

# NuMicro<sup>®</sup> MCU MikroDenetleyici Eđitimi

M031/M480 Seviye 1 - Ders 3

Çeviren ve Anlatan: **Doç. Dr. Barış GÖKÇE**

Aralık 2024

**Not:** Bu doküman'ın telif hakkı tamamen Nuvoton'a aittir. Bu eğitim dokümanı mikrodenetleyici eğitim kapsamında Türkçe'ye orjinalinden çevrilmiştir. Dokümanın orijinali ingilizce'dir ve oluşan çeviri hataları tamamen Dr. Barış GÖKÇE'ye aittir.

Joy of innovation  
**nuvoTon**

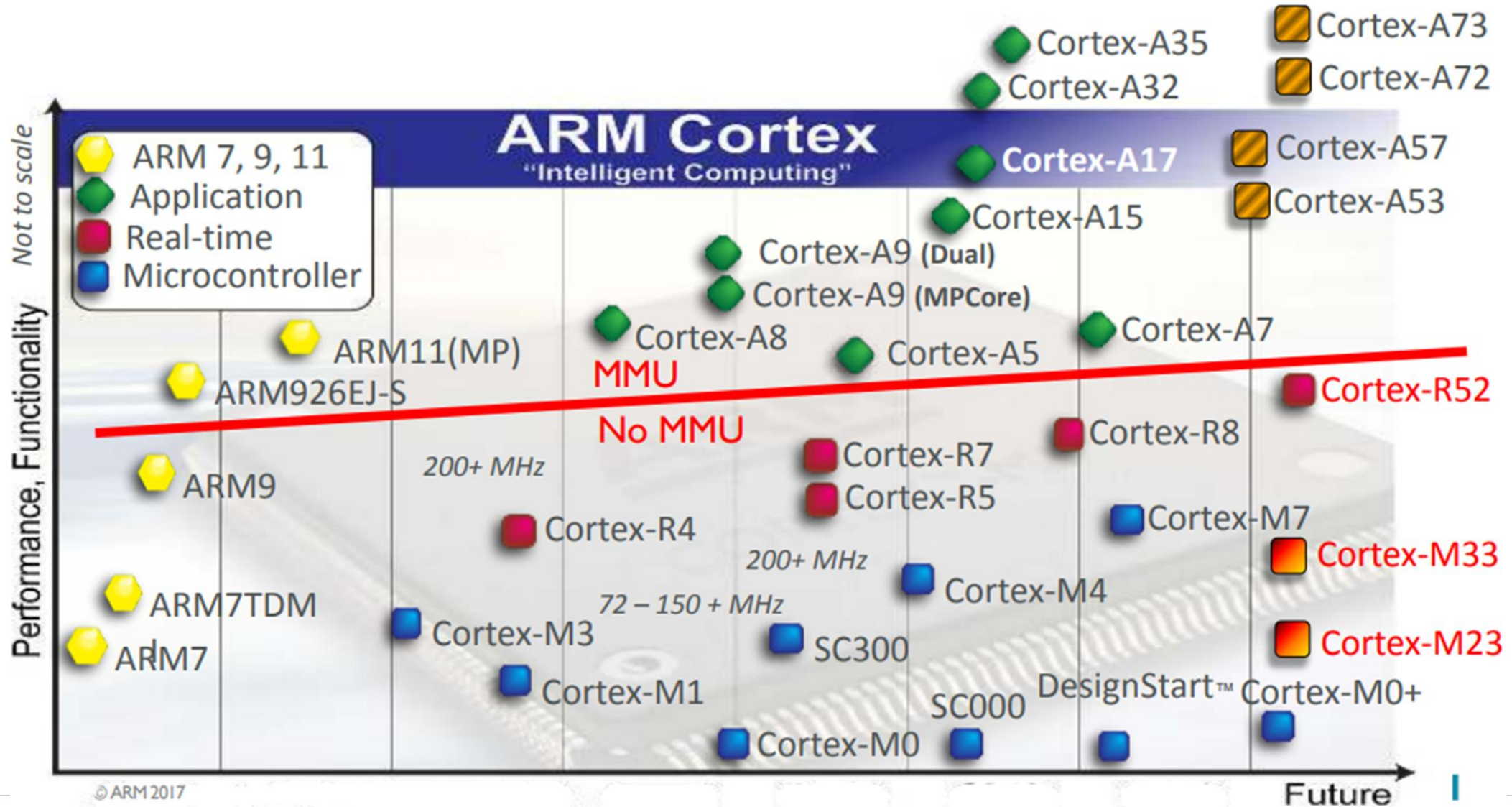
**özdisan**

# ARM Cortex®-M'e Genel Bakış





# ARM Mimarisi Yol Haritası

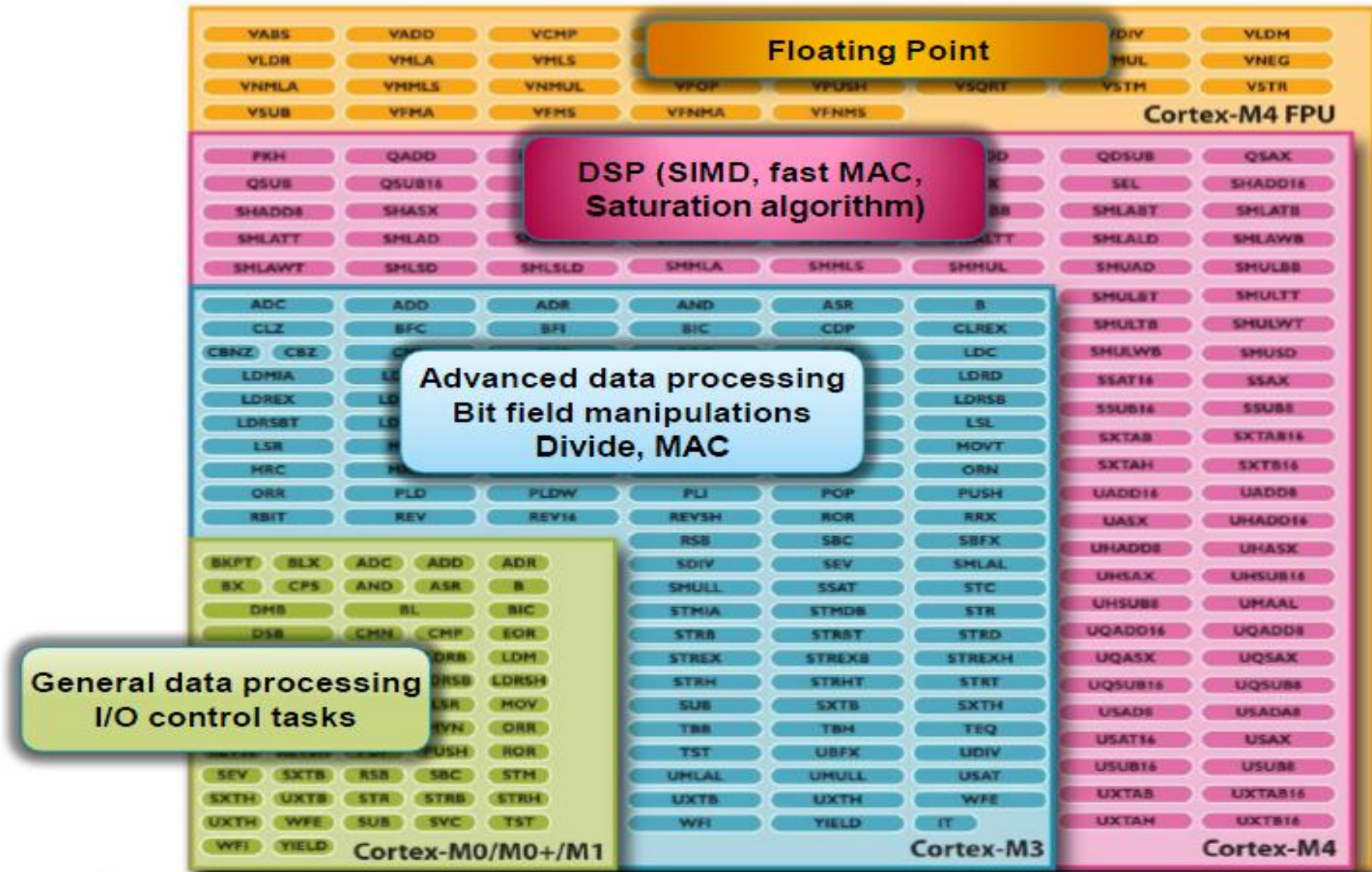


# Cortex-M İşlemci Ailesi (1/2)

Feature	Cortex-M0	Cortex-M23	Cortex-M4
Instruction set architecture	Armv6-M	Armv8-M Baseline	Armv7-M
	Thumb, Thumb-2	Thumb, Thumb-2	Thumb, Thumb-2
DMIPS/MHz range	0.87-1.27	0.99	1.25-1.95
CoreMark®/MHz	2.33	2.5	3.42
Memory Protection Unit (MPU)	No	Yes (option)	Yes (option)
Trace (ETM or MTB)	No	MTB (option) or ETMv3 (option)	ETMv3 (option)
DSP	No	No	Yes
Floating point hardware	No	No	Yes (option SP)
Systick Timer	Yes (option)	Yes (2 x)	Yes
TrustZone for Armv8-M	No	Yes (option)	No
Maximum # external interrupts	32	240	240
Hardware divide	No	Yes	Yes
Single cycle multiply	Yes (option)	Yes	Yes

