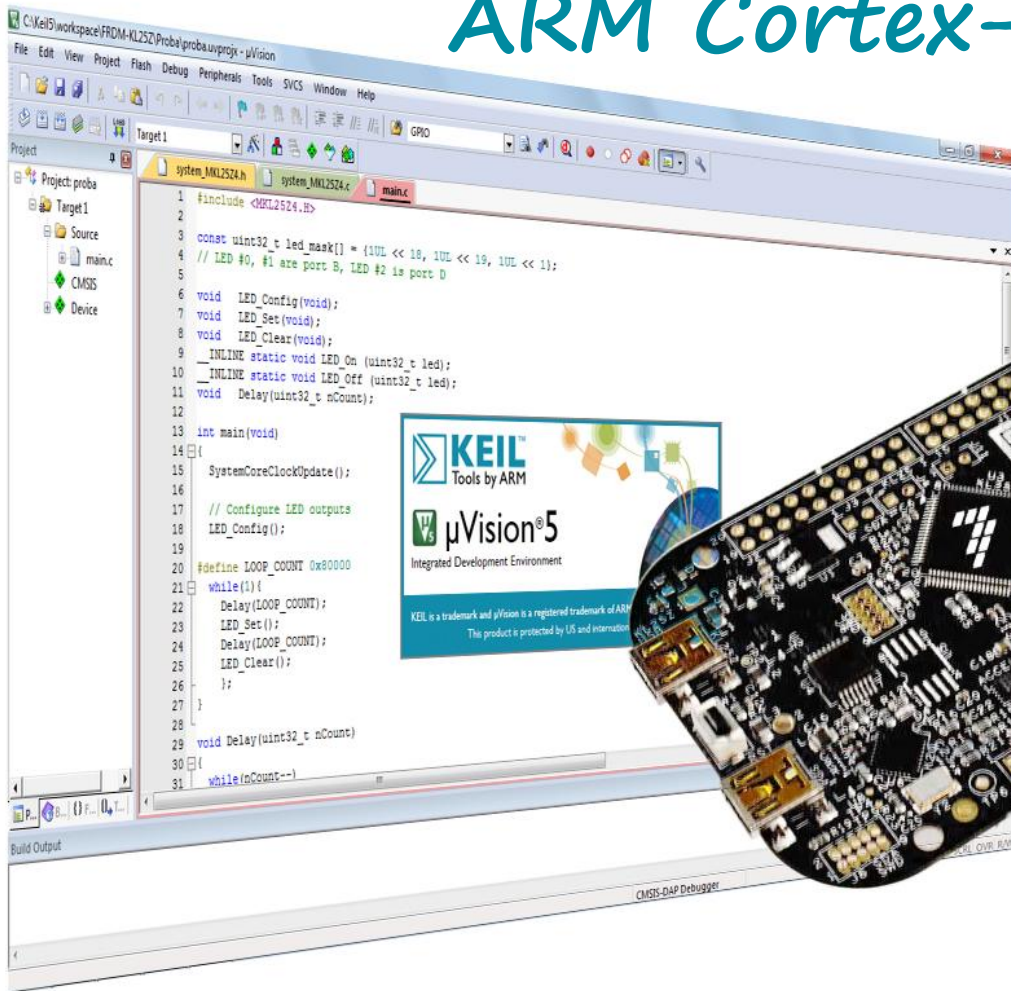


ARM Cortex-M0+ mikrovezérlő programozása KEIL MDK 5 környezetben



Cortex
Intelligent Processors by ARM®

11. Impulzus-szélesség moduláció (PWM)

Felhasznált anyagok, ajánlott irodalom

- ❑ Muhammad Ali Mazidi, Shujen Chen, Sarmad Naimi, Sepehr Naimi: **Freescale ARM Cortex-M Embedded Programming**
- ❑ ARM University Program: **Course/Lab Material for Teaching Embedded Systems/MCUs** (for the FRDM-KL25Z board)
- ❑ Trevor Martin: **The Designer's Guide to the Cortex-M Processor Family**
- ❑ Freescale: [MKL25Z128VLK4 MCU datasheet](#)
- ❑ Freescale: [KL25 Sub-Family Reference Manual](#)
- ❑ Freescale: [FRDM-KL25Z User Manual](#)

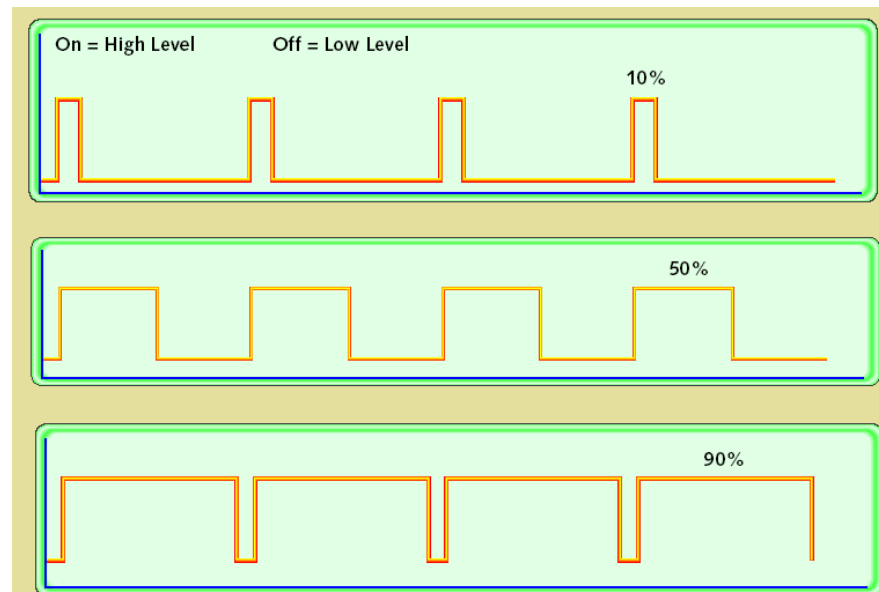
Impulzus-szélesség moduláció (PWM)

□ PWM alkalmazások

- ❖ **Digitális teljesítmény-fokozatot** – jobb hatásfokúak és olcsóbbak, mint az analóg erősítők
 - Alkalmazások: motor sebességvezérlés, fényerősség szabályozás, kapcsolóüzemű tápegységek
 - Elve: a fogyasztó (motor, világítás, stb.) lassan reagál, kiátlagolja a PWM jelet.
- ❖ **Digitális kommunikáció** kevésbé érzékeny a zajokra, mint az analóg átvitel
 - PWM az analóg jel digitálisan kódolt formája (pl. RC szervo vezérlés)
 - Kevésbé érzékeny a zajokra

□ PWM jel jellemzői

- ❖ **Modulációs frekvencia** – hány impulzus esik egy másodpercre (fix)
- ❖ **Periódus idő** – $1/(\text{modulációs frekvencia})$
- ❖ **Bekapcsolási idő** – az időtartam, amíg az impulzus aktív állapotú (beállítható)
- ❖ **Kitöltés** – Bekapcsolási idő/Periódus idő
- ❖ A bekapcsolási idő (a kitöltés) változása képviseli az analóg információt.



