1) {
21
22        // Mavi LED'i yak (PortF 2. bite 1 ver)
23        GPIO_PORTF_DATA_R |= 0b000000010;
24
25        for(delay = 0; delay < 400000; delay++)
```
- Outline Panel:** Shows the project structure with `stdint.h`, `inc/tm4c123gh6pm.h`, and `main(void) : int`.
- Console Panel:** Shows the output of the debug session, including the error message: "Error: open failed in procedure 'init' in procedure 'ocd_bouncer'".

İlk Projenin Çalıştırılması

The screenshot shows the Eclipse IDE interface with the following components:

- Top Menu Bar:** File, Edit, Source, Refactor, Navigate, Search, **Run** (highlighted with a blue circle), Window, Help.
- Toolbar:** Various icons for file operations, editing, and debugging.
- Debugger Console:** Shows the status of the debug session. The top line indicates the session is terminated: `<terminated> Tiva_C_LaunchPad Debug [GDB OpenOCD Debugging]`. Below it, the exit value is shown: `<terminated, exit value: 1>openocd`.
- Variables Panel:** Empty, showing no variables are currently being monitored.
- Breakpoints Panel:** Empty.
- Registers Panel:** Empty.
- Modules Panel:** Empty.
- Peripherals Panel:** Empty.
- Editor:** Displays the source code of `main.c`. The code includes headers, defines a delay function, and sets up GPIO pins for an LED. The code is as follows:

```
1
2
3
4 #include <stdint.h>
5 #include "inc/tm4c123gh6pm.h"
6
7 int main(void) {
8     volatile unsigned long delay;
9
10    // PORTF'yi aktifleştir
11    SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOF;
12    delay = SYSCTL_RCGC2_R; // zaman gecirmek için yukarıda degistirdigimiz registeri okuyoruz
13
14    // PORTF'nin 3. pinini cikis olarak ayarlıyoruz
15    GPIO_PORTF_DIR_R |= 0b000000010;
16
17    // PORTF'nin 3. pinini aktifleştiriyoruz
18    GPIO_PORTF_DEN_R |= 0b000000010;
19
20    while(1) {
21
22        // Mavi LED'i yak (PortF 2. bite 1 ver)
23        GPIO_PORTF_DATA_R |= 0b000000010;
24
25        for(delay = 0; delay < 400000; delay++)
```
- Outline Panel:** Shows the project structure with `stdint.h`, `inc/tm4c123gh6pm.h`, and `main(void) : int`.
- Console:** Shows the output of the debug session. The top line indicates the session is terminated: `<terminated> Tiva_C_LaunchPad Debug [GDB OpenOCD Debugging] openocd`. Below it, the Open On-Chip Debugger version and license information are displayed. The console also shows an error message: `Error: open failed in procedure 'init' in procedure 'ocd_bouncer'`.

Hatalar

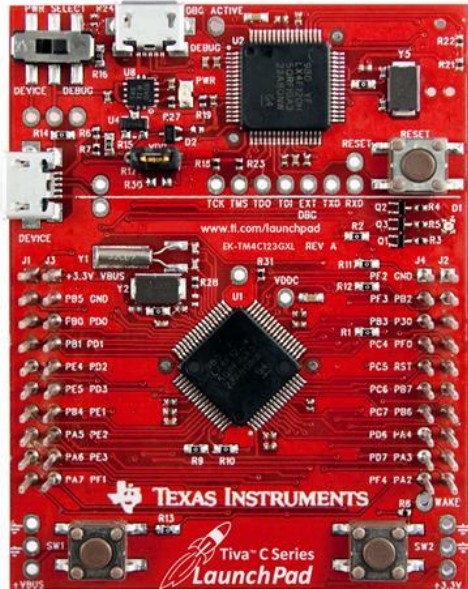
1. Stellaris/Tiva donanımları bilgisayara bağlı mı?
2. Donanımlar sanal makinaya aktarıldı mı?
3. “DEBUG” islemi açık unutuldu mu?



Gelistirme Kartlari

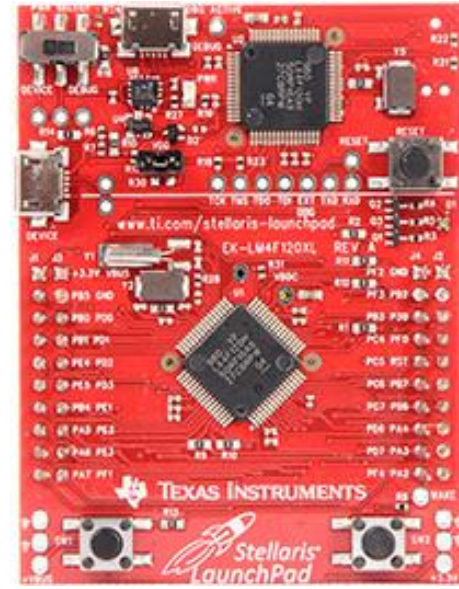
Tiva C Series EK-TM4C123GXL

<http://www.ti.com/litv/pdf/spmu296>

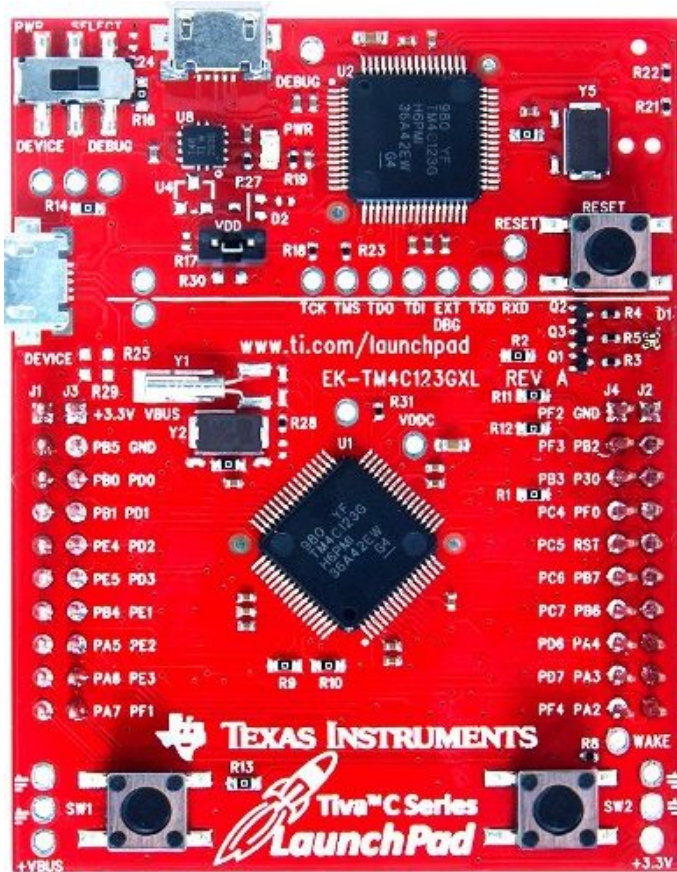


Stellaris EK-LM4F120XL

<http://www.ti.com/lit/ug/spmu289c/spmu289c>



Tiva & Stellaris Port Bağlantıları



Hardware

digitalRead() and digitalWrite() PORTS

analogRead()

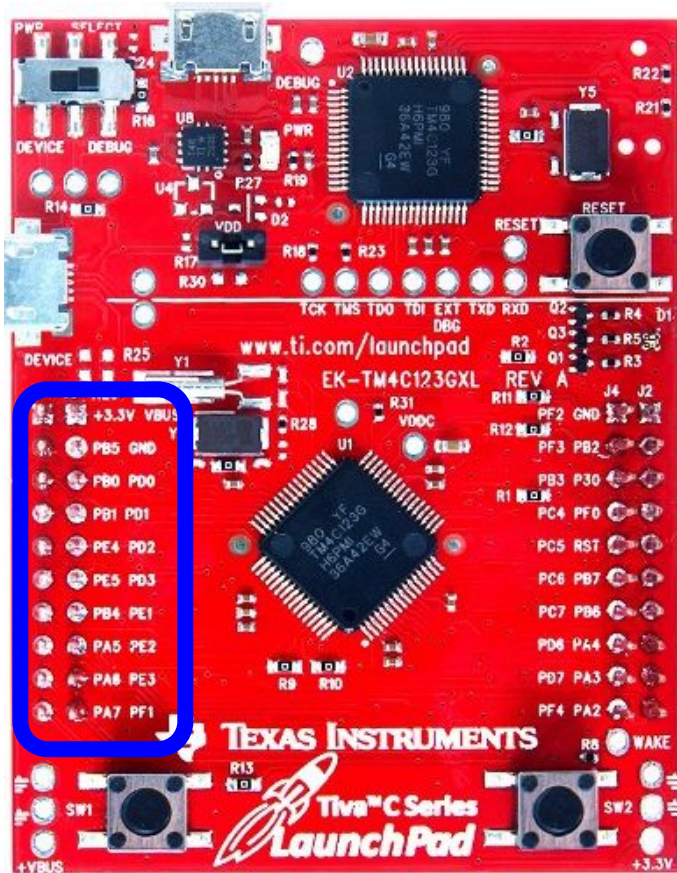
analogWrite()

I²C (TWI)

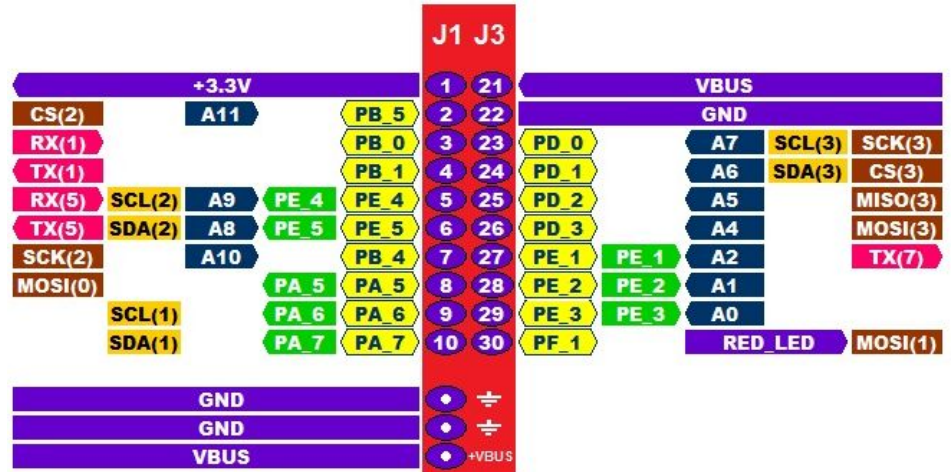
SPI

Hardware Serial

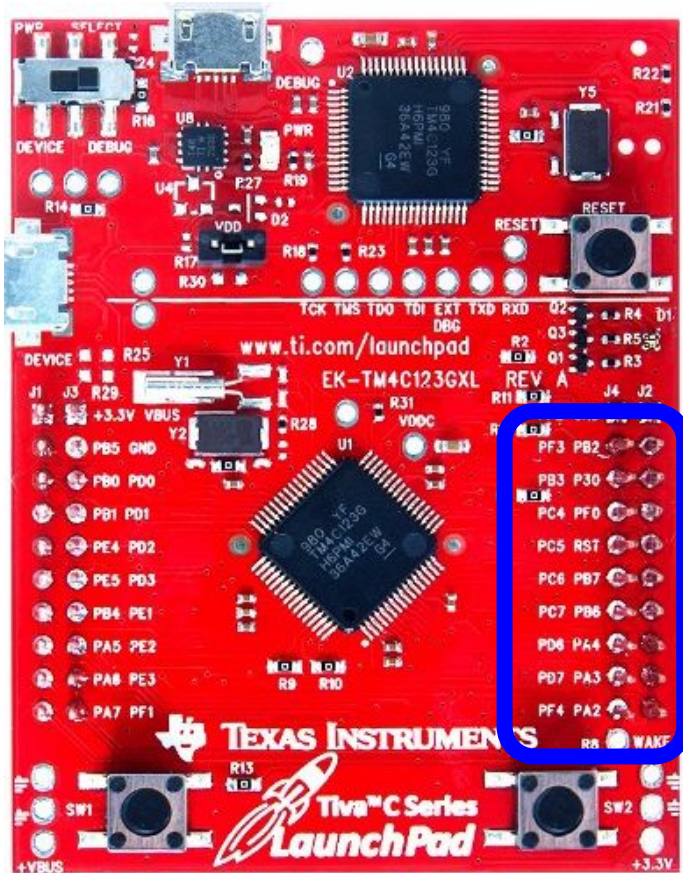
Tiva & Stellaris Port Bağlantıları



Hardware
digitalRead() and digitalWrite() PORTS
analogRead()
analogWrite()
I ² C (TWI)
SPI
Hardware Serial



Tiva Port Bağlantıları



Hardware
digitalRead() and digitalWrite() PORTS
analogRead()
analogWrite()
I ² C (TWI)
SPI
Hardware Serial

				J4	J2				
SCK(1)	BLUE_LED	PF2	PF 2	40	20	GND			
CS(1)	GREEN_LED	PF 3	PF 3	39	19	PB_2	PB 2	SCL(0)	
	SDA(0)	PB 3	PB 3	38	18	PE_0		A3	RX(7)
RX(1)		PC 4	PC 4	37	17	PF_0		PUSH2	MISO(1)