Reference from ARM

# Cortex-M İşlemci Ailesi(2/2)





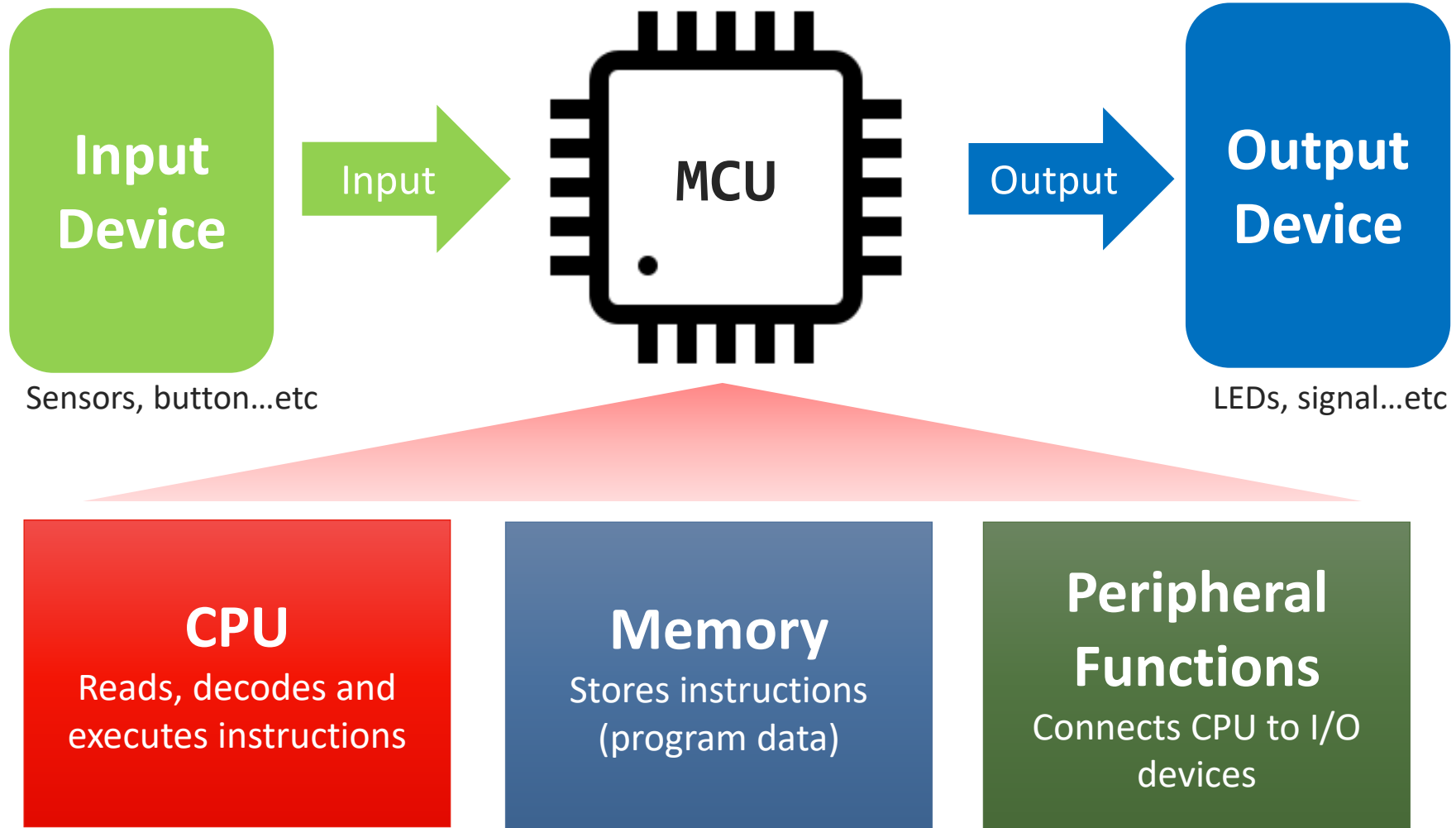
# NuMicro<sup>®</sup> Cortex<sup>®</sup>-M Mimarisi



# | Overview

- Sistem Güç Dağıtımı
- Saat Denetleyicisi
- Bellek Haritası
- Sistem Sıfırlama
- İç İçe Vektörlü Kesme Denetleyicisi (NVIC)
- 24 bit Sistem zamanlayıcısı (SysTick)

