



Akademia Górniczo-Hutnicza
w Krakowie
Katedra Elektroniki
WIET



Laboratorium Techniki Mikroprocesorowej 2

Ćwiczenie 4

Moduły licznika TPM/PWM - ich obsługa i sposób wykorzystania

Autorzy: Sebastian Koryciak & Paweł Russek
ver. 06.11.2020

1. Cel ćwiczenia

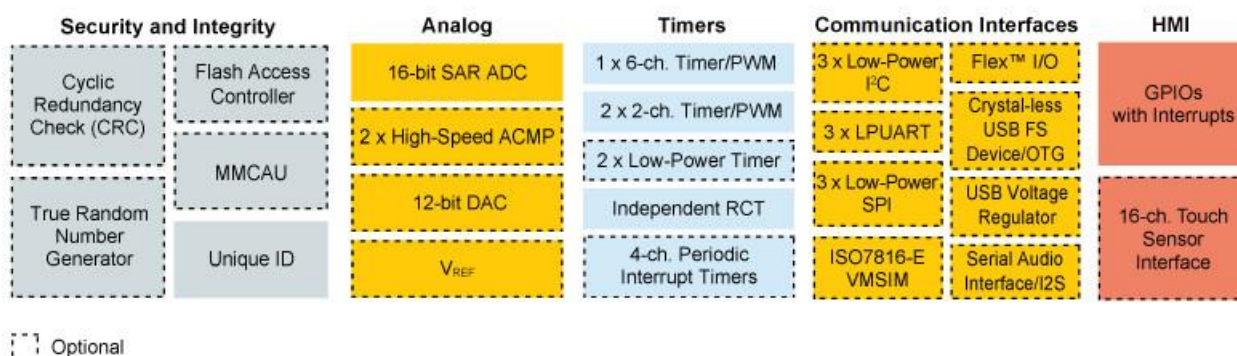
Celem tego ćwiczenia, jest zaznajomienie uczestników z układami czasowymi/licznikami dostępnymi w mikrokontrolerze Kinetis KL05Z. Do realizacji tego celu posłużą przykłady wykorzystania tych układów do generowania przebiegów prostokątnych, zliczania impulsów zewnętrznych, pomiaru czasu, generowania przerwań czy żądań transmisji od układu DMA.

1.1. Wymagania

- Komputer PC z zainstalowanym środowiskiem Keil uVision
- Zestaw uruchomieniowy FRDM-KL05Z
- Znajomość podstaw programowania układu FRDM-KL05Z w języku C

1.2. Literatura

- Joseph Yiu, The Definitive Guide to the ARM Cortex-M0, Elsevier, 2011
- KL05 Sub-Family Reference Manual, Freescale Semiconductor
- Kinetis L Peripheral Module Quick Reference, Freescale Semiconductor
- Power Management for Kinetis and ColdFire+ MCUs. , Freescale Semiconductor