TX(1)		PC 5	PC 5	36	16	RESET			
RX(3)		PC 6	PC 6	35	15	PB_7			MOSI(2)
TX(3)		PC 7	PC 7	34	14	PB_6			MISO(2)
RX(2)		PD_6	PD 6	33	13	PA_4	PA_4		MISO(0)
TX(2)		PD 7	PD 7	32	12	PA_3	PA-3		CS(0)
	PUSH1	PF 4	PF 4	31	11	PA_2	PA-2		SCK(0)
						GND			
						GND			
						+3.3V			

Tiva & Stellaris Port Bağlantıları

Flash	256	KB
SRAM	32	KB

Serial	hardware	
ADC	12	bits
Use pins numbers only!		

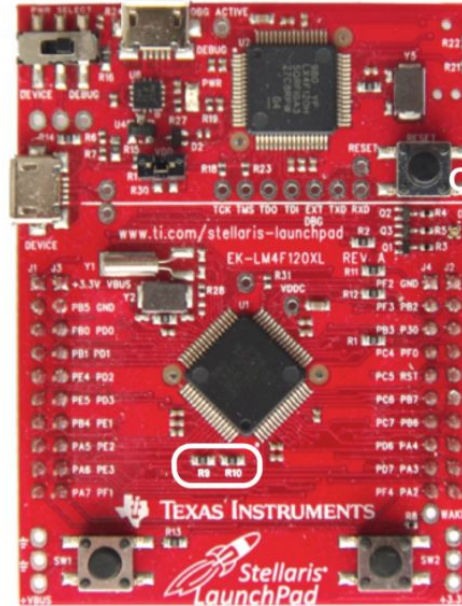
Hardware
Pin number
Other pin number

I2C

Serial UART

SPI

- analogRead()
- digitalRead() and digitalWrite()
- digitalRead(), digitalWrite() and analogWrite()

[illegible]VBUS detection PD_7

		+3.3V		J1	J3				
CS (2)		A11	PB_5	1	21		VBUS		
RX (1)			PB_0	2	22		GROUND		
TX (1)			PB_1	3	23	PD_0	A7	SCL (3)	SCK (3)
RX (5)	SCL (2)	A9	PE_4	4	24	PD_1	A6	SDA (3)	CS (3)
TX (5)	SDA (2)	A8	PE_5	5	25	PD_2	A5		MISO (3)
SCK (2)		A10	PB_4	6	26	PD_3	A4		MOSI (3)
MOSI (0)			PA_5	7	27	PE_1	A2		TX (7)
			PA_6	8	28	PE_2	A1		
	SCL (1)		PA_6	9	29	PE_3	A0		
	SDA (1)		PA_7	10	30	PF_1	RED_LED		MOSI (1)
							GROUND		
							GROUND		
							VBUS		

			0Ω shunt		
23	SCL (3)	PD_0	R9	PB_6	14
24	SDA (3)	PD_1	R10	PB_7	15

Tiva & Stellaris Port Bağlantıları

Flash	256	KB
SRAM	32	KB

Serial	hardware	
ADC	12	bits
Use pins numbers only!		

		GROUND	J2 20	J4 40	PF_2	BLUE_LED	SCK (1)
	SCL (0)	PB_2	19	39	PF_3	GREEN_LED	CS (1)
RX (7)	A3	PE_0	18	38	PB_3		
MISO (1)	PUSH2	PF_0	17	37	PC_4		RX (1)
	RESET		16	36	PC_5		TX (1)
		PB_7	15	35	PC_6		RX (3)
MOSI (2)		PB_6	14	34	PC_7		TX (3)
MISO (2)		PA_4	13	33	PD_6		RX (2)
MISO (0)		PA_3	12	32	PD_7	Detection	TX (2)
CS (0)		PA_2	11	31	PF_4	PUSH1	
SCK (0)							
					GROUND		
					GROUND		
					VBUS		

			0Ω shunt		
23	SCL (3)	PD_0	R9	PB_6	14
24	SDA (3)	PD_1	R10	PB_7	15



Hardware
Pin number
Other pin number



I²C

Serial UART

SPI

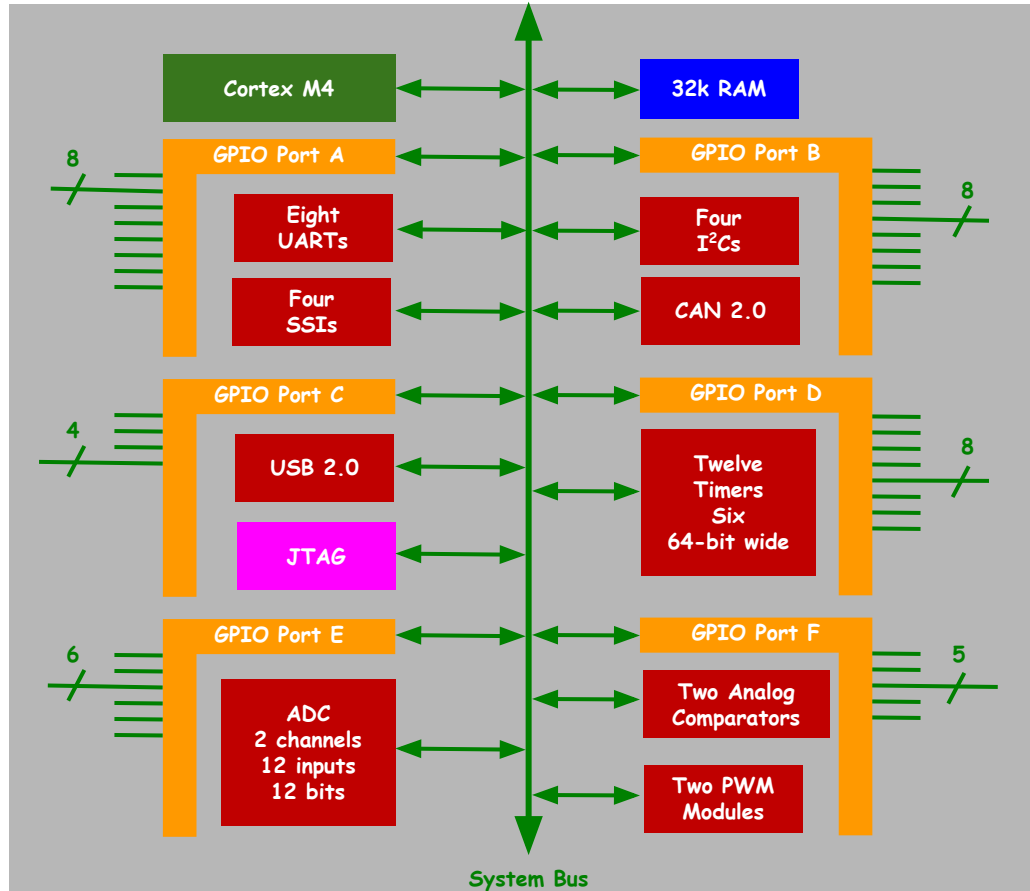
- analogRead()
- digitalRead() and digitalWrite()
- digitalRead(), digitalWrite() and analogWrite()

[illegible]

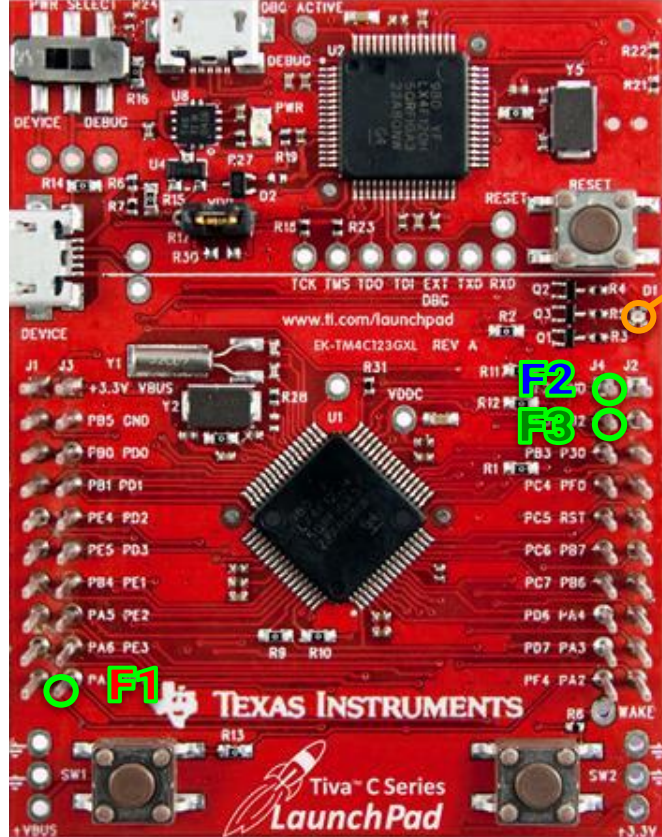
	GROUND	PA_0	RX (0)
	GROUND	PA_1	TX (0)
	VBUS	PD_4	RX (6)
		PD_5	TX (6)

VBUS detection PD_7

I/O Portları



Kartın üzerindeki ledi yakma



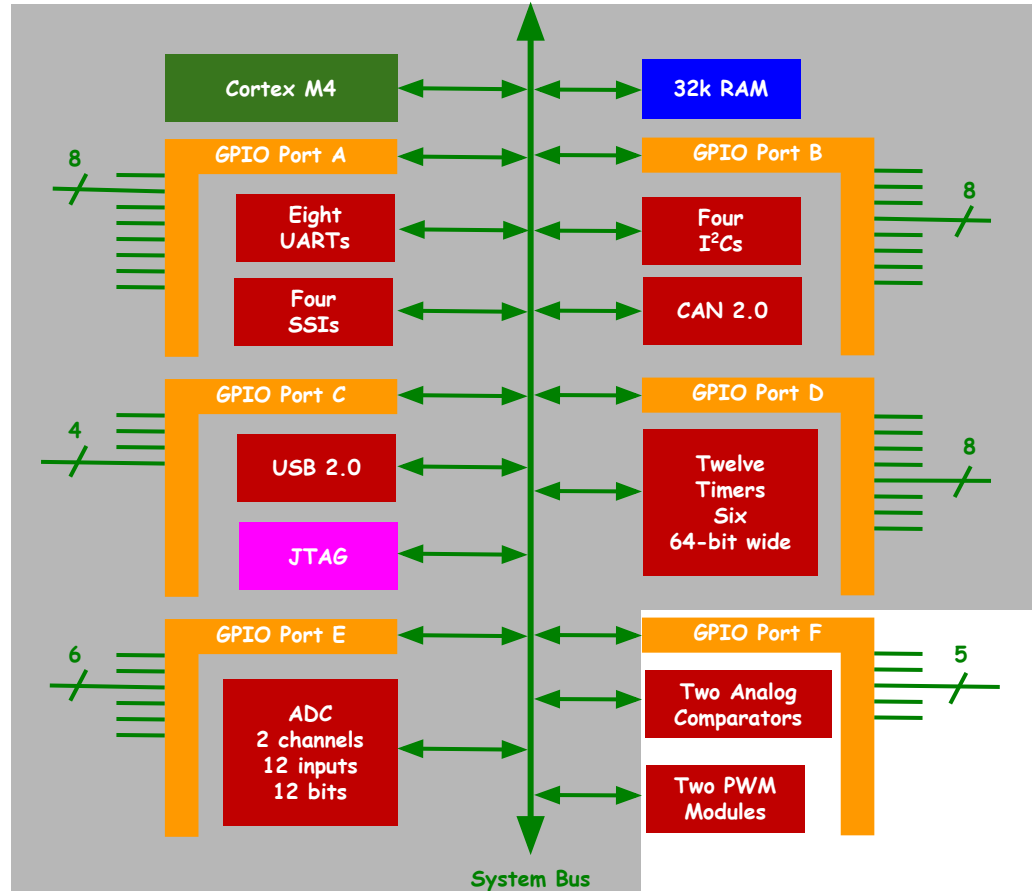
Kırmızı LED
Mavi LED
Yesil LED

Port F1 --> Kırmızı LED

Port F2 --> Mavi LED

Port F3 --> Yesil LED

Kartın üzerindeki ledi yakma



Kartın üzerindeki ledi yakma