# | Mimariye genel bakış



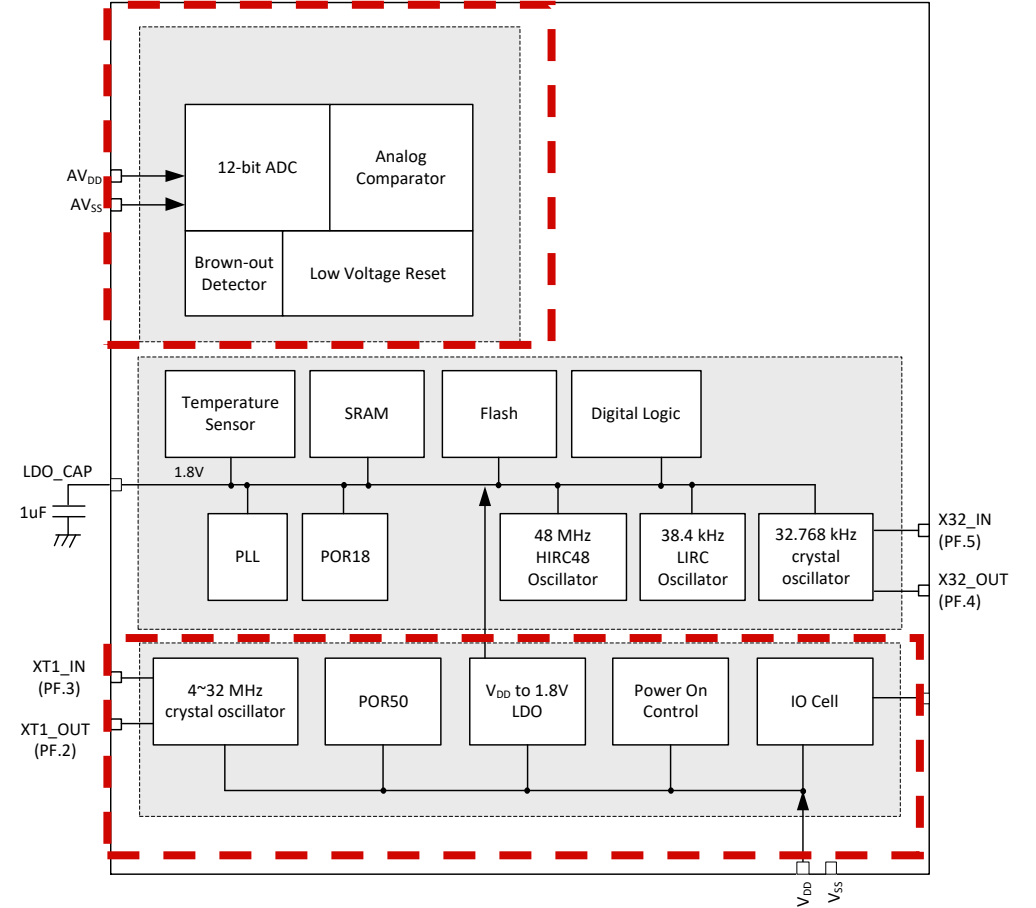


# Sistem Güç Dağıtımı (System Power Distribution)



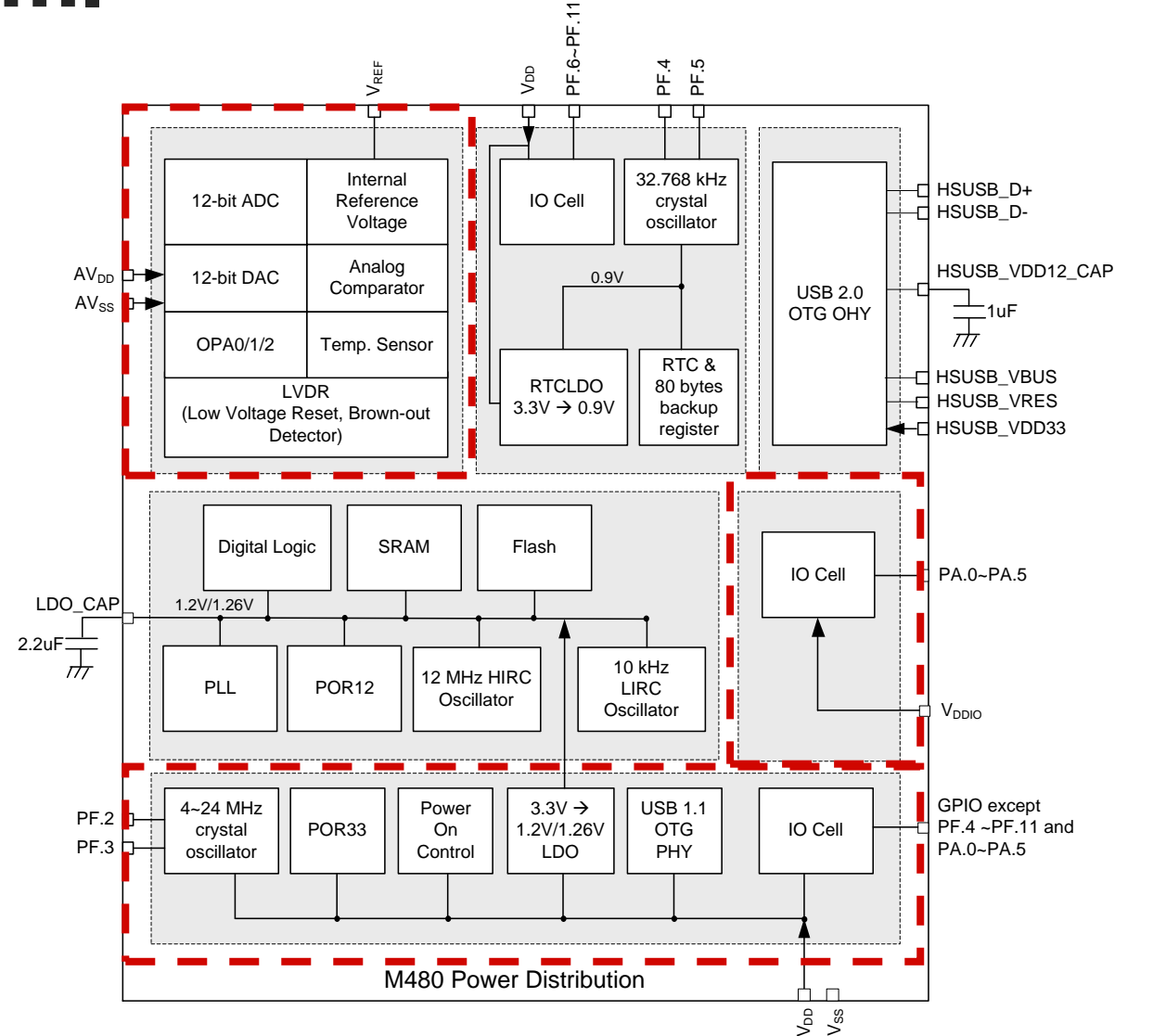
# M031 Sistem Güç Dağıtımı

- VDD = 1,8 ila 3,6 V: Harici Güç Kaynağı
- AVDD = 1,8 ila 3,6 V: Harici Analog Güç Kaynağı
- VDD ve AVDD için izin verilen voltaj farkı:  $\pm 50$  mV



# M480 Sistem Güç Dağıtımı

- VDD = 1,8 ila 3,6 V: Harici Güç Kaynağı
- AVDD = 1,8 ila 3,6 V: Harici Analog Güç Kaynağı
- VDDIO = 1,8 ila 3,6 V: Voltaj Ayarlama Arayüzü İçin
- VDD ve AVDD için izin verilen voltaj farkı:  $\pm 50$  mV





# | M031 Güç Modları

- Uyanma kaynağı (Wake-up source)
  - WDT, I<sup>2</sup>C, Timer, UART, USCI, BOD, GPIO, USB, and ACMP
- APIs
  - Bekleme modu (Idle mode): CLK\_Idle(void)
  - Güç kapatma modu: CLK\_PowerDown(void)

```
/* Enter idle mode*/  
CLK_Idle();  
/* Enter power down mode*/  
CLK_PowerDown();
```

Power mode	CPU	Clock	RAM	Peripheral	Power Consumption
Normal run	48 MHz	All	All	All	8.5 mA
Idle	Sleep	All	All	All	3.3 mA
Power down	Deep Sleep	LIRC/LXT	All	WDT/Timer/UART/GPIO	12 uA

# M480 Güç Modları

Power mode	CPU	Clock	RAM	Peripheral	Analog	Power consumption
Run (192MHz)	ON	All	All	All	All	34 mA
Idle	Sleep	All	All	All	All	10.32 mA
Fast wakeup Power Down	Deep Sleep	LIRC/LXT	All	RTC/WDT/Timer/ UART/GPIO	OPA/ACMP	0.49 mA
Power down	Deep Sleep	LIRC/LXT	All	RTC/WDT/Timer/ UART/GPIO	OPA/ACMP	0.37 mA
Low-leakage Power Down	Deep Sleep	LIRC/LXT	All	RTC/WDT/Timer/ UART/GPIO	OPA/ACMP	0.14 mA
Standby Power Down 0	Power off	LIRC/LXT	32K	RTC /GPIO	X	0.04 mA
Standby Power Down 1	Power off	LIRC/LXT	X	RTC /GPIO	X	0.03 mA
Deep Power Down	Power off	LIRC	X	RTC /GPIO	X	0.95 uA