Általános célú időzítők (TPM)

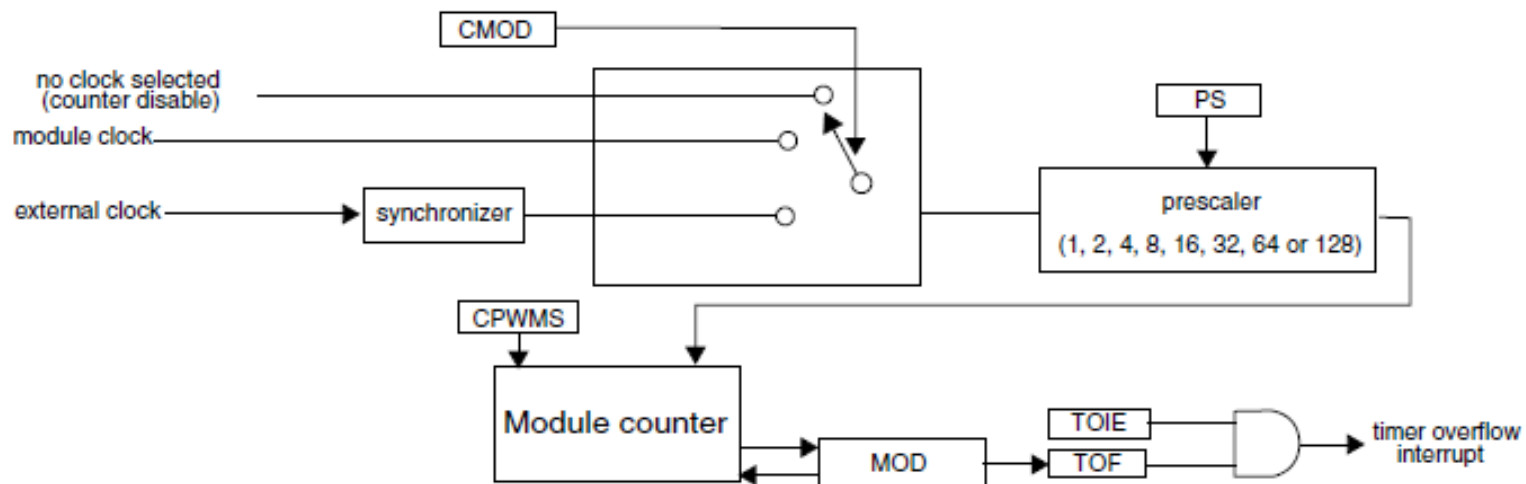
Az **MKL25Z128VLK4** mikrovezérlő 3 db általános felhasználású időzítő/számláló egységgel rendelkezik: **TPM0**, **TPM1**, **TPM2**, nevük a **Timer/PWM Module** elnevezés rövidítése.

Részletes leírás: [Freescale KL25 Sub Family Reference Manual](#) 31. fejezetében.

Mindegyik számlálóhoz több impulzus-szélesség modulációra (PWM), bemeneti jelrögzítésre (Input Capture), vagy digitális komparálásra (Output Compare) használható csatorna is kapcsolódik: **TPM0** 6 db, **TPM1** és **TPM2** pedig két-két csatornával rendelkezik.

Egy **TPM** számláló/időzítő részegységének vázlata az alábbi ábrán látható.

A főbb összetevők: a bemeneti jelválasztó, az előszámláló, a modulo számláló, a modulo adatregiszter, valamint a túlcscordulást jelző és megszakításkérő áramkörök.



A TPMx modulok regiszterkészlete

TPM0 báziscíme: 0x4003 8000

TPM1 báziscíme: 0x4003 9000

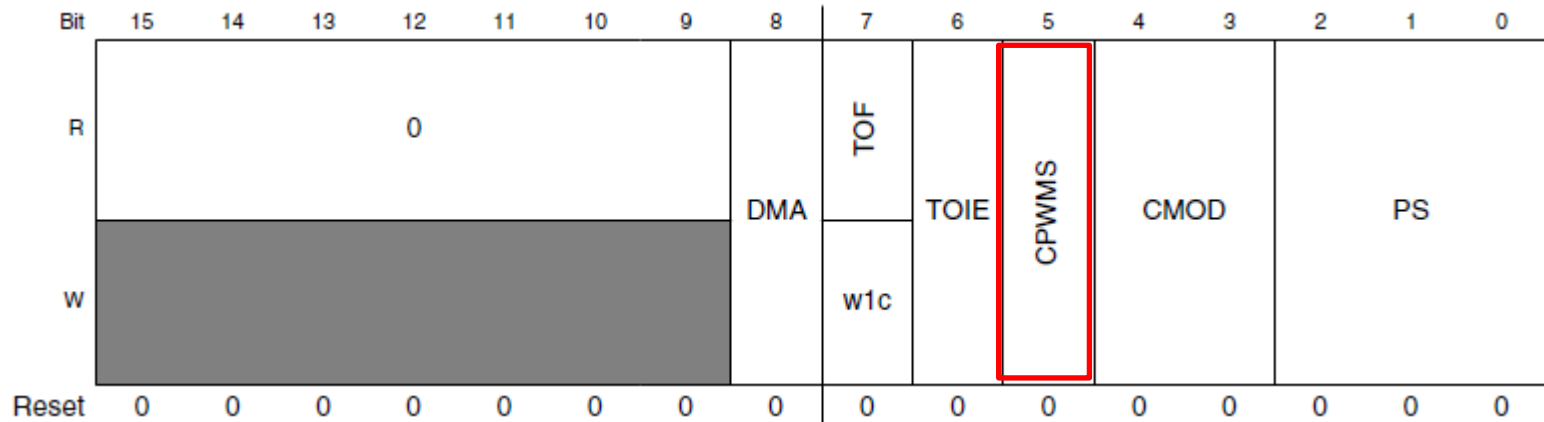
TPM2 báziscíme: 0x4003 A000

A **TPM0** modul regiszterei

Timer regiszterek

Regiszternév	A regiszter funkciója	Cím
TPM0_SC	Status and Control (állapotjelző és vezérlő regiszter)	4003_8000
TPM0_CNT	Counter (számláló regiszter)	4003_8004
TPM0_MOD	Modulo (számlálási határ)	4003_8008
TPM0_C0SC	Channel 0 Status and Control (0. csatorna állapotjelző és vezérlő)	4003_800C
TPM0_C0V	Channel 0 Value (0. csatorna adatregiszter)	4003_8010
TPM0_C1SC	Channel 1 Status and Control (1. csatorna állapotjelző és vezérlő)	4003_8014
TPM0_C1V	Channel 1 Value (1. csatorna adatregiszter)	4003_8018
...	•Csatornavezérlő regiszterek	...
TPM0_STATUS	Capture and Compare Status (Jelrögzítés és komparálás státusza)	4003_8050
TPM0_CONF	Configuration (Konfiguráció)	4003_8084

TPMx_SC: állapotjelző és vezérlő



Csak a legalsó 9 bitet használjuk. A **TOF** jelzőbitet '1' írásával lehet törölni.

A bitek szerepe:

DMA - DMA átvitel engedélyezése a TOF jelzőbitre

TOF - Timeout jelzőbit (TOF = 1, amikor a számláló a MOD értéket meghaladja)

TOIE - Megszakításkérés engedélyezése (**0**: megszakítás tiltva, **1**: megszakítás engedélyezve)

CPWMS - Szimmetrikus (Center-aligned) PWM mód választása (**0**: a számláló felfelé számlál, **1**: a számláló fel/le számlál)

CMOD - Órajel választás (**00**: számláló letiltva, **01**: belső órajel, **10**: külső órajel, **11**: fenntartva)

PS - Előosztó osztási arány választás: 000 Divide by 1, 001 Divide by 2, 010 Divide by 4, 011 Divide by 8, 100 Divide by 16, 101 Divide by 32, 110 Divide by 64, 111 Divide by 128

Fölfelé számlálás

Modulo számlálás fölfelé: **CPWMS = 0** esetén a számlálás a **TPMx_MOD** regiszterben megadott értékig történik, utána nullára vált a számláló és '1'-be áll a TOF bit.