```
#include <stdint.h>
```

```
#include "inc/tm4c123gh6pm.h"
```

```
int main(void) {
```

```
}
```

Gerekli Kütüphanelerin Eklenmesi

(int8_t, int16_t, int32_t, uint8_t, uint16_t, uint32_t) tipindeki tamsayı değerlerinin kullanılmasını sağlar

```
#include <stdint.h>
```

Gerekli Kütüphanelerin Eklenmesi

(int8_t, int16_t, int32_t, uint8_t, uint16_t, uint32_t) tipindeki tamsayı değerlerinin kullanılmasını sağlar

```
#include <stdint.h>
```

tiva kütüphanesi

```
#include "inc/tm4c123gh6pm.h"
```

Gerekli Kütüphanelerin Eklenmesi

(int8_t, int16_t, int32_t, uint8_t, uint16_t, uint32_t) tipindeki tamsayı değerlerinin kullanılmasını sağlar

```
#include <stdint.h>
```

tiva kütüphanesi

```
#include "inc/tm4c123gh6pm.h"
```

stellaris kütüphanesi

```
#include "inc/lm4f120h5qr.h"
```

Kartın üzerindeki ledi yakma

```
#include <stdint.h>
```

```
#include "inc/tm4c123gh6pm.h"
```

```
int main(void) {
```

```
    volatile unsigned long delay;
```

```
}
```

“volatile” (uçucu) degisken

```
int delay = 0;  
while(delay == 0);
```

“volatile” (uçucu) degisken

Degiskenin degerini donanım degistirebilir;

```
int delay = 0;
```

```
while(delay == 0);
```


“volatile” (uçucu) degisken

Degiskenin degerini donanım degistirebilir;

```
int delay = 0;
```

```
while(delay == 0);
```

!!!!!!!!!!!!

Derleyici donanım seviyesine inmeden degiskeni kaldırır:

“volatile” (uçucu) degisken

Degiskenin degerini donanım degistirebilir;

```
int delay = 0;  
while(delay == 0);
```

!!!!!!!!!!!!

Derleyici donanım seviyesine inmeden degiskeni kaldırır:

Derleyicinin degiskene müdahil olmasını engellemek için;

```
volatile int delay = 0;  
while(delay == 0);
```

Kartın üzerindeki ledi yakma

```
#include <stdint.h>
```

```
#include "inc/tm4c123gh6pm.h"
```

```
int main(void) {
```

```
    volatile unsigned long delay;
```

```
    SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOF;
```

```
}
```

System Control Run Mod Clock Gate Control Register

// Port F sayacını aktifleştirir

```
SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOF;
```

System Control Run Mod Clock Gate Control Register

Adres	7	6	5	4	3	2	1	0	İsim
\$400F.E108	GPIOF	GPIOE	GPIOD	GPIOC	GPIOB	GPIOA	SYSCTL_RCGC2_R

0b00000001 = 0x01

0b00000010 = 0x02

0b00000100 = 0x04

0b00001000 = 0x08

0b00010000 = 0x10

0b00100000 = 0x20

System Control Run Mod Clock Gate Control Register

Adres	7	6	5	4	3	2	1	0	İsim
\$400F.E108	GPIOF	GPIOE	GPIOD	GPIOC	GPIOB	GPIOA	SYSCTL_RCGC2_R

0b00000001 = 0x01 SYSCTL_RCGC2_R |= 0x01; // Port A sayacını aktifleştirir
 0b00000010 = 0x02 SYSCTL_RCGC2_R |= 0x02; // Port B sayacını aktifleştirir
 0b00000100 = 0x04 SYSCTL_RCGC2_R |= 0x04; // Port C sayacını aktifleştirir
 0b00001000 = 0x08 SYSCTL_RCGC2_R |= 0x08; // Port D sayacını aktifleştirir
 0b00010000 = 0x10 SYSCTL_RCGC2_R |= 0x10; // Port E sayacını aktifleştirir
 0b00100000 = 0x20 SYSCTL_RCGC2_R |= 0x20; // Port F sayacını aktifleştirir

System Control Run Mod Clock Gate Control Register

// Port F sayacını aktifleştirir

```
SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOF;
```

SYSCTL_RCGC2_R x x x x x x x x

System Control Run Mod Clock Gate Control Register

// Port F sayacını aktifleştirir

```
SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOF;
```

SYSCTL_RCGC2_R	x x x x	x x x x
----------------	---------	---------

SYSCTL_RCGC2_GPIOF	0 0 1 0	0 0 0 0
--------------------	---------	---------

System Control Run Mod Clock Gate Control Register

// Port F sayacını aktifleştirir

```
SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOF;
```

SYSCTL_RCGC2_R	x x x x	x x x x
----------------	---------	---------

SYSCTL_RCGC2_GPIOF	0 0 1 0	0 0 0 0
--------------------	---------	---------

OR

System Control Run Mod Clock Gate Control Register

// Port F sayacını aktifleştirir

```
SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOF;
```

SYSCTL_RCGC2_R	x x x x	x x x x
----------------	---------	---------

SYSCTL_RCGC2_GPIOF	0 0 1 0	0 0 0 0
--------------------	---------	---------

OR

SYSCTL_RCGC2_R	x x 1 x	x x x x
----------------	---------	---------

System Control Run Mod Clock Gate Control Register

Adres	7	6	5	4	3	2	1	0	İsim
\$400F.E108	GPIOF	GPIOE	GPIOD	GPIOC	GPIOB	GPIOA	SYSCTL_RCGC2_R

```
SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOA; // Port A clock aktifleştirir
SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOB; // Port B clock aktifleştirir
SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOC; // Port C clock aktifleştirir
SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOD; // Port D clock aktifleştirir
SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOE; // Port E clock aktifleştirir
SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOF; // Port F clock aktifleştirir
```

Kartın üzerindeki ledi yakma