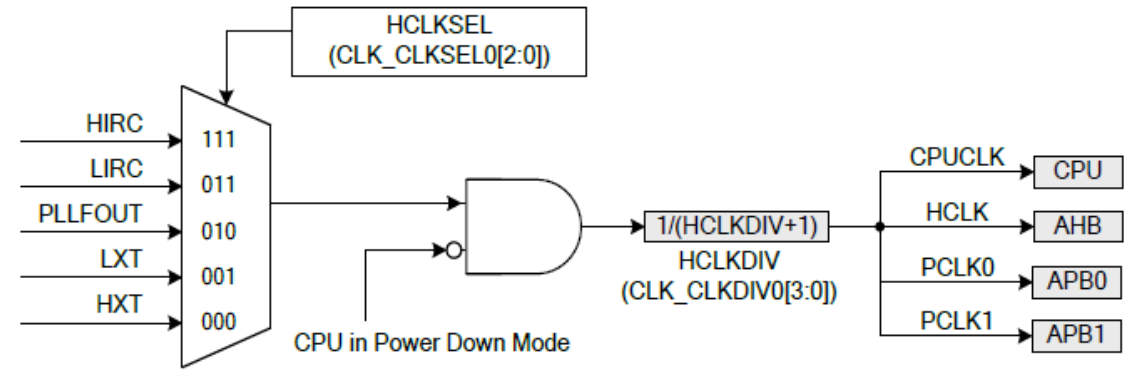
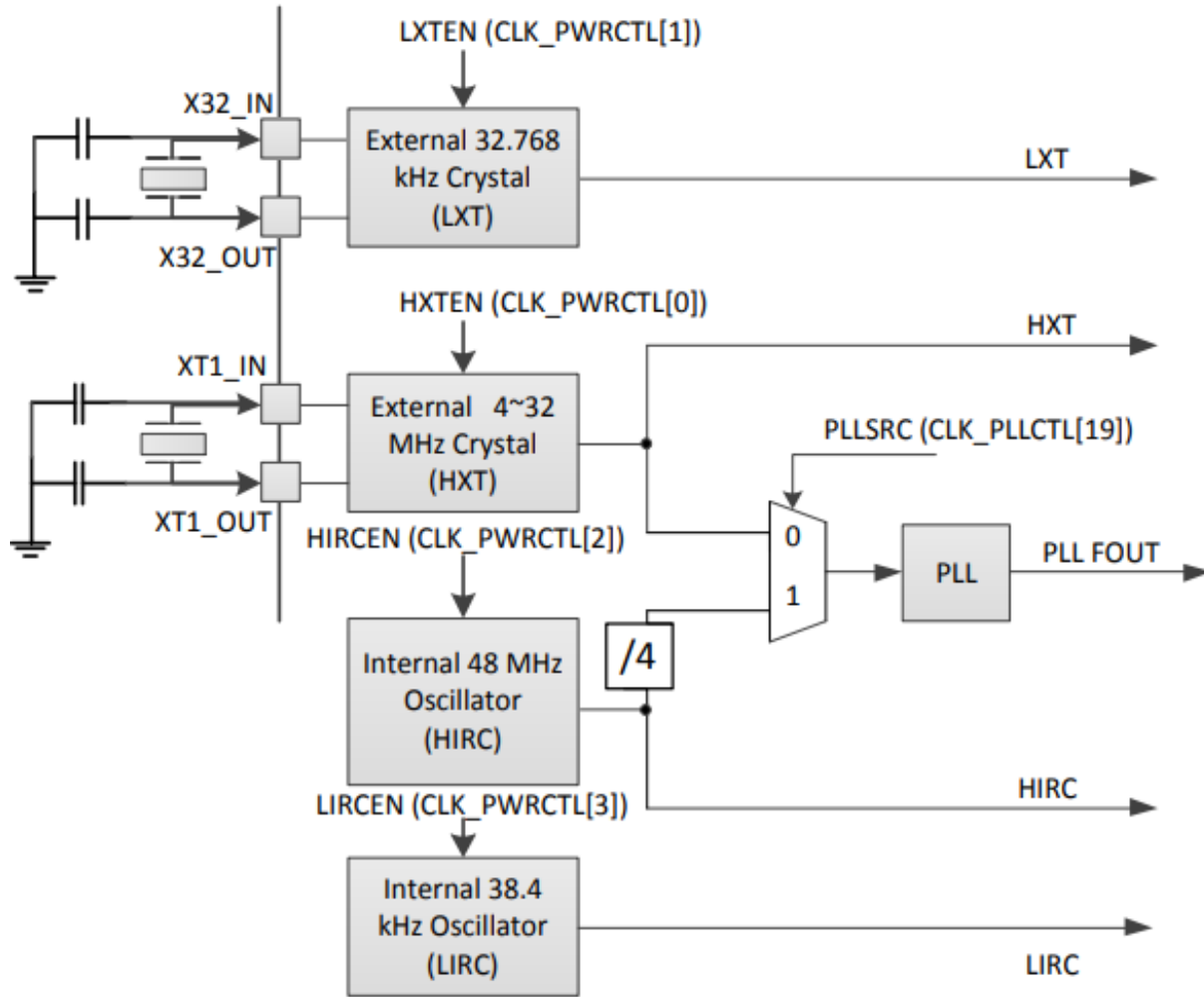
\*User must turn on LIRC before entering SPD0/1 mode.

# Saat Denetleyicileri (Clock Controller)

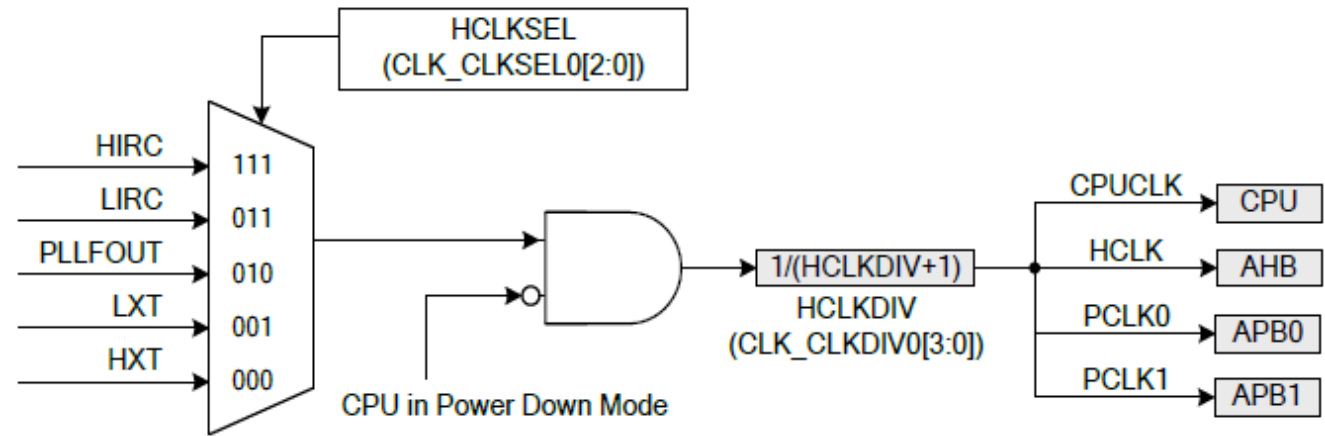
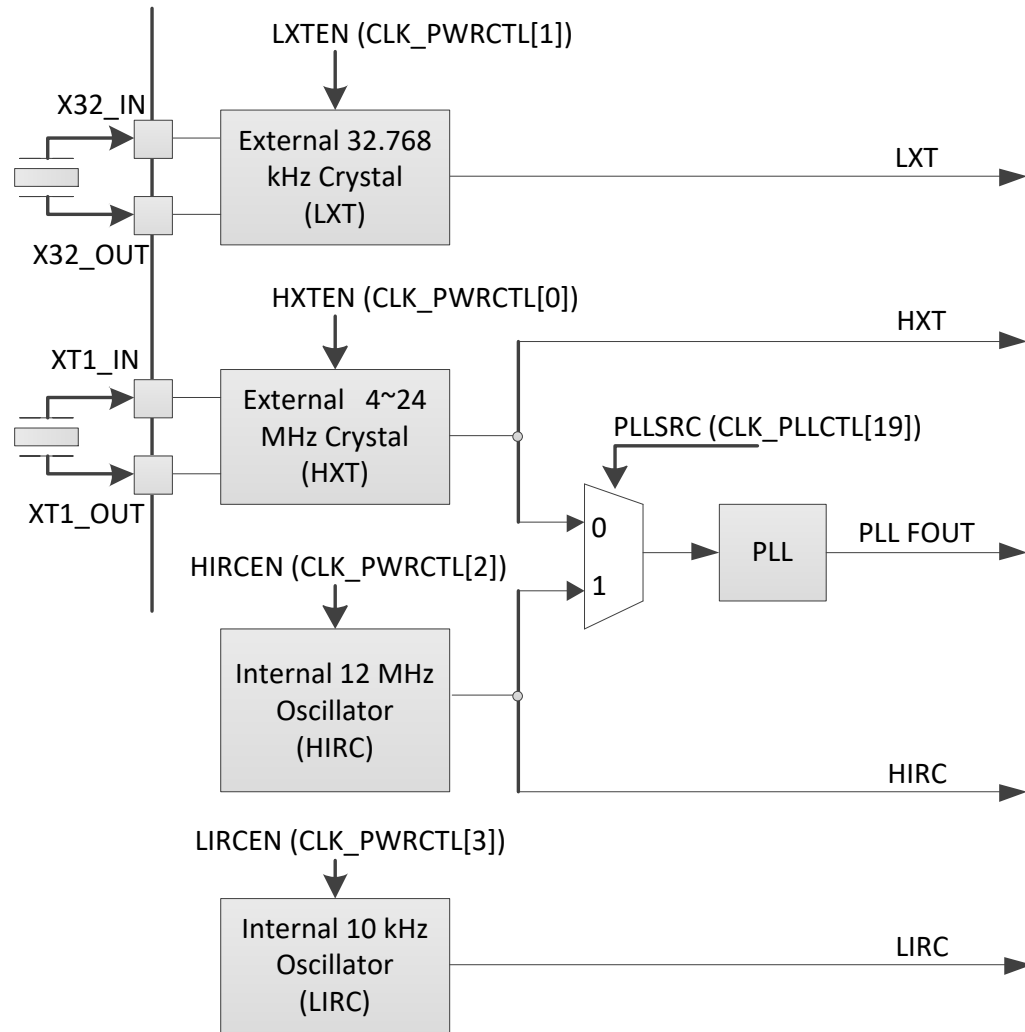