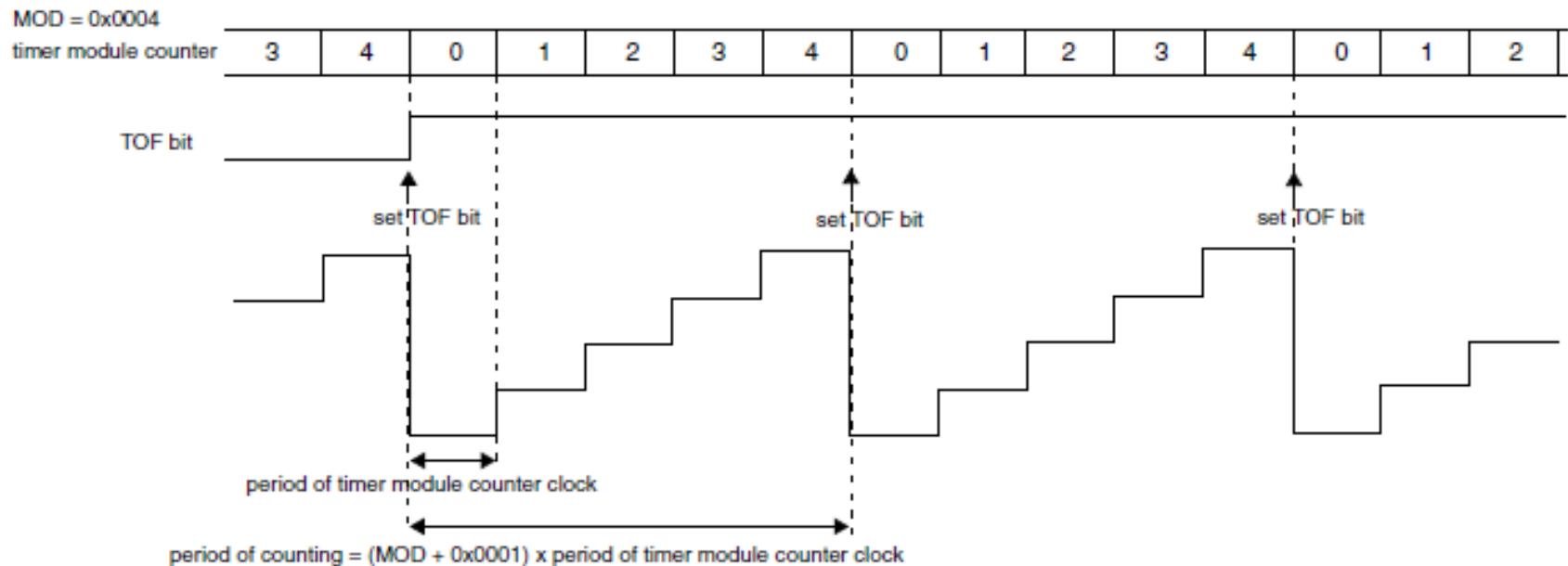


Figure 31-79. Example of TPM Up Counting

Számlálás fel- le

Modulo számlálás fel és le: **CPWMS = 1** esetén a felfelé számlálás a **TPMx_MOD** regiszterben megadott értékig történik, utána MOD-1-re vált a számláló és '1'-be áll a TOF bit, a számlálás pedig nulláig folytatódik.

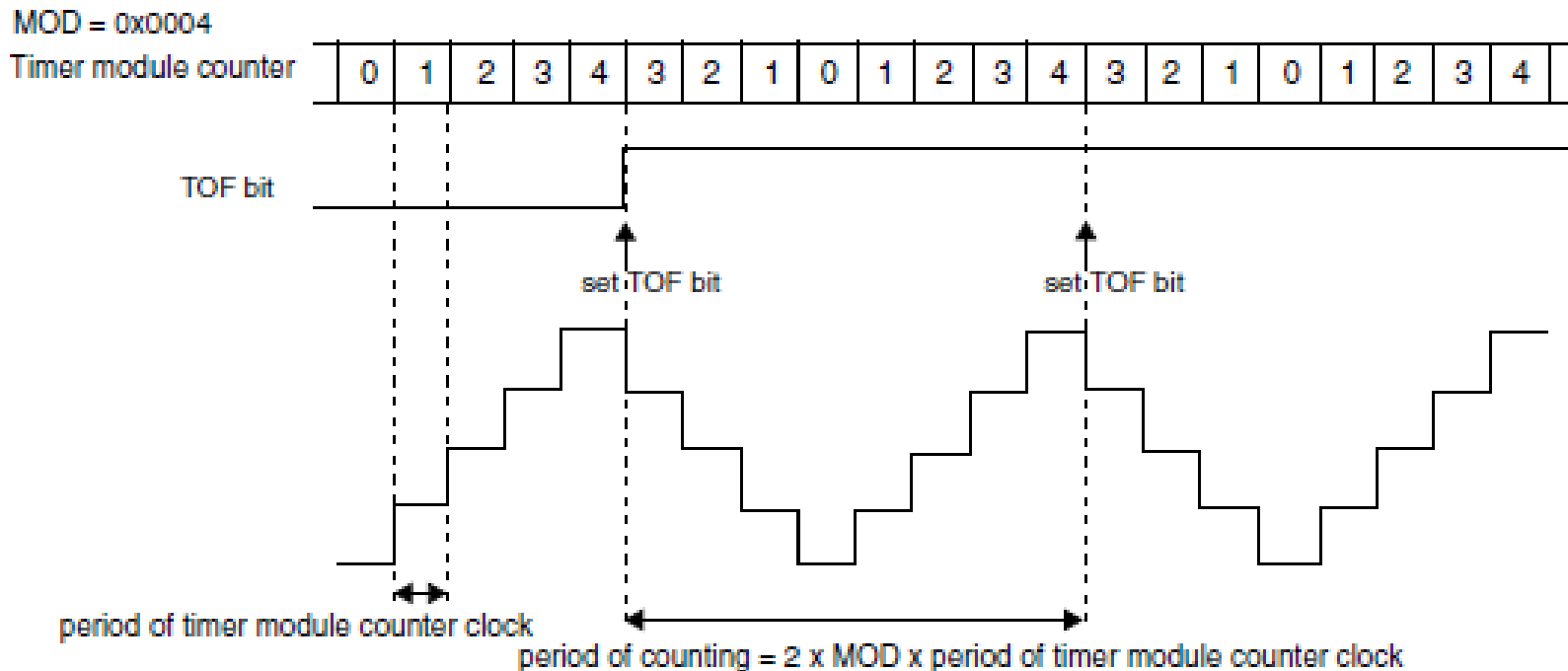


Figure 31-80. Example of Up-Down Counting

A TPMx modulok engedélyezése

Mielőtt valamelyik **TPMx** időzítőt használatba vesszük, engedélyezni a működését a System Clock Gating Control 6 (**SIM_SCGC6**) nevű regiszter megfelelő bitjének '1'-be állításával az alábbi ábra szerint.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R		0		0									0			
W	DAC0		RTC		ADC0	TPM2	TPM1	TPM0	PIT							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

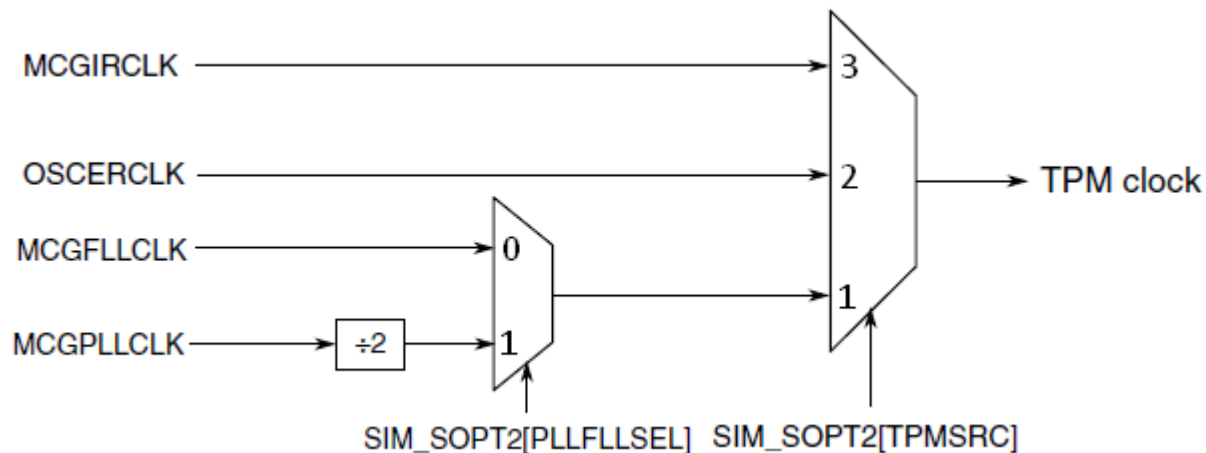
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0							0								
W															DMAMUX	FTF
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

A **SIM_SCGC6** regiszter a Rendszerintegrációs Modulhoz (**SIM**) tartozik, részletes leírása a [Freescale KL25 Sub Family Reference Manual](#) 12. fejezetében található.