```
#include <stdint.h>
```

```
#include "inc/tm4c123gh6pm.h"
```

```
int main(void) {
```

```
    volatile unsigned long delay;
```

```
    SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOF;
```

```
    delay = SYSCTL_RCGC2_R;
```

```
}
```

Kartın üzerindeki ledi yakma

// zaman gecirmek için yukarıda degistirilen saklayıcı okunur

```
delay = SYSCTL_RCGC2_R;
```

Kartın üzerindeki ledi yakma

```
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
int main(void) {
    volatile unsigned long delay;

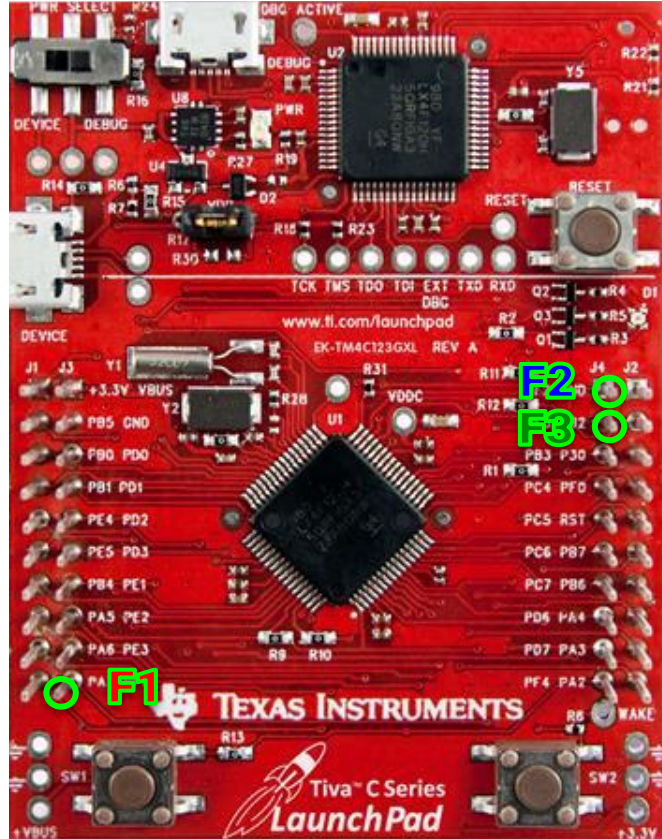
    SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOF;

    delay = SYSCTL_RCGC2_R;

    GPIO_PORTF_DIR_R |= 0b00000100;

}
```

Kartın üzerindeki ledi yakma

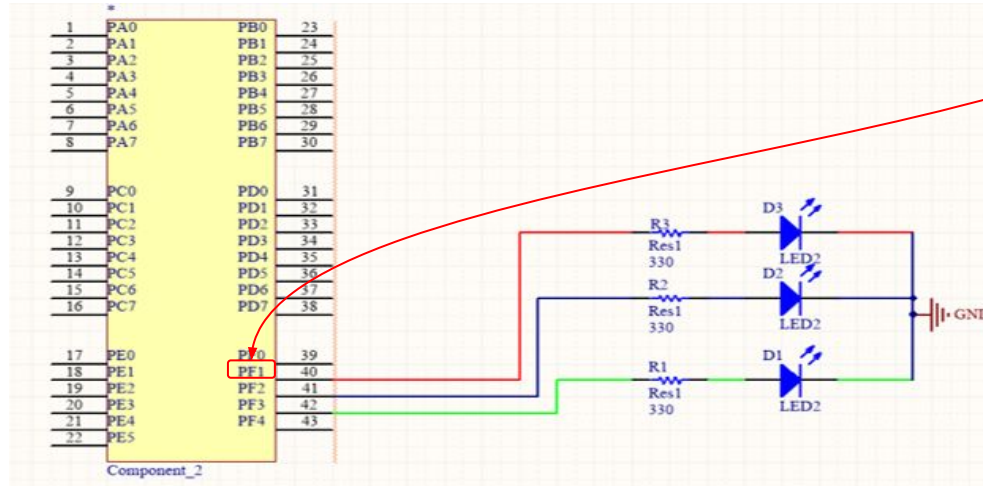
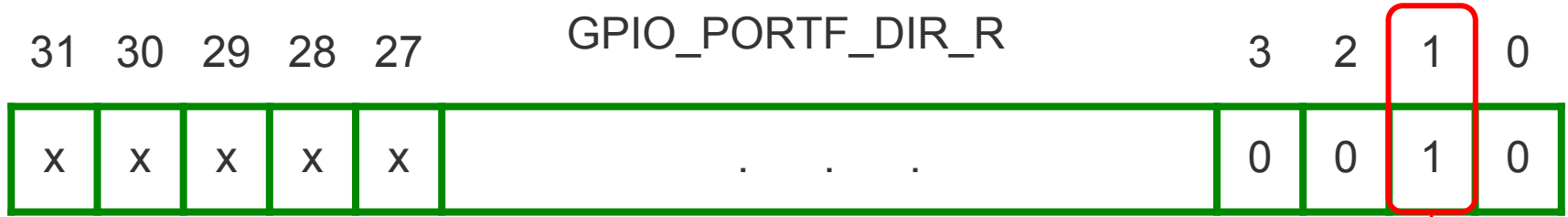


Port F1 --> Kırmızı LED

Port F2 --> Mavi LED

Port F3 --> Yeşil LED

Kartın üzerindeki ledi yakma



Direction Register

```
GPIO_PORTF_DIR_R |= 0b00000010; // PF1 pinini cikis yap
```

Direction Register

```
GPIO_PORTF_DIR_R |= 0b00000010; // PF1 pinini cikis yap
```

GPIO_PORTF_DIR_R

x x x x x x x x

Direction Register