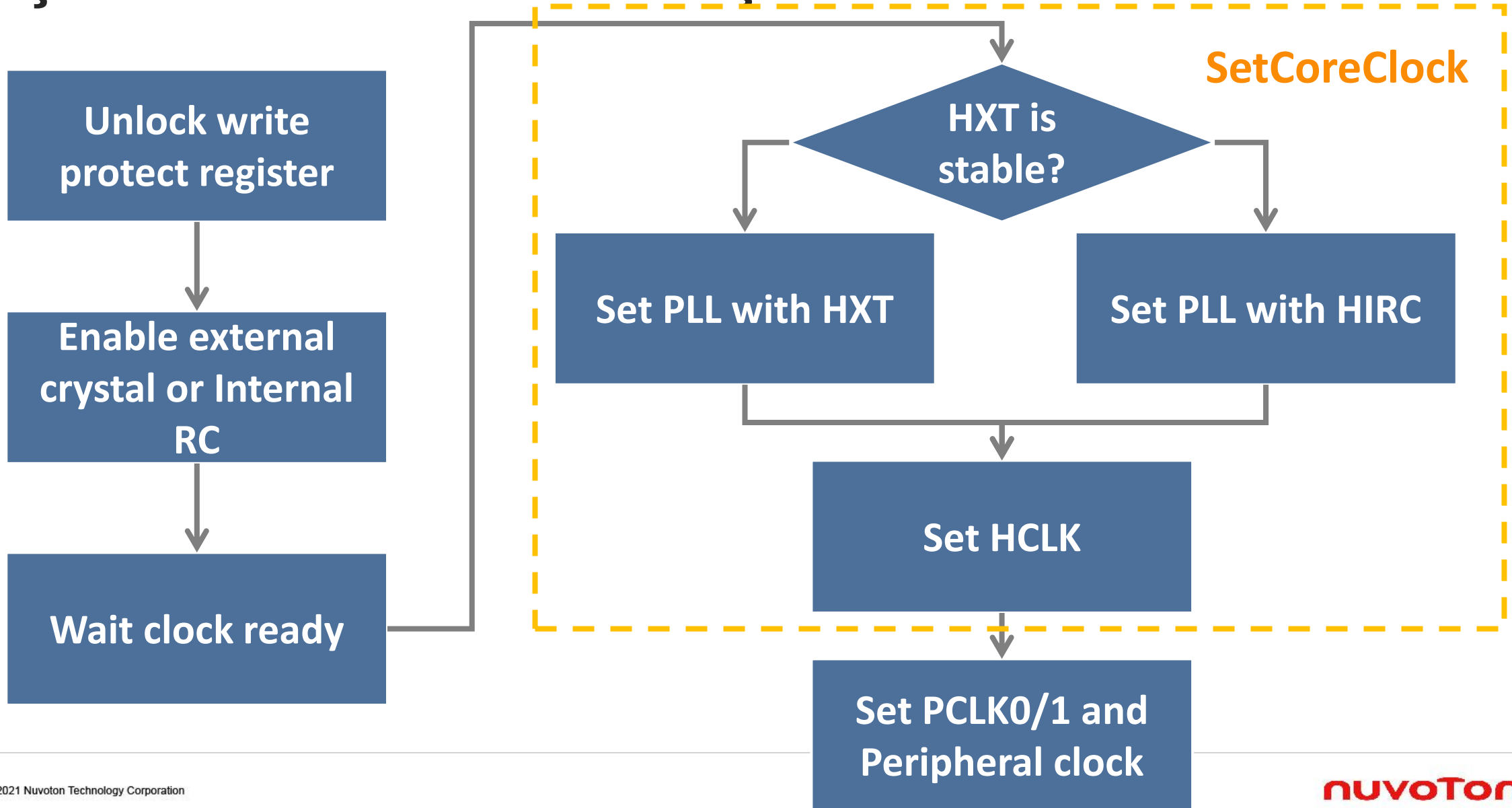
# M031 Saat (Clock) Diyagramı



# M031 Saat (Clock) Diyagramı



# Başlatma Sistemi Saat Akışı





# | Başlangıç (Initial Clock) Saat Fonksiyonları

## ◆ CLK\_EnableXtalRC()

```
void CLK_EnableXtalRC ( uint32_t u32ClkMask )
```

Enable clock source.

### Parameters

[in] **u32ClkMask** is clock source mask. Including :

- CLK\_PWRCTL\_HXTEN\_Msk
- CLK\_PWRCTL\_LXTEN\_Msk
- CLK\_PWRCTL\_HIRCEN\_Msk
- CLK\_PWRCTL\_LIRCEN\_Msk

### Returns

None

## ◆ CLK\_SetCoreClock()

```
uint32_t CLK_SetCoreClock ( uint32_t u32Hclk )
```

Set HCLK frequency.

### Parameters

[in] **u32Hclk** is HCLK frequency. The range of u32Hclk is 25.5MHz ~ 48MHz.

### Returns

HCLK frequency

```
/* Enable HXT clock (external XTAL 12MHz) */
CLK_EnableXtalRC(CLK_PWRCTL_HXTEN_Msk);
/* Set core clock */
CLK_SetCoreClock(FREQ_48MHZ);
```

# Modül Saat Fonksiyonlarını Set Etme

## ◆ CLK\_SetModuleClock()

```
void CLK_SetModuleClock ( uint32_t u32ModuleIdx,  
                          uint32_t u32ClkSrc,  
                          uint32_t u32ClkDiv  
                          )
```

This function set selected module clock source and module clock divider.

### Parameters

[in] **u32ModuleIdx** is module index.  
[in] **u32ClkSrc** is module clock source.  
[in] **u32ClkDiv** is module clock divider.

### Returns

None

## ◆ CLK\_EnableModuleClock()

```
void CLK_EnableModuleClock ( uint32_t u32ModuleIdx )
```

This function enable module clock.

### Parameters

[in] **u32ModuleIdx** is module index. Including :

- PDMA\_MODULE
- ISP\_MODULE
- EBI\_MODULE
- HDIV\_MODULE
- CRC\_MODULE
- WDT\_MODULE
- WWDT\_MODULE
- TMR0\_MODULE
- TMR1\_MODULE

```
/* Set module clock */
```

```
CLK_SetModuleClock(UART0_MODULE, CLK_CLKSEL1_UART0SEL_HXT, CLK_CLKDIV0_UART0(1));
```

```
/* Enable module clock */
```

```
CLK_EnableModuleClock(UART0_MODULE);
```

# I API - SYS\_Init()

```
/* Korunan kayıtları kilidini aç (Unlock protected registers) */
SYS_UnlockReg();
/* HXT saatini etkinleştir (Enable HXT clock) (external XTAL 12MHz) */
CLK_EnableXtalRC(CLK_PWRCTL_HXTEN_Msk);
/* HXT saatinin hazır olmasını bekleyin */
CLK_WaitClockReady(CLK_STATUS_HXTSTB_Msk);

/* Çekirdek saatini PLL'den PLL_CLOCK olarak Set Edin Set core clock as PLL_CLOCK from PLL */
CLK_SetCoreClock(FREQ_48MHZ);
/* PCLK0/PCLK1'i HCLK/2'ye ayarlayın (Set PCLK0/PCLK1 to HCLK/2) */
CLK->PCLKDIV = (CLK_PCLKDIV_APB0DIV_DIV2 | CLK_PCLKDIV_APB1DIV_DIV2);

/* Modül saatini etkinleştir (Enable module clock) */
CLK_EnableModuleClock(UART0_MODULE);
/* Modül saatini Set Edin, Set module clock */
CLK_SetModuleClock(UART0_MODULE, CLK_CLKSEL1_UART0SEL_HXT, CLK_CLKDIV0_UART0(1));
/* Lock protected registers */
SYS_LockReg();
```



# Hafıza Haritası (Memory Map)



# **| M031 Sistem Hafıza Haritası (Memory Map)**

	<b>Reserved</b>
0x0030_000B	<b>User Configuration (12 B)</b>
0x0030_0000	
	<b>Reserved</b>
0x0020_01FF	<b>Security Protection Memory (512 B SPROM)</b>
0x0020_0000	
	<b>Reserved</b>
0x0010_1FFF	<b>Loader Program Memory (8 KB LDROM)</b>
0x0010_0000	
	<b>Reserved</b>
0x0001_FFFF	<b>Application Program Memory (128 KB APROM)</b>
0x0000_0000	