A TPM órajel kiválasztása

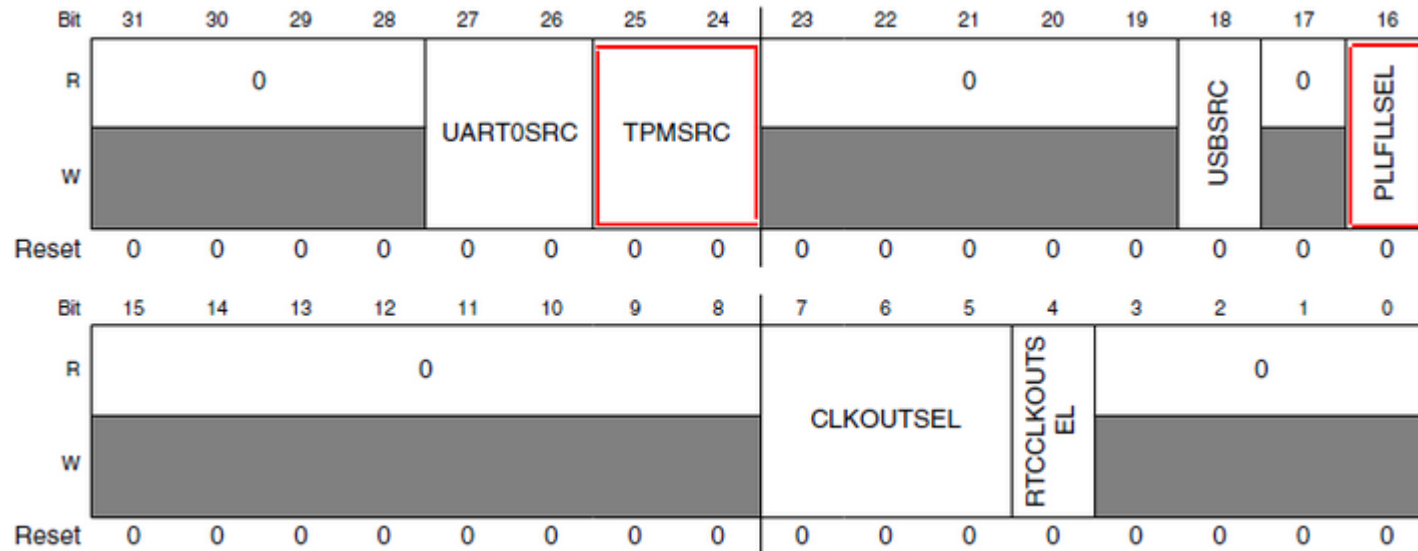
Az órajel kiválasztása a **SIM** rendszerintegrációs modul **SIM_SOPT2** regiszterének konfigurálásával történik. A választott órajel rendelkezésre állása függ az **MCG** órajel generátor konfigurálásától, esetünkben az RTE Device Setup állományokban beállított **MCG** módtól (**CLOCK_SETUP** definiált értéke a **system_MKL25Z4.h** állományban)

CLOCK_SETUP	MCGFLLCLK	MCGPLLCLK/2	OSCERCLK	OSCIRCLK
0	20,97 MHz	–	–	32,768 kHz
1	–	48 MHz	8 MHz	–
2	–	–	–	4 MHz
3	–	–	4 MHz	–
4	–	48 MHz	8 MHz	–



A SIM_SOPT2 regiszter

A **SIM_SOPT2** regiszter bitcsoportjai közül **TPMSRC[1:0]** a **TPM_x** modulok közös órajelének kiválasztására szolgál (**00**: órajel letiltva, **01**: MCGFLLCLK vagy MCGPLLCLK/2, **10**: OSCERCLK, **11**: MCGIRCLK), a **PLLFLLSEL** bit az **FLL** vagy **PLL** áramkör kimenő órajelének kiválasztására szolgál (**0**: MCGFLLCLK, **1**: MCGPLLCLK/2), s ez a bit más perifériák (pl. **UART0**) működésére is hatással van!

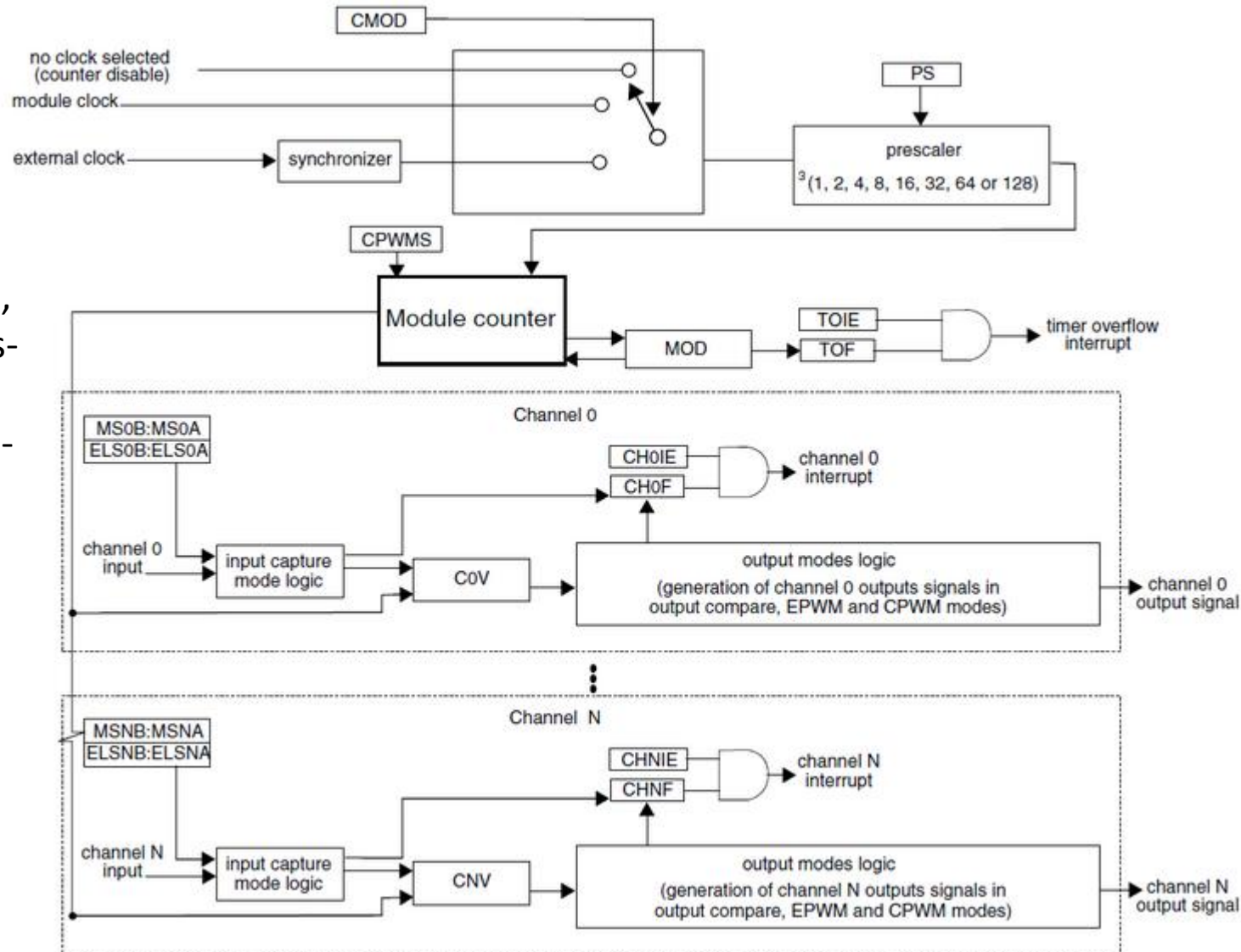


Az **MKL25Z128VLK4** mikrovezérlő órajeleinek előállításáról és elnevezéseiről a [KL25 Sub-Family Reference Manual](#) 5. fejezetében, a **SIM** rendszerintegrációs modul regisztereinek szerepéről és bitkiosztásairól a 12. fejezetében, a többcélú órajel generátor (**MCG**) felépítéséről és konfigurálásáról pedig a 24. fejezetben találunk részletes információt.