```
GPIO_PORTF_DIR_R |= 0b00000010; // PF1 pinini cikis yap
```

GPIO_PORTF_DIR_R

x x x x x x x x

0 0 0 0 0 0 1 0

OR

Direction Register

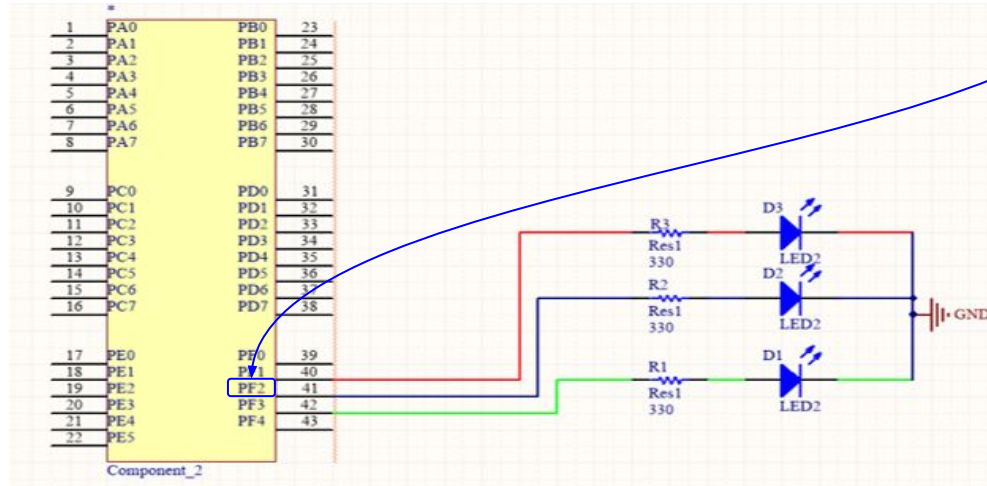
```
GPIO_PORTF_DIR_R |= 0b00000010; // PF1 pinini cikis yap
```

GPIO_PORTF_DIR_R	x x x x	x x x x
	0 0 0 0	0 0 1 0

OR

GPIO_PORTF_DIR_R	x x x x	x x 1 x
------------------	---------	---------

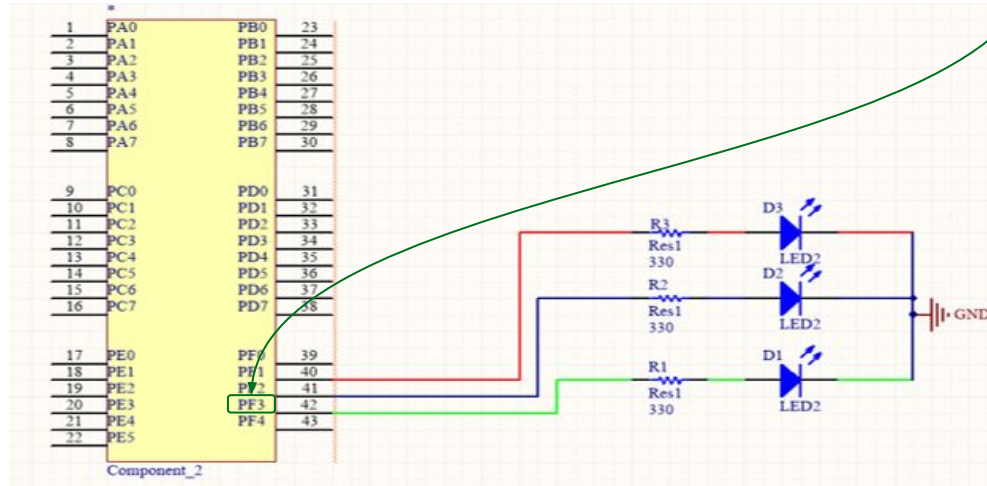
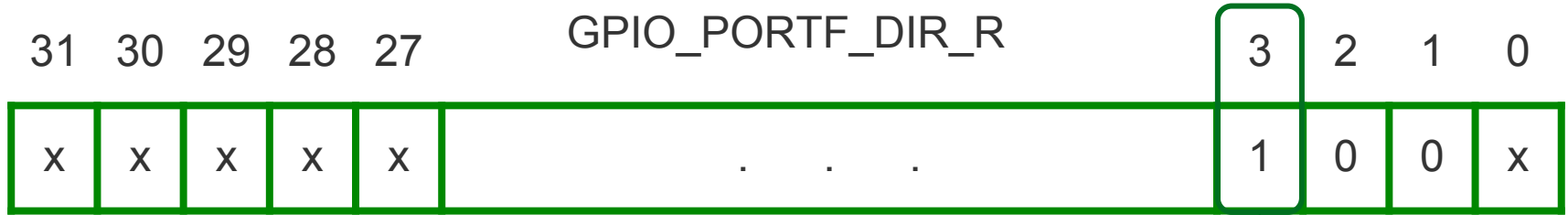
Direction Register



Direction Register

```
GPIO_PORTF_DIR_R |= 0b00000010; // PF1 pinini cikis yap  
GPIO_PORTF_DIR_R |= 0b00000100; // PF2 pinini cikis yap
```


Direction Register



Direction Register

```
GPIO_PORTF_DIR_R |= 0b00000010; // PF1 pinini cikis yap  
GPIO_PORTF_DIR_R |= 0b00000100; // PF2 pinini cikis yap  
GPIO_PORTF_DIR_R |= 0b00001000; // PF3 pinini cikis yap
```

Kartın üzerindeki ledi yakma

```
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
int main(void) {
    volatile unsigned long delay;

    SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOF;

    delay = SYSCTL_RCGC2_R;

    GPIO_PORTF_DIR_R |= 0b00000100;
    GPIO_PORTF_DEN_R |= 0b00000100;

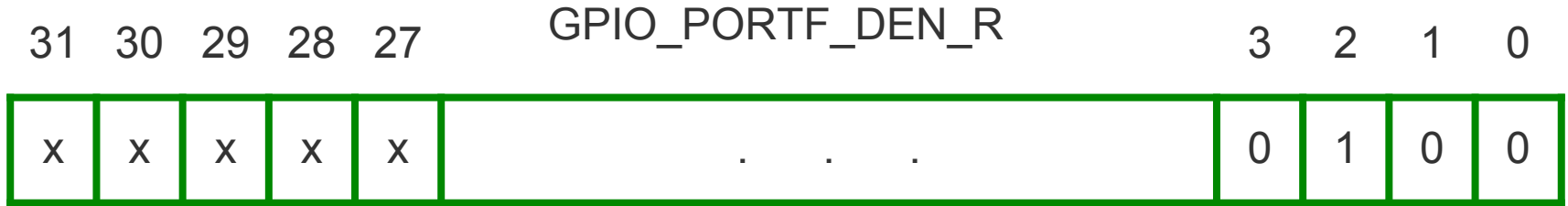
}
```

Digital Enable Register

```
GPIO_PORTF_DEN_R |= 0b00000100; // PF2 pinini aktiflestir
```

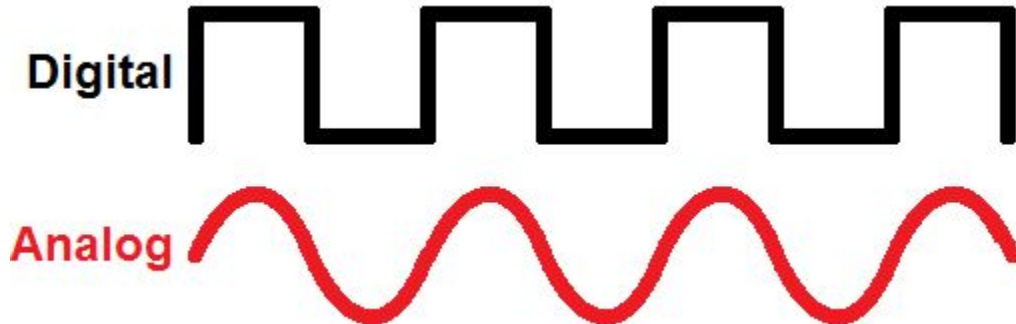
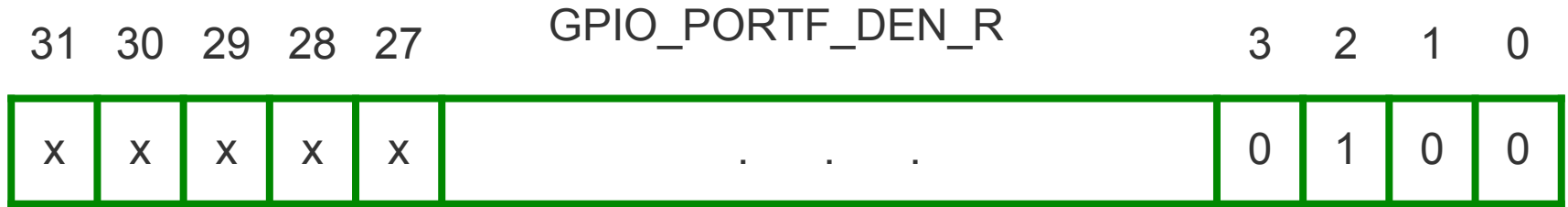
Digital Enable Register

```
GPIO_PORTF_DEN_R |= 0b00000100; // PF2 pinini aktiflestir
```



Digital Enable Register

```
GPIO_PORTF_DEN_R |= 0b00000100; // PF 3 pinini aktiflestir
```



Digital Enable Register

`GPIO_PORTF_DEN_R |= 0b00000100; // PF2 pinini aktiflestir`

`GPIO_PORTF_DEN_R`

`x x x x x x x x`

Digital Enable Register

`GPIO_PORTF_DEN_R |= 0b00000100; // PF2 pinini aktiflestir`

GPIO_PORTF_DEN_R

x x x x x x x x

0 0 0 0 0 1 0 0

OR

Digital Enable Register

`GPIO_PORTF_DEN_R |= 0b00000100; // PF2 pinini aktiflestir`

GPIO_PORTF_DEN_R	x x x x	x x x x
	0 0 0 0	0 1 0 0

OR

GPIO_PORTF_DEN_R	x x x x	x 1 x x
------------------	---------	---------

Kartın üzerindeki ledi yakma