# | M480 Sistem Hafıza Haritası (Memory Map)

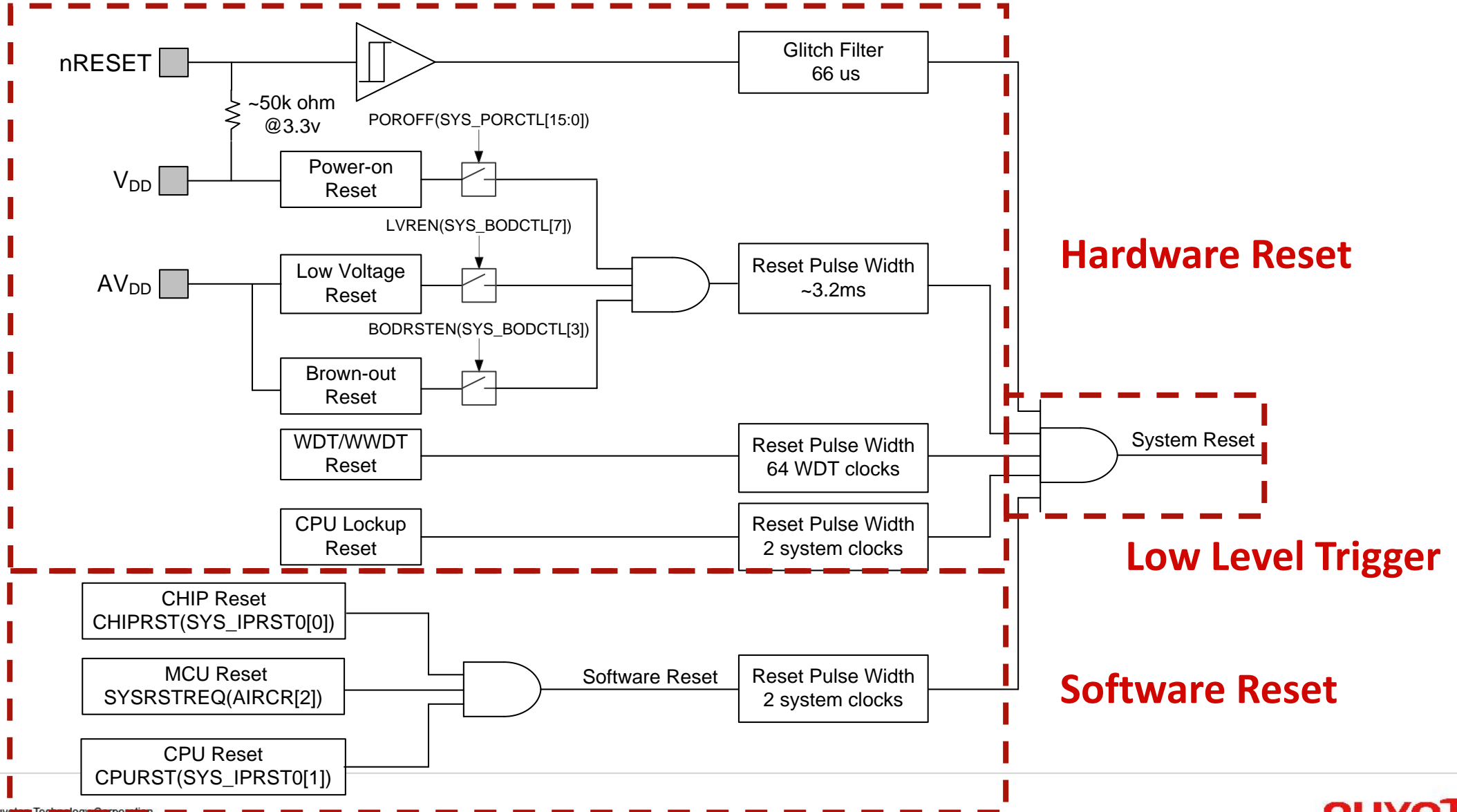
0x0080_7FFF 0x0080_0000	Boot Loader (32 KB)
	Reserved
0x0031_0BFF 0x0031_0000	OTP (3 KB)
	Reserved
0x0030_2FFF 0x0030_1000	Key Protection ROM (8 KB KPROM)
	Reserved
0x0030_000F 0x0030_0000	User Configuration (16 B)
	Reserved
0x0020_0FFF 0x0020_0000	Security Protection Memory (4 KB SPROM)
	Reserved
0x0010_0FFF 0x0010_0000	Loader Program Memory (4 KB LDROM)
	Reserved
0x0007_FFFF 0x0000_0000	Application Program Memory (512 KB APROM)

# Sistem Reset





# Sistem Reset



# | Sistem Reset

## Donanimsal (Hardware) Reset

- The Power-On Reset
- nRESET pin Reset
- Watchdog Time Out Reset
- Low Voltage Detected Reset
- Brown-Out-Detected Reset
- CPU Lock up Reset

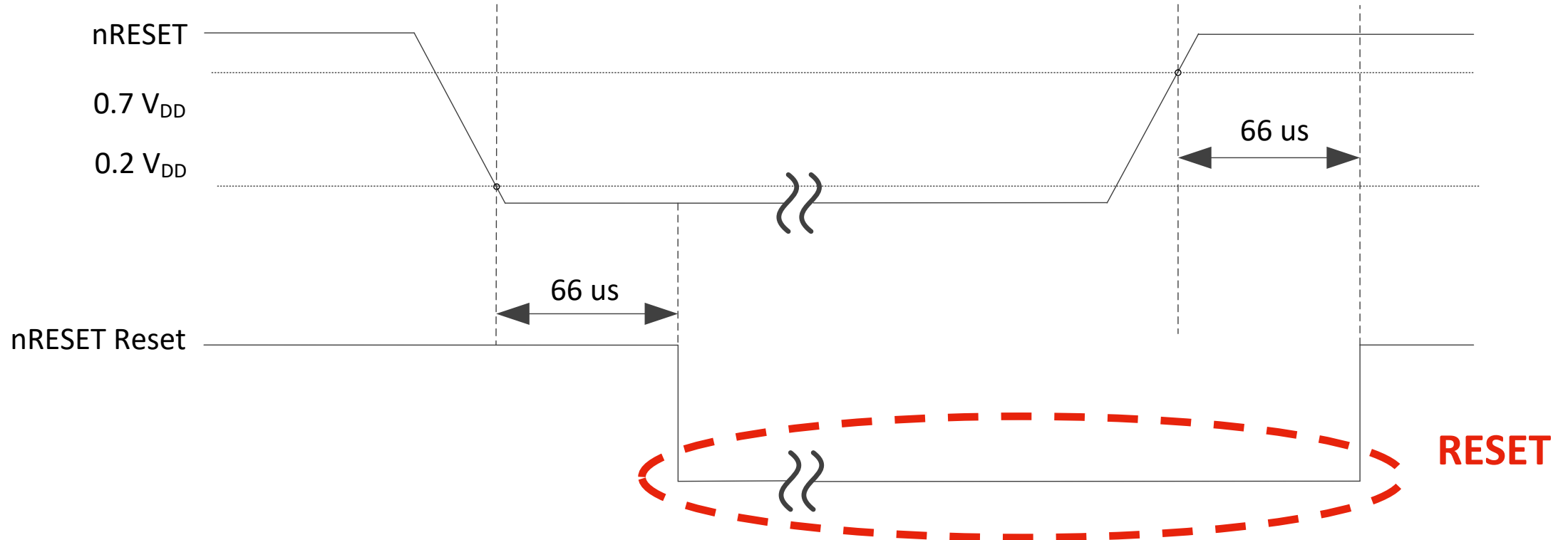
## Yazılımsal (Software) Reset

- CPU\_RST
  - Just only reset CPU & flash controller
  - `SYS_UnlockReg(); SYS_ResetCPU();`
- MCU\_RST
  - To reset the whole chip
  - `SYS_UnlockReg(); NVIC_SystemReset();`
- CHIP\_RST
  - To reset the whole chip like “Power-on reset”
  - `SYS_UnlockReg(); SYS_ResetChip();`