A TPM modulok csatornáí

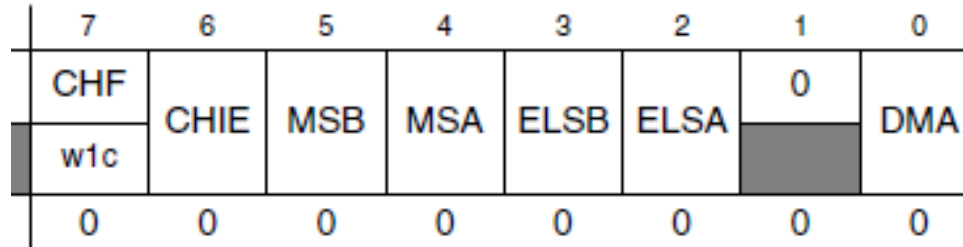
Az **MKL25Z128VLK4** mikrovezérlő általános célú időzítőihez több impulzus-szélesség modulációra (PWM), bemeneti jelrögzítésre (Input Capture), vagy digitális komparálásra (Output Compare) használható csatorna is kapcsolódik:

- TPM0** 6 db,
- TPM1** 2 db,
- TPM2** 2 db csatornával rendelkezik.



Csatorna konfigurációs és adatregiszterek

❖ Konfigurációs regiszter: **TPM_x_CnSC**



- CHF – '1'-be áll, amikor az esemény bekövetkezik
 - CHIE – megszakításkérés engedélyezés
 - MSB:MSA – üzemmód választás
 - ELSB:ELSA – él vagy jelszint választás
 - DMA – DMA átvitel engedélyezése
-
- Adatregiszter: **TPM_x_CnV**
 - 16-bites érték az összehasonlításhoz, vagy bemeneti jelrögzítés eredménye

x – az időzítő sorszáma ($x = 0, 1, 2$)

n – a csatorna sorszáma ($n = 0, \dots, 5$)

Beállítható csatorna üzemmódok

CPWMS	MSB:MSA	ELSB:ELSA	Üzemmód	Konfiguráció
x	00	00	nincs	csatorna letiltva
x	01/10/11	00	komparálás szoftveres vezérléssel	nincs hardveres kimenetvezérlés
0	00	01	Bemeneti jelrögzítés (Input Capture)	rögzítés felfutó élnél
		10		rögzítés lefutól élnél
		11		rögzítés mindkét élnél
	01	01	kimenetvezérlés digitális összehasonlítással (Output Compare)	kimenet átbillentése egyezéskor
		10		kimenet törlése egyezéskor
		11		kimenet '1'-be állítása egyezéskor
0	10	10	Él igazított jelű PWM	kimenet törlése egyezéskor
		x1		kimenet '1'-be állítása egyezéskor
	11	10	impulzus módú kimenetvezérlés digitális összehasonlítással (Output Compare)	negatív impulzus egyezéskor
		x1		pozitív impulzus egyezéskor
1	10	10	Középre igazított jelű PWM	pozitív impulzus jelalak
		x1		negatív impulzus jelalak

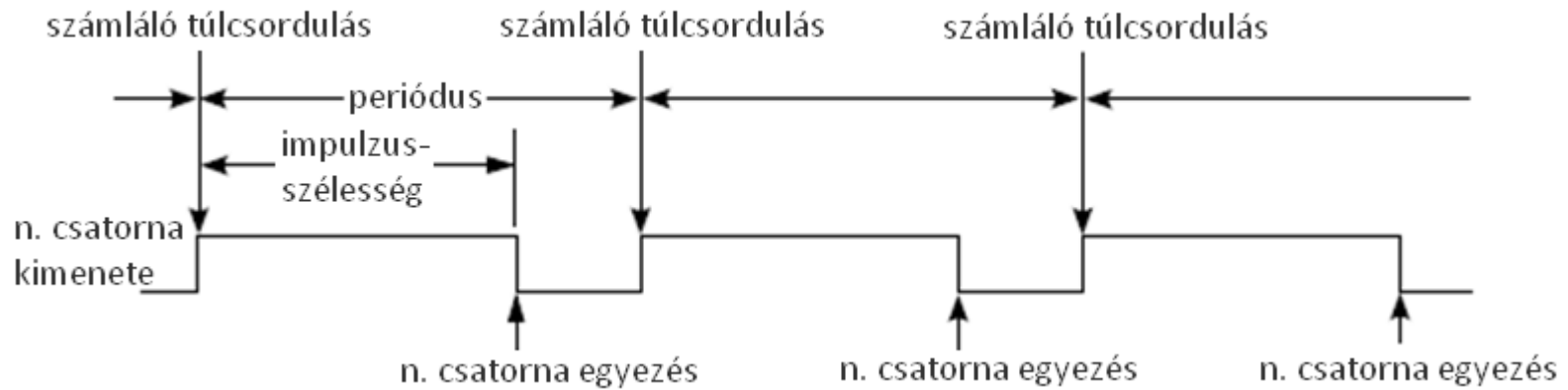
A csatornákhöz rendelhető kivezetések

Mindegyik **TPMx** modul mindegyik csatornájához külön kivezetés rendelhető, s általában egynél több lehetőség közül választhatunk. A választás a megfelelő **PORTx_PCRn** portvezérlő regiszter **MUX** bitcsoportjába írt számmal történik (**Alt3** vagy **Alt4** funkciót kell választanunk, ahogy az alábbi táblázat, illetve a [FRDM-KL25Z Pinouts](#) dokumentum jelzi). A hardverütközések kiszűrésével a **FRDM-KL25Z** kártya esetében az alábbi lehetőségek közül választhatunk:

Időzítő	Csatorna	Kivezetés
TPM0	CH0	PTC1 (4), PTD0 (4)
	CH1	PTA4 (3), PTC2 (4), PTD1 (4)
	CH2	PTA5 (3), PTC3 (4), PTD2 (4), PTE29 (3)
	CH3	PTC4 (4), PTD3 (4), PTE30 (3)
	CH4	PTC8 (3), PTD4 (4), PTE31 (3)
	CH5	PTC9 (3), PTD5 (4)
TPM1	CH0	PTA12 (3), PTB0 (3), PTE20 (3)
	CH1	PTA13 (3), PTB1 (3), PTE21 (3)
TPM2	CH0	PTA1 (3), PTB2 (3), PTB18 (3), PTE22 (3)
	CH1	PTA2 (3), PTB3 (3), PTB19 (3), PTE23 (3)

A zárójelbe írt számok az üzemmód beállításához választandó **Altn** mód sorszámát jelzik.

Él-igazított PWM jel



❑ **Él-igazított kimenet** – a PWM csatornák felfutó éle egymáshoz igazított

- Felfelé számlálási módot használ
- $\text{Periódus} = (\text{MOD} + 1)$ ciklus
- $\text{Impulzus szélesség} = (\text{CnV})$ ciklus

❑ **MSnB:MSnA = 01, CPWMS = 0**

- **ELSnB:ELSnA = 10** – pozitív impulzus
- **ELSnB:ELSnA = x1** – negatív impulzus

Program11_1: 60 Hz, 33 %-os PWM jel

Az alábbi programban a TPM0 1. csatornán 60 Hz-es, 33 %-os kitöltésű jelet állítunk elő, ami a **PTD1** kimeneten jelenik meg (a beépített kék LED-et vezérli).