```
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
int main(void) {
    volatile unsigned long delay;

    SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOF;

    delay = SYSCTL_RCGC2_R;

    GPIO_PORTF_DIR_R |= 0b00000100;
    GPIO_PORTF_DEN_R |= 0b00000100;

    while(1) {

}
```

Kartın üzerindeki ledi yakma

```
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
int main(void) {
    volatile unsigned long delay;

    SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOF;

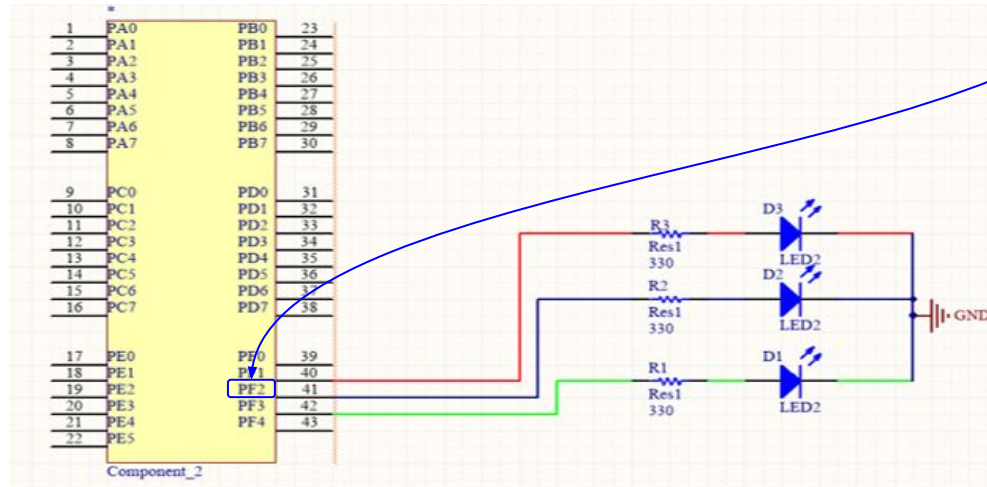
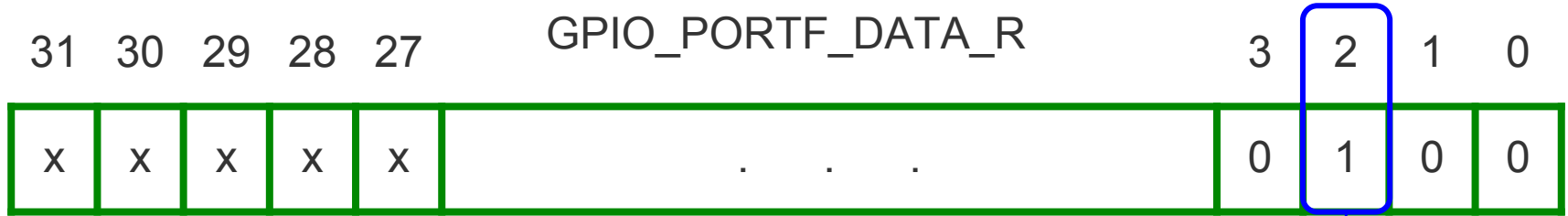
    delay = SYSCTL_RCGC2_R;

    GPIO_PORTF_DIR_R |= 0b00000100;
    GPIO_PORTF_DEN_R |= 0b00000100;

    while(1) {
        GPIO_PORTF_DATA_R |= 0b00000100;

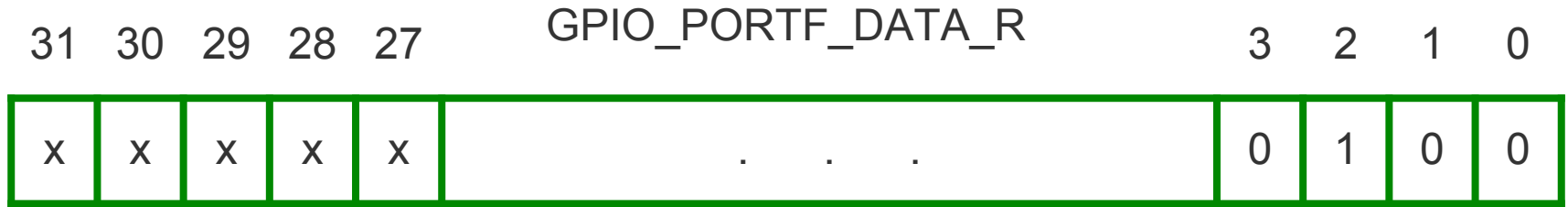
    }
}
```

Data Register



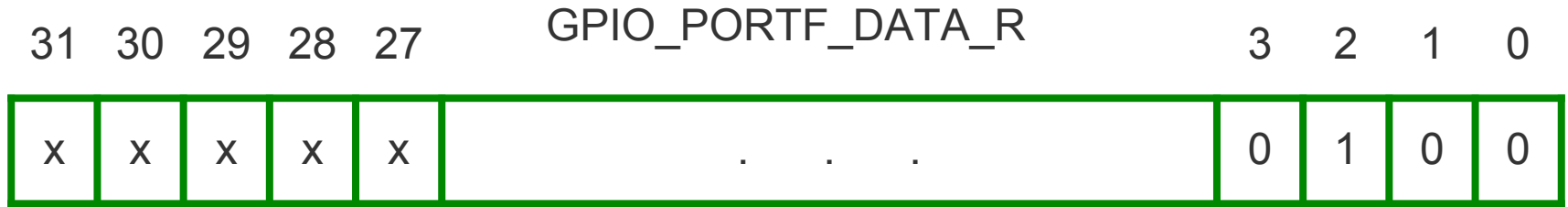
Data Register

GPIO_PORTF_DATA_R |= 0b00000100;



Data Register

GPIO_PORTF_DATA_R |= 0b00000100;



GPIO_PORTF_DATA_R

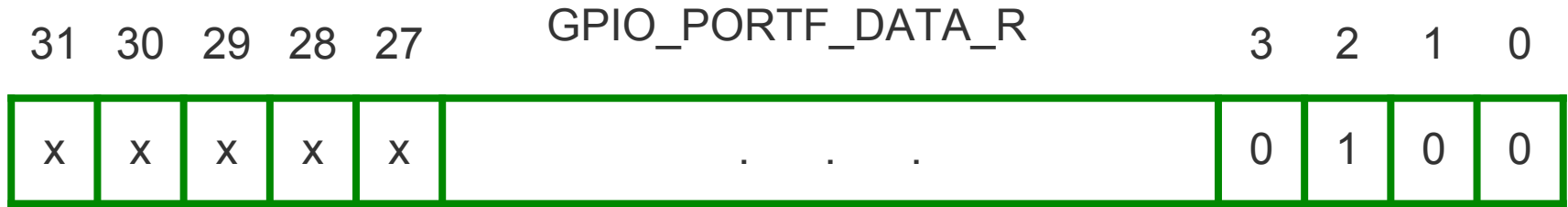
x x x x

0 1 0 0

OR

Data Register

GPIO_PORTF_DATA_R |= 0b00000100;



GPIO_PORTF_DATA_R

x x x x

0 1 0 0

OR

GPIO_PORTF_DATA_R

x 1 x x

Data Register

```
GPIO_PORTF_DATA_R |= 0b00000100;
```



Kartın üzerindeki ledi yakma

```
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
int main(void) {
    volatile unsigned long delay;

    SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOF;

    delay = SYSCTL_RCGC2_R;