“RSTSTS” register identify chip’s reset source from last operation

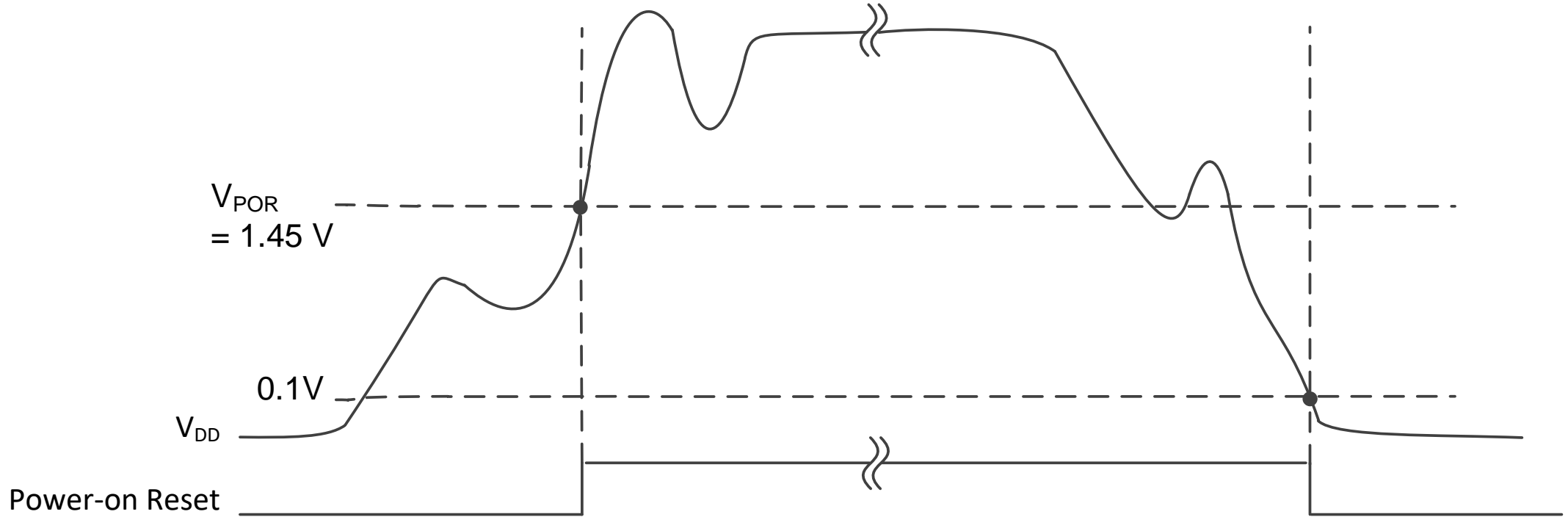
# | nRESET Reset

- nRESET reset, asenkron reset giriş pini olan ve sistemi herhangi bir anda resetlemek için kullanılabilen nRESET pinini düşüğe çekerek reset sinyali üretmek anlamına gelir.



# | Power-on Reset (POR)

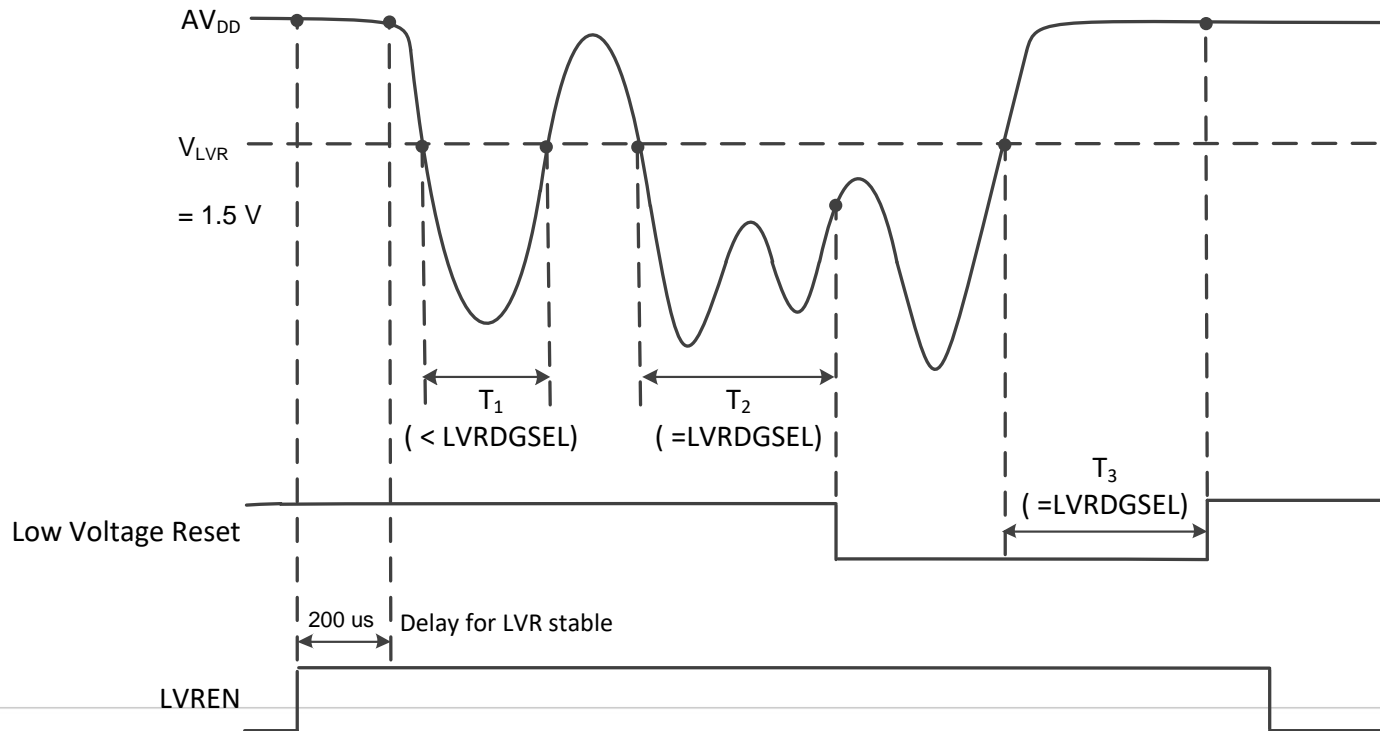
- Güç açılırken sıfırlama (POR), kararlı bir sistem sıfırlama sinyali oluşturmak için kullanılır ve MCU'nun beklenmedik davranışlarını önlemek için güç açıldığında sistemin sıfırlanmasını zorlar.





# Low Voltage Reset(LVR)

- Düşük Voltaj Sıfırlama (LVR) fonksiyonu sistem çalışması sırasında  $AV_{DD}$ 'yi algılar.  $AV_{DD}$  voltajı  $V_{LVR}$ 'den düşük olduğunda, çip sıfırlanacaktır.