TPM0 a 48 MHz-es **MCGPLLCLK/2** órajelet használja. Az előszámláló 16-szoros leosztásra van beállítva ($f_{timer} = 3\ 000\ 000\ Hz$). A 60 Hz-es periódushoz a **MOD** regisztert 50000-re, a **CnV** regisztert pedig 16666-ra állítjuk.

```
#include <MKL25Z4.H>

int main (void) {
    SIM->SCGC5 |= 0x1000;           // A D port engedélyezése
    PORTD->PCR[1] = 0x0400;        // PTD1 legyen TPM0 kimenet

    SIM->SCGC6 |= 0x01000000;      // A TPM0 modul engedélyezése
    SIM->SOPT2 |= 0x01000001;      // MCGPLLCLK/2 legyen TPM0 órajele

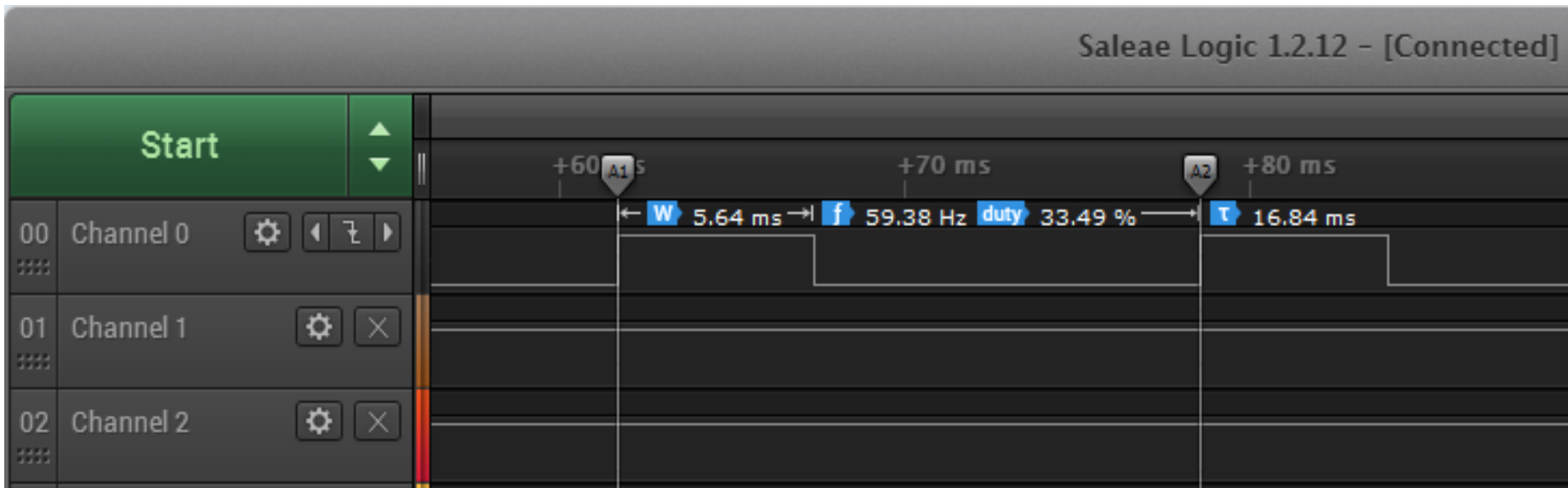
    TPM0->SC = 0;                  // TPM0 letiltása a konfiguráláshoz
    TPM0->CONTROLS[1].CnSC = 0x20 | 0x08; // Él-igazított mód, magas kimenet
    TPM0->MOD = 50000;             // A periódusidő beállítása
    TPM0->CONTROLS[1].CnV = 16666; // A 33 %-os kitöltés beállítása
    TPM0->SC = 0x0C;              // TPM0 engedélyezése (CMOD=01, PS=100)

    while (1) {
    }
}
```

Program11_1: 60 Hz, 33 %-os PWM jel

$T = (1/60) \text{ s} = 16.67 \text{ ms}$, 33 % kitöltésnél a bekapcsolási idő $t_{BE} = 33/100 \cdot 16.67 \text{ ms} = 5,5 \text{ ms}$

Az alábbi ábrán logikai analizátorral 25 kHz-es mintavételezéssel vizsgáltuk a kimenetet. Az időmérés hibája a mintavételezésből adódóan $\pm 0,08 \text{ ms}$ körüli érték (a mintavételezési pontatlanság kétszerese, mert két mintavételezett pont között mérjük az időt).



Megjegyzés: a negatív logikával vezérel LED számára a fenti jel 66 %-os kitöltést jelent!

Program11_2: Változó kitöltésű PWM

```
#include <MKL25Z4.H>
void delayMs(int n);
```

A kék LED-et (PTD1 kimenet, TPM0 1. csatorna) 60Hz-es, változó kitöltésű PWM jellel vezéreljük

```
int main (void) {
    int pulseWidth = 0; // A kitöltési tényező beállításához

    SIM->SCGC5 |= 0x1000; // A D port engedélyezése
    PORTD->PCR[1] = 0x0400; // PTD1 legyen TPM0 kimenet

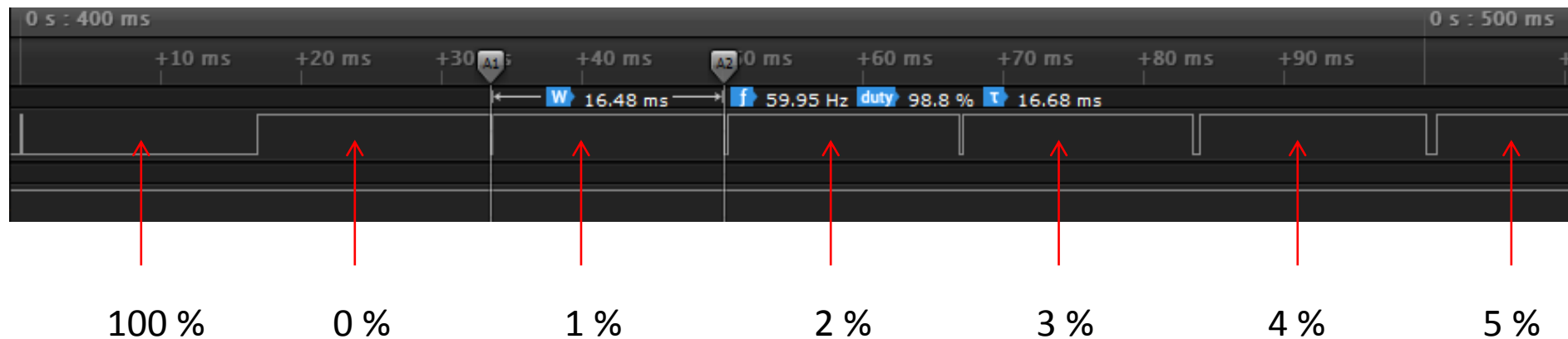
    SIM->SCGC6 |= 0x01000000; // A TPM0 modul engedélyezése
    SIM->SOPT2 |= 0x01000001; // MCGPLLCLK/2 legyen TPM0 órajele
    TPM0->SC = 0; // TPM0 letiltása a konfiguráláshoz
    TPM0->CONTROLS[1].CnSC = 0x20 | 0x04; // Él-igazított mód, alacsony kimenet
    TPM0->MOD = 50000; // A periódusidő beállítása
    TPM0->CONTROLS[1].CnV = 16666; // A 33 %-os kitöltés beállítása
    TPM0->SC = 0x0C; // TPM0 engedélyezése (CMOD=01, PS=100)

    while (1) {
        pulseWidth += 500; // Kitöltés növelése 1 %-os lépésekben
        if (pulseWidth > 50000) // 100 % után visszaállunk 0-ra
            pulseWidth = 0;
        TPM0->CONTROLS[1].CnV = pulseWidth; // Kitöltés beállítása
        delayMs(20); // Hagyjuk érvényesülni a beállítást
    }
}
```

