

STM32CubeIDE'de Sadece CMSIS ile Proje Kurulumu

Yöntem 1: Empty Project ile Başlama

Adım 1: Yeni Proje Oluşturma

1. **File** → **New** → **STM32 Project** seçeneğini tıklayın
2. **Target Selection** sayfasında:
 - **Board Selector** yerine **MCU/MPU Selector** sekmesini seçin
 - Kullanacağınız MCU'yu seçin (örn: STM32F407VGT6)
 - **Next** butonuna tıklayın

Adım 2: Proje Ayarları

Project Setup sayfasında:

- **Project Name:** Projenizin adını girin
- **Location:** Proje konumunu seçin
- **Targeted Project Type:** **Empty** seçeneğini işaretleyin
- **Targeted Language:** **C** seçili bırakın
- **Targeted Binary Type:** **Executable (.elf)** seçili bırakın
- **Next** butonuna tıklayın

Adım 3: Firmware Package

- **Download target firmware package** kutucuğunu işaretleyin
- **Finish** butonuna tıklayın

Yöntem 2: Manuel CMSIS Kurulumu

Gerekli Dosyalar ve Klasör Yapısı

Proje oluştuktan sonra şu klasör yapısını oluşturun:

```
ProjectName/
├── Core/
│   ├── Inc/
│   │   ├── main.h
│   │   └── stm32f4xx.h (veya chip'inize göre)
│   ├── Src/
│   │   ├── main.c
│   │   ├── system_stm32f4xx.c
│   │   └── syscalls.c
│   └── Startup/
│       └── startup_stm32f407vgtx.s (veya chip'inize göre)
├── Drivers/
│   ├── CMSIS/
│   │   ├── Include/
│   │   │   ├── core_cm4.h
│   │   │   ├── cmsis_version.h
│   │   │   ├── cmsis_compiler.h
│   │   │   └── mpu_armv7.h
│   │   └── Device/
│   │       ├── ST/
│   │       │   └── STM32F4xx/
│   │       │       ├── Include/
│   │       │       │   ├── stm32f4xx.h
│   │       │       │   ├── stm32f407xx.h
│   │       │       └── system_stm32f4xx.h
│   └── Linkers/
│       └── STM32F407VGTX_FLASH.ld
```

Adım 4: CMSIS Dosyalarını İndirme

1. [STM32 CMSIS Repository](#) adresinden gerekli dosyaları indirin
2. Alternatif: STM32CubeMX'den bir proje oluşturup CMSIS dosyalarını kopyalayın

Adım 5: Include Paths Ayarlama

1. **Project Explorer**'da projeye sağ tık → **Properties**
2. **C/C++ Build** → **Settings** → **Tool Settings** sekmesine gidin
3. **MCU GCC Compiler** → **Include paths** altına şu yolları ekleyin:

```
../Core/Inc
../Drivers/CMSIS/Include
../Drivers/CMSIS/Device/ST/STM32F4xx/Include
```

Adım 6: Preprocessor Defines Ekleme

MCU GCC Compiler → **Preprocessor** altına şu define'ları ekleyin:

```
STM32F407xx
USE_FULL_LL_DRIVER (isteğe bağlı)
ARM_MATH_CM4
```

Adım 7: Linker Script Ayarlama

1. **MCU GCC Linker** → **General** altında
2. **Script files (.ld)** bölümüne linker script yolunu ekleyin:

```
../Linkers/STM32F407VGTX_FLASH.ld
```

Temel Dosya İçerikleri

main.h

```
C

#ifndef __MAIN_H
#define __MAIN_H

#ifdef __cplusplus
extern "C" {
#endif

#include "stm32f4xx.h"

#ifdef __cplusplus
}
#endif

#endif /* __MAIN_H */
```

main.c

```
C
```

```

#include "main.h"

// System clock configuration
void SystemClock_Config(void);

int main(void)
{
    // System initialization
    SystemInit();
    SystemClock_Config();

    // Enable GPIO clock
    RCC->AHB1ENR |= RCC_AHB1ENR_GPIOAEN;

    // Configure PA5 as output (LED)
    GPIOA->MODER |= GPIO_MODER_MODER5_0;

    while (1)
    {
        // Toggle LED
        GPIOA->ODR ^= GPIO_ODR_ODR_5;

        // Simple delay
        for(volatile uint32_t i = 0; i < 1000000; i++);
    }
}

void SystemClock_Config(void)
{
    // Clock configuration will be here
    // For now, using default HSI clock
}

```

system_stm32f4xx.c

Bu dosyayı ST'nin CMSIS paketinden kopyalayın. SystemInit() ve SystemCoreClock değişkenini içerir.

Compiler ve Linker Ayarları

Optimization Settings

MCU GCC Compiler → Optimization:

- **Optimization Level:** -O0 (debug için) veya -Os (release için)

Debug Information

MCU GCC Compiler → Debugging:

- **Debug Level:** Maximum (-g3)
- **Debug Format:** DWARF-2

Linker Settings

MCU GCC Linker → General:

- **Remove unused sections:** İşaretleyin
- **Link time optimization:** İsteğe bağlı

Debugging Ayarları

Debug Configuration

1. Run → Debug Configurations
2. **STM32 Cortex-M C/C++ Application** seçin
3. **New launch configuration** oluşturun
4. **Debugger** sekmesinde:
 - **Debug probe:** ST-LINK (ST-LINK GDB server)
 - **Interface:** SWD
 - **Speed:** 4000 kHz

Yaygın Sorunlar ve Çözümleri

Problem 1: "undefined reference to `_sbrk`"

Çözüm: syscalls.c dosyasını projeye ekleyin veya şu define'ı ekleyin:

```
c

// In main.c
extern caddr_t _sbrk(int incr) {
    return (caddr_t) -1;
}
```

Problem 2: Startup dosyası bulunamıyor

Çözüm: Doğru startup dosyasını (.s) Startup klasörüne ekleyin ve build path'e dahil edin.

Problem 3: Clock ayarları çalışmıyor

Çözüm: system_stm32f4xx.c dosyasındaki SystemInit() fonksiyonunu kontrol edin.

Register Seviyesinde Programlama Örneği

c

```
// GPIO LED control example
void LED_Init(void) {
    // Enable GPIOA clock
    RCC->AHB1ENR |= RCC_AHB1ENR_GPIOAEN;

    // Configure PA5 as output
    GPIOA->MODER &= ~GPIO_MODER_MODER5;
    GPIOA->MODER |= GPIO_MODER_MODER5_0;

    // Set output speed
    GPIOA->OSPEEDR |= GPIO_OSPEEDER_OSPEEDR5;

    // No pull-up, no pull-down
    GPIOA->PUPDR &= ~GPIO_PUPDR_PUPDR5;
}

void LED_Toggle(void) {
    GPIOA->ODR ^= GPIO_ODR_ODR_5;
}
```

Bu rehberi takip ederek STM32CubeIDE'de sadece CMSIS kütüphanesi kullanarak, HAL olmadan register seviyesinde programlama yapabilirsiniz.