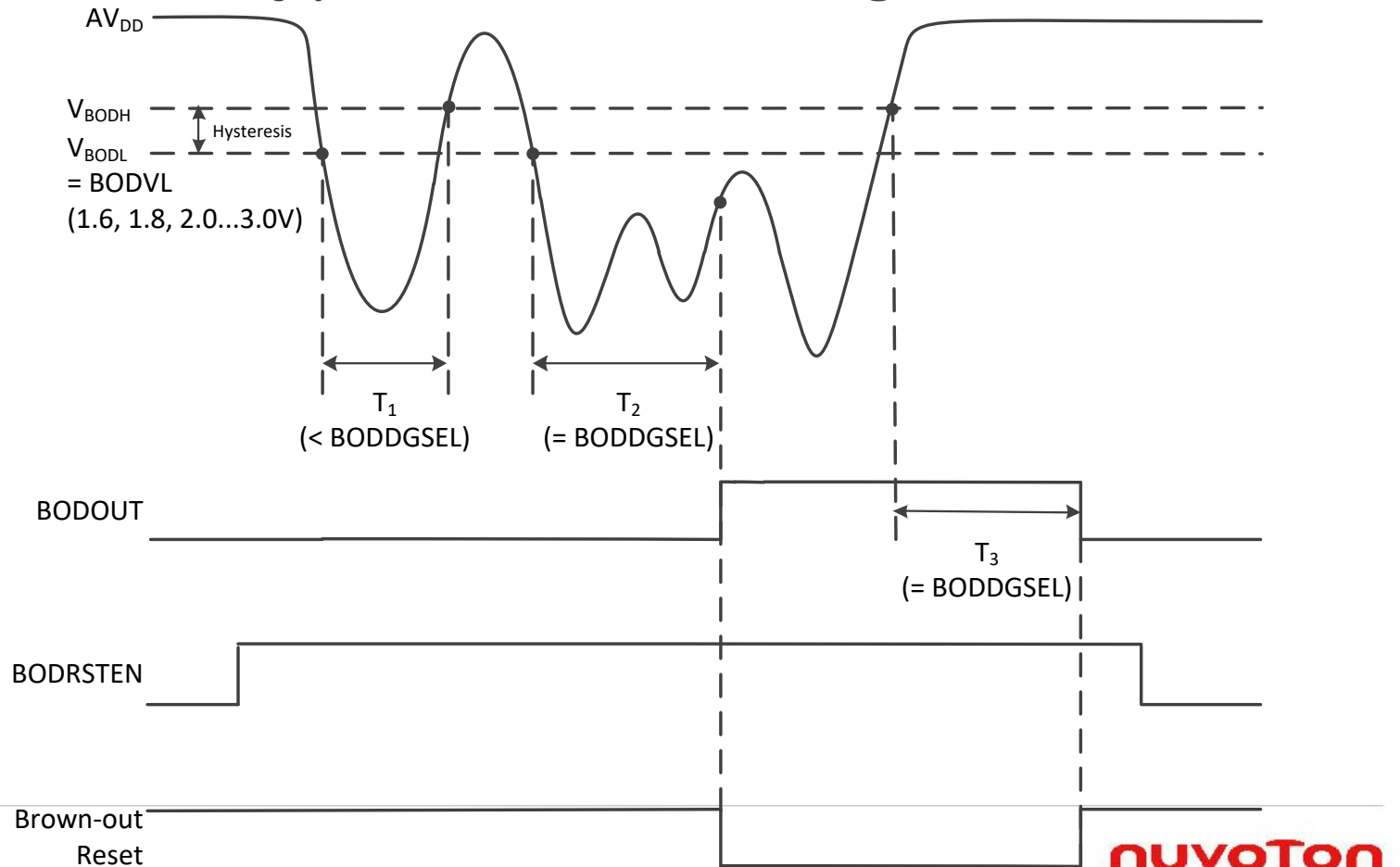


LVRDGSEL	De-glitch Time
000	Without de-glitch
001	4 system clock
010	8 system clock
011	16 system clock
100	32 system clock
101	64 system clock
110	128 system clock
111	256 system clock

# | Brown-out Detector Reset (BOD)

- Brown-out Dedektörü Sıfırlama (BOD), güç kaynağı voltajında önemli bir düşüş olan brown-out durumunda çipin sıfırlanmasını sağlar.

BODDGSSEL	De-glitch Time
000	Sampled by RC10K clock
001	4 system clock
010	8 system clock
011	16 system clock
100	32 system clock
101	64 system clock
110	128 system clock
111	256 system clock



# | Watchdog Timer Reset (WDT)

- Çoğu endüstriyel uygulamada sistem güvenilirliği çok önemlidir. MCU'yu arıza durumundan otomatik olarak kurtarmak, sistem güvenilirliğini artırmanın bir yoludur.
- Watchdog zamanlayıcısı (WDT), sistemin düzgün çalışıp çalışmadığını kontrol etmek için yaygın olarak kullanılır. MCU çökerse veya kontrolden çıkarsa, watchdog zaman aşımına neden olabilir.

# **CPU Kilitlenmesi Sıfırlama (Lockup Reset)**

- CPU, işlemcinin yerleşik sistem durumu koruma donanımını etkinleştirmek için kurtarılamaz bir istisna olayı nedeniyle kilitlendi.
- CPU, donanım hatası işleyicisinde donanım hatası ürettikten sonra kilitleme durumuna girer. Ve ardından CPU Kilitleme Sıfırlama işlevi çipin sıfırlanmasına neden olur.
- Çip hata ayıklama moduna girdiğinde, CPU kilitleme sıfırlaması yok sayılır.



# | Çevrebirimi Sıfırlama (Peripheral IP Reset)

- Her çevre biriminin bir sıfırlama kaydı (reset register) vardır
- “**SYS\_IPRST0~SYS\_IPRST2**” kaydı çevre birimi asenkron sıfırlama sinyalini tanımlamıştır
- API: SYS\_ResetModule(uint32\_t u32ModuleIndex)

```
/* Reset selected module */  
SYS_ResetModule(TMR0_RST);
```

# İç içe Vektörlü Kesme Denetleyicisi

(Nested Vectored  
Interrupt Controller)



# | NuMicro NVIC (İç İçe Vektörlü Kesme Denetleyicisi)

- NVIC (İç İçe Vektörlü kesme Denetleyicisi)
  - Cortex-M işlemcisinin entegre bir parçasıdır
  - Kesme sinyallerinin Seviye ve darbe algılamasını destekler.
  - NMI (Maskelenemeyen Kesme) girişini destekler
  - “Kuyruk Zincirleme” ve “Geç Varış”ı destekler
  - M0 : Her biri 4 öncelik seviyesine sahip 32 harici kesme girişi
  - M4 : Her biri 16 öncelik seviyesine sahip 240 harici kesme girişi

# I NVIC (İç İçe Vektörlü Kesme Denetleyicisi)

- Kuyruk zincirleme
  - Yığın Pop'unu atla ve yeni kesmeyi hemen servis et. (Skip the stack Pop and services the new interrupt immediately).

## **Interrupt Chaining (kesme Zincirleme) Nedir?**

Bir kesme servis rutini (ISR) içinde başka bir kesmeyi tetikleme işlemidir. Yani, bir kesme işlenirken, bu ISR içinde daha düşük öncelikli bir başka kesmeyi etkinleştirerek, ardışık bir kesme zinciri oluşturulur.

# I NVIC Önceliği (priority)

- NVIC\_EnableIRQ(IRQn\_Type IRQn)
  - Enable External Interrupt

```
/* Enable interrupt */  
NVIC_EnableIRQ(GPIO_PAPB_IRQn);
```

- NVIC\_SetPriority(IRQn\_Type IRQn, uint32\_t priority)
  - Set Interrupt Priority
  - Priority can be M0: 0~3, M4: 0~15

```
/* Enable HXT clock (external XTAL 12MHz) */  
NVIC_SetPriority(GPIO_PAPB_IRQn, 1)
```



# IRQ Fonksiyon Adı

In “startup\_xxxx.s” vector table address  
save peripherals **IRQ** function address

```
startup_M480.s
67
68      ; External Interrupts
69      DCD      BOD_IRQHandler      ; 0: Brown Out detection
70      DCD      IRC_IRQHandler      ; 1: Internal RC
71      DCD      PWRWU_IRQHandler    ; 2: Power down wake up
72      DCD      RAMPE_IRQHandler    ; 3: RAM parity error
73      DCD      CKFAIL_IRQHandler   ; 4: Clock detection fail
74      DCD      Default_Handler     ; 5: Reserved
75      DCD      RTC_IRQHandler       ; 6: Real Time Clock
76      DCD      TAMPER_IRQHandler    ; 7: Tamper detection
77      DCD      WDT_IRQHandler       ; 8: Watchdog timer
78      DCD      WWDT_IRQHandler      ; 9: Window watchdog timer
79      DCD      EINT0_IRQHandler     ; 10: External Input 0
80      DCD      EINT1_IRQHandler     ; 11: External Input 1
81      DCD      EINT2_IRQHandler     ; 12: External Input 2
82      DCD      EINT3_IRQHandler     ; 13: External Input 3
83      DCD      EINT4_IRQHandler     ; 14: External Input 4
84      DCD      EINT5_IRQHandler     ; 15: External Input 5
85      DCD      GPA_IRQHandler       ; 16: GPIO Port A
86      DCD      GPB_IRQHandler       ; 17: GPIO Port B
87      DCD      GPC_IRQHandler       ; 18: GPIO Port C
88      DCD      GPD_IRQHandler       ; 19: GPIO Port D
89      DCD      GPE_IRQHandler       ; 20: GPIO Port E
--      ---      ---      ---      --
```

# | Sistem Zamanlayıcısı (system timer) - SysTick

- A 24-bit azalan (decrementing), sıfıra sarma (wrap-on-zero) sayma.

## ◆ CLK\_SysTickDelay()

```
__STATIC_INLINE void CLK_SysTickDelay ( uint32_t us )
```

This function execute delay function.

### Parameters

[in] **us** Delay time. The Max value is  $2^{24} / \text{CPU Clock(MHz)}$ . Ex: 50MHz => 335544us, 48MHz => 349525us, 28MHz => 699050us ...

### Returns

None

```
/* Delay 10 ms */  
CLK_SysTickDelay(10000);
```

*Joy of innovation*  
**nuvoTon**

Thank You

Danke

Merci

ありがとう

Gracias

Kiitos

감사합니다

धन्यवाद

كل ارکش

הודות