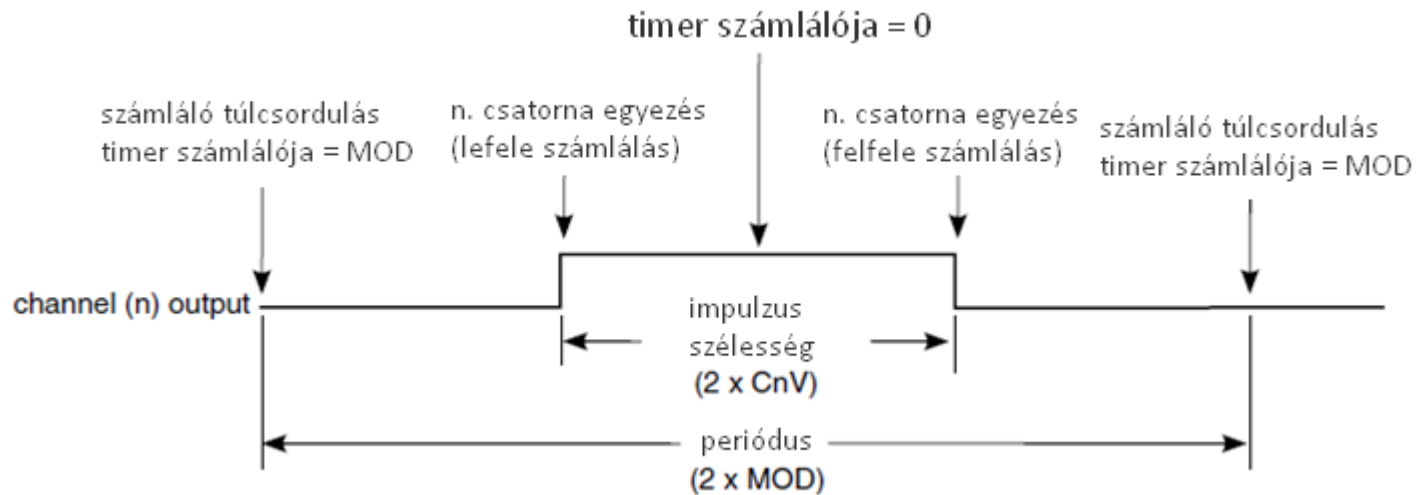
Program11_2: Változó kitöltésű PWM

A LED bekötésének negatív logikája miatt az alacsony állapotban eltöltött idő a bekapcsolási idő (a logikai analizátor ennek a komplementjét számolja és írja ki).

Az alábbi ábrán a 100 %-ról 0 %-ra ugrás és az azt követő néhány lépés látható (a **PTD1** kimeneten jelenik meg a jel).



Középre igazított PWM jel



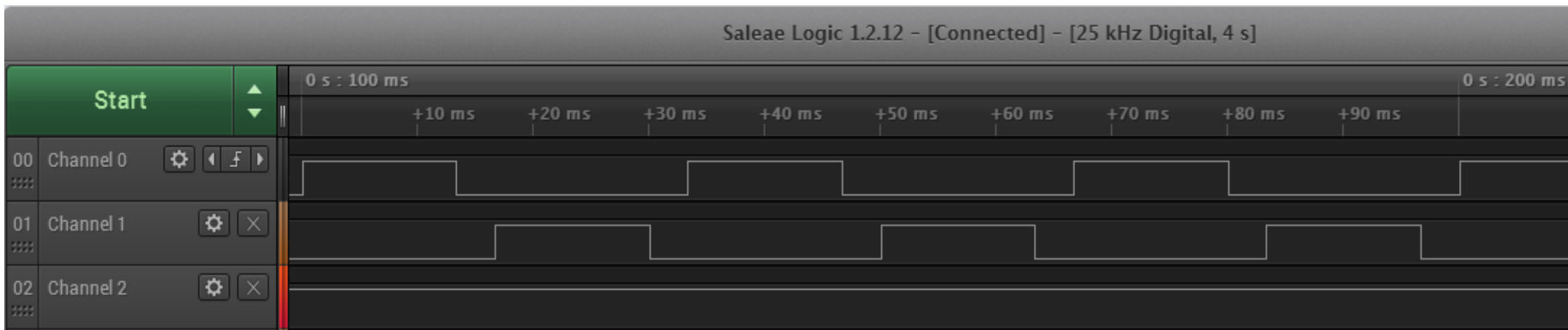
- ❑ **Középre igazított jel** – a PWM csatornák jelének közepe egymáshoz igazított
 - A le/fel számlálási módot használjuk (CPWMS = 1)
 - Periódus = $2 * MOD$ ciklus. ($0x0001 \leq MOD \leq 0x7FFF$ legyen!)
 - Impulzus-szélesség = $2 * CnV$ ciklus
- ❑ **MSnB:MSnA = 10, CPWMS = 1**
 - ELSnB:ELSnA = 10 – pozitív logikájú impulzus (a magas szint az „igaz” állapot)
 - ELSnB:ELSnA = x1 – negatív logikájú impulzus (alacsony szint az „igaz” állapot)

Program11_5: holtsáv generálás

Az alábbi programban TPM0 0. csatornával (**PTD0** kimenet) 60 %-os kitöltésű, középre igazított, negatív logikájú jelet keltünk, az 1. csatornával (**PTD1** kimenet) pedig 40%-os kitöltésű, középre igazított, pozitív logikájú jelet állítunk elő. Ez végeredményben 10 %-os holtsávot eredményez a 0. és az 1. csatorna jele között.

A holtsáv feltétele, hogy a két jel bekapcsolási idejének összege 100 %-nál kevesebb legyen. Azonos hosszúságú bekapcsolási idők esetén ez azt jelenti, hogy mindkettő 50 % alatti legyen!

Ilyen holtsáv generálásra például akkor van szükség, amikor nagyáramú fogyasztót *totem-pole* ellenütemű kimenettel vezérlünk (pl. motorvezérlés H-híddal). A holtsáv biztosítja, hogy a korábban vezető tranzisztor biztosan lezárjon, mielőtt a vele sorba kötött félvezető vezetni kezdene.