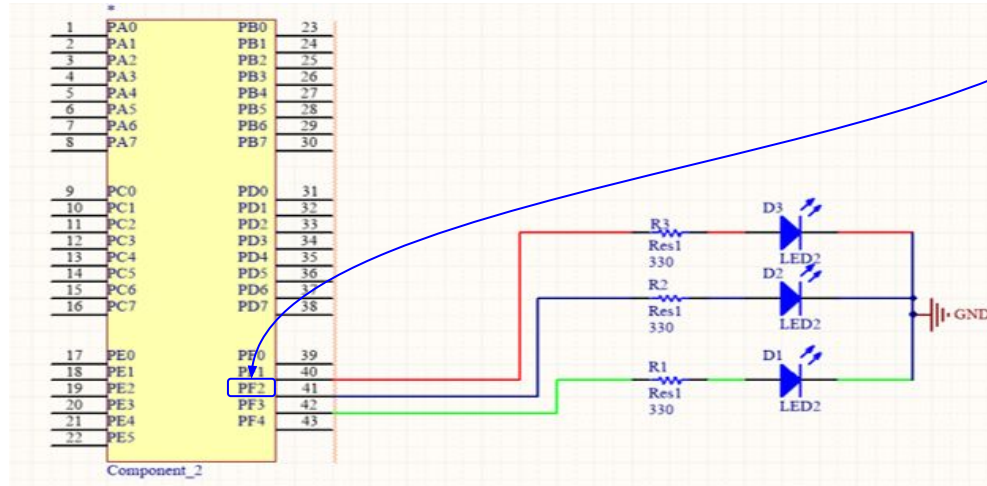
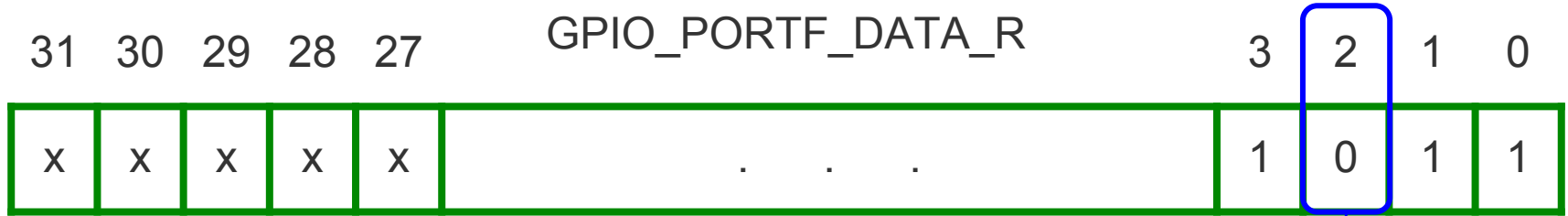
    GPIO_PORTF_DIR_R |= 0b00000100;
    GPIO_PORTF_DEN_R |= 0b00000100;

    while(1) {
        GPIO_PORTF_DATA_R |= 0b00000100;

        GPIO_PORTF_DATA_R &= ~(0b00000100);

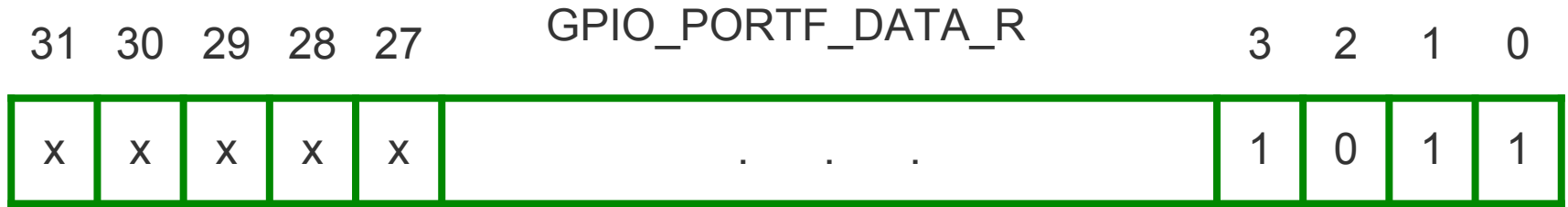
    }
}
```

Data Register



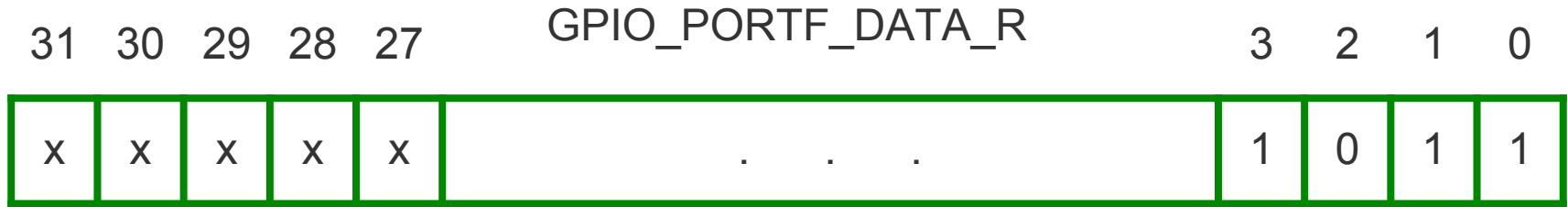
Data Register

```
GPIO_PORTF_DATA_R &= ~(0b00000100);
```



Data Register

```
GPIO_PORTF_DATA_R &= ~(0b00000100);
```



GPIO_PORTF_DATA_R

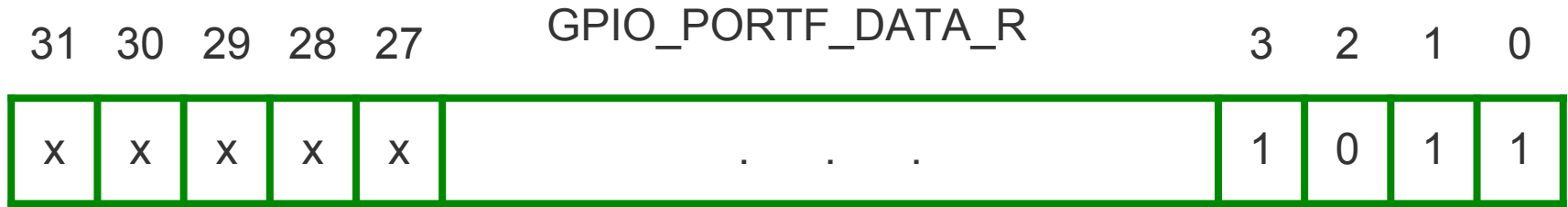
x x x x

1 0 1 1

AND

Data Register

```
GPIO_PORTF_DATA_R &= ~(0b00000100);
```



GPIO_PORTF_DATA_R

x x x x

1 0 1 1

AND

GPIO_PORTF_DATA_R

x 0 x x

Data Register

```
GPIO_PORTF_DATA_R &= ~(0b00000100);
```



Kartın üzerindeki ledi yakma

```
#include <stdint.h>
#include "inc/tm4c123gh6pm.h"
int main(void) {
    volatile unsigned long delay;

    SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOF;

    delay = SYSCTL_RCGC2_R;

    GPIO_PORTF_DIR_R |= 0b00000100;
    GPIO_PORTF_DEN_R |= 0b00000100;

    while(1) {
        GPIO_PORTF_DATA_R |= 0b00000100;
        for(delay = 0; delay < 400000; delay++)
            /* bos dongu ile bekle */;

        GPIO_PORTF_DATA_R &= ~(0b00000100);
        for(delay = 0; delay < 400000; delay++)
            /* bos dongu ile bekle */;
    }
}
```

Data Register

```
while(1) {  
    GPIO_PORTF_DATA_R |= 0b00000100;  
    for(delay = 0; delay < 400000; delay++)  
        /* bos dongu ile bekle */;  
  
    GPIO_PORTF_DATA_R &= ~(0b00000100);  
    for(delay = 0; delay < 400000; delay++)  
        /* bos dongu ile bekle */;  
}
```



Sorular