Program11_5: holtcsík generálás

```
#include <MKL25Z4.H>

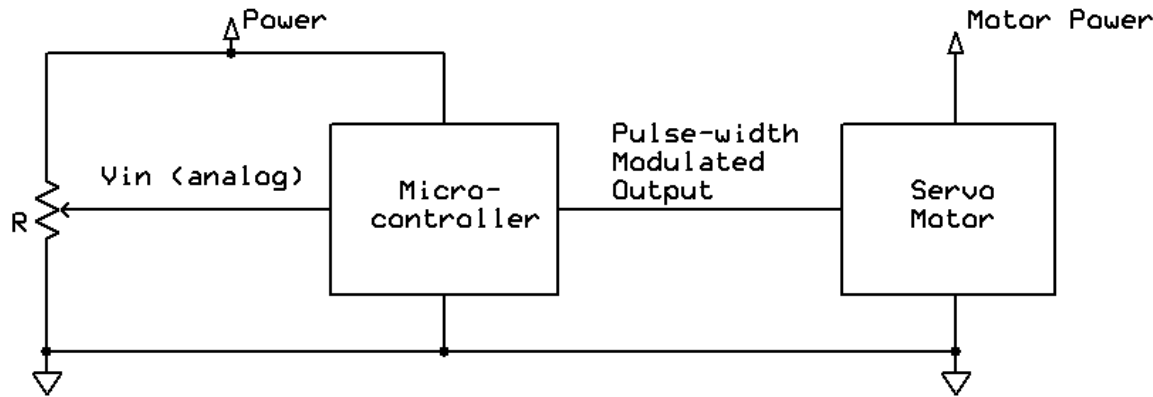
int main (void) {
    SIM->SCGC5 |= 0x1000;           // A D port engedélyezése
    PORTD->PCR[0] = 0x0400;        // PTD0 legyen TPM0 kimenet
    PORTD->PCR[1] = 0x0400;        // PTD1 legyen TPM0 kimenet

    SIM->SCGC6 |= 0x01000000;       // A TPM0 modul engedélyezése
    SIM->SOPT2 |= 0x01000001;       // MCGPLLCLK/2 legyen TPM0 órajele
    TPM0->SC = 0;                   // TPM0 letiltása a konfiguráláshoz
    TPM0->CONTROLS[0].CnSC = 0x20 | 0x04; // Közép-igazított mód, alacsony kimenet
    TPM0->CONTROLS[1].CnSC = 0x20 | 0x08; // Közép-igazított mód, magas kimenet
    TPM0->MOD = 50000;              // A periódusidő beállítása (30 Hz)
    TPM0->CONTROLS[0].CnV = 30000; // 60 %-os kitöltés beállítása
    TPM0->CONTROLS[1].CnV = 20000; // 40 %-os kitöltés beállítása
    TPM0->SC = 0x0C | 0x20;         // TPM0 eng., /16 előosztás, le-fel számlálás
                                    // (CMOD=01, PS=100, CPWMS=1)

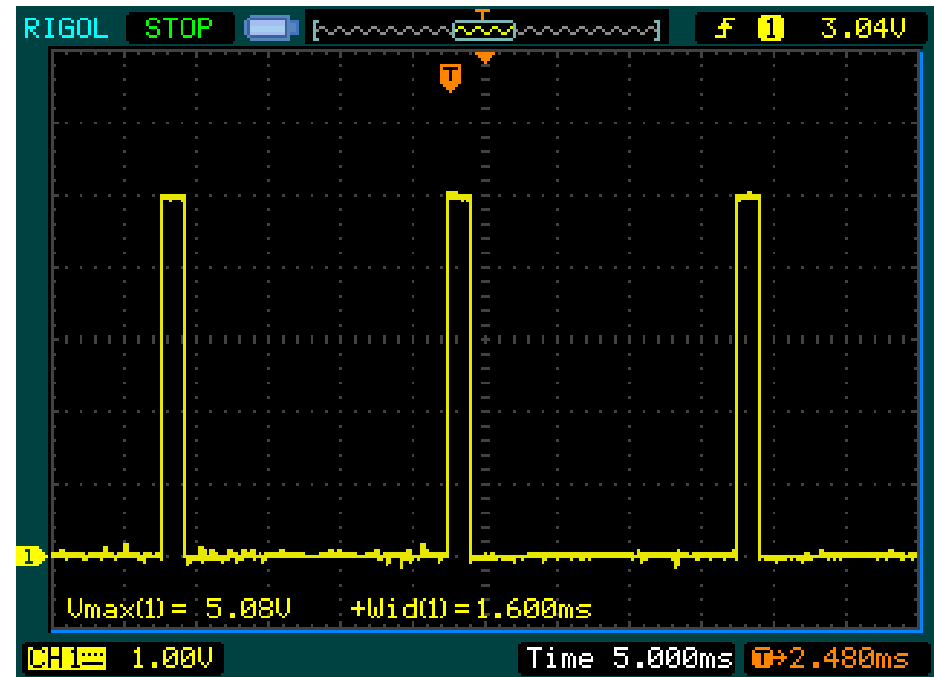
    while (1) {
    }
}
```

Megjegyzés: a `TPM0->MOD = 50000;` beállítás kívül esik az ajánlott tartományon ($0x0001 \leq \text{MOD} \leq 0x7FFF$), de nem tapasztaltunk emiatt problémát.

PWM to Drive Servo Motor

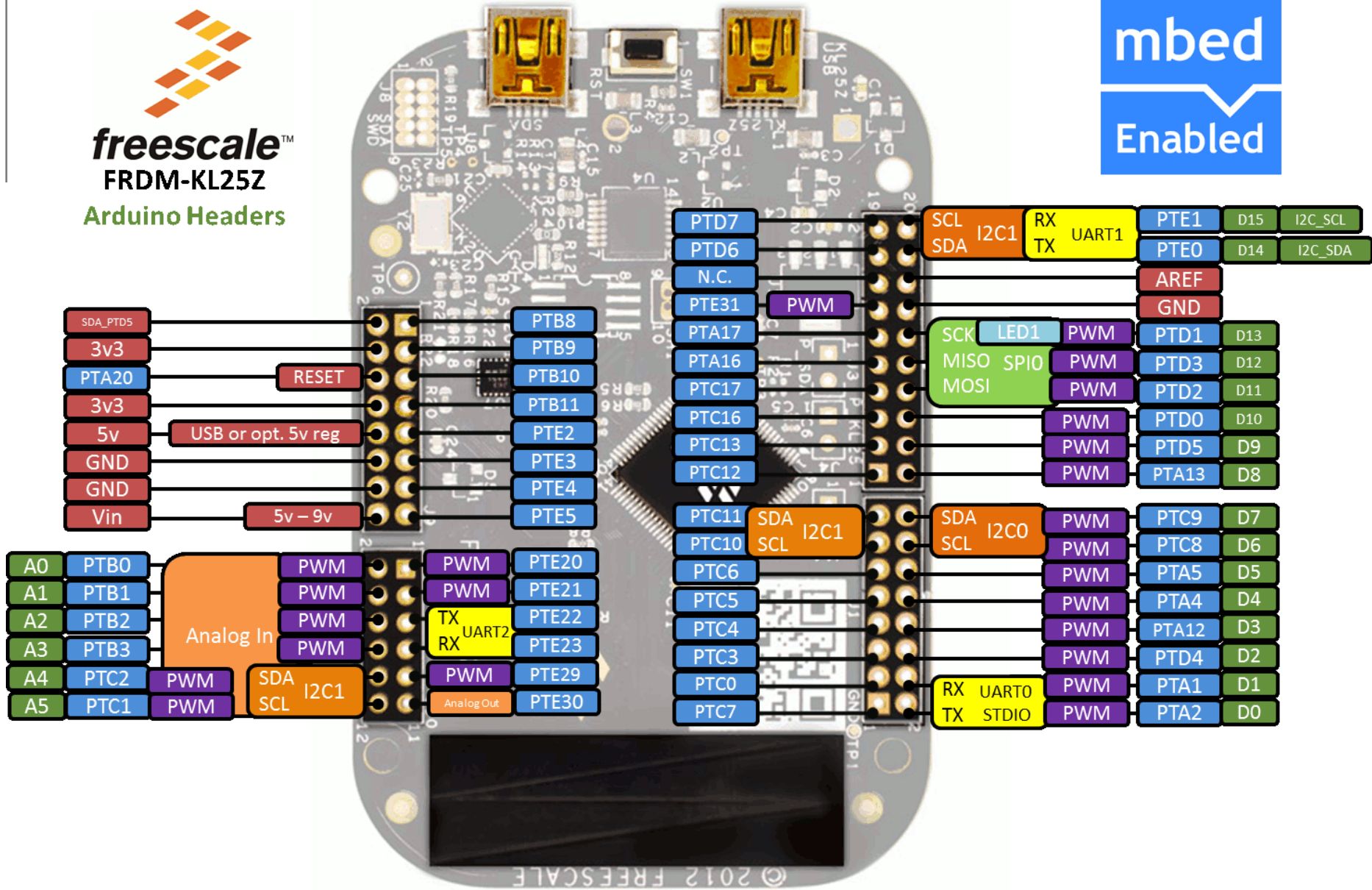
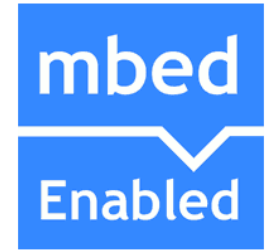


- Servo PWM signal
 - 20 ms period
 - 1 to 2 ms pulse width





freescale™
FRDM-KL25Z
 Arduino Headers





freescale™
FRDM-KL25Z

Additional Peripherals



- PTE24 SCL
- PTE25 SDA
- PTA14 INT1
- PTA15 INT2

freescale
MMA8451Q
Accelerometer

- LED1 PTB18 PWM
- LED2 PTB19 PWM
- LED3 PTD1 PWM

RGB
LED

Capacitive Touch Slider
PTB16 PTB17