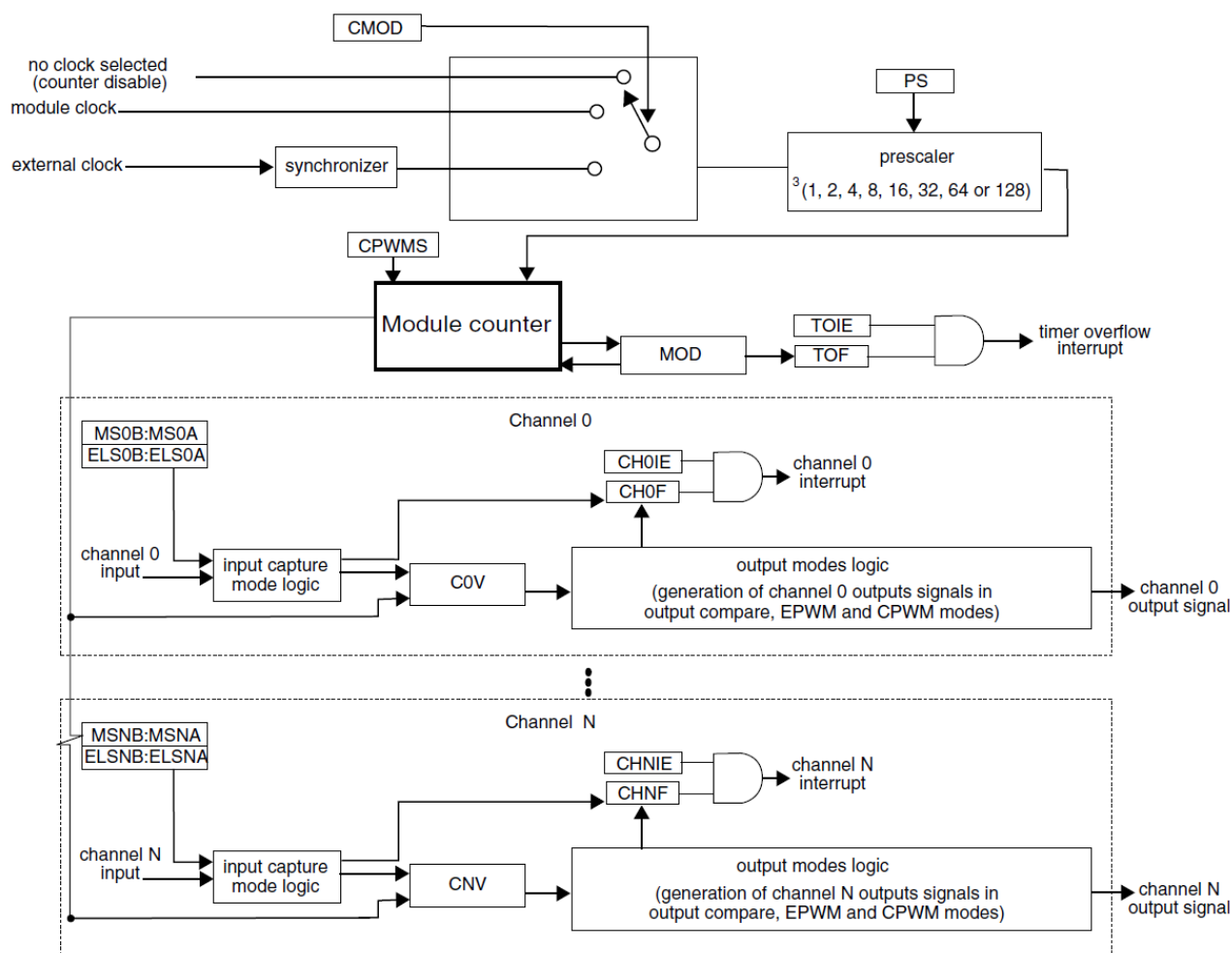
Rysunek 1. Schemat blokowy peryferiów mikrokontrolerów Kinetis KL0x

2. Moduł TPM

Blok TPM (Timer/PWM Module) w przypadku mikrokontrolera KL05 posiada dwa moduły. Pierwszy, TPM0 posiada sześć kanałów korzystających ze wspólnego licznika i drugi - TPM1, posiadający dwa kanały korzystające z drugiego licznika. Schemat blokowy modułu TPM przedstawiony jest na rysunku 2.

Funkcjonalność modułu TPM składa się z trzech podstawowych trybów pracy:

- **Input capture**, czyli pomiarów czasowych sygnałów podawanych na wejście modułu,
- **Output compare**, czyli generacji przebiegów na wyjściu w oparciu o porównywanie zawartości swobodnego licznika i wartości odniesienia w innym rejestrze,
- **PWM**, czyli generacji (cyklicznych) impulsów o zmiennej, programowanej szerokości.



Rysunek 2. Schemat blokowy modułu TPM

Wejście „channel N input” i wyjście „channel N output signal” to jedna i ta sama końcówka, której rola zależy od wybranego trybu pracy.

TPM zawiera dwa elementy służące taktowaniu jego pracy. Zegar magistrali systemowej służy komunikacji z rejestrami i generacji przerw oraz żądań DMA. Zegar licznika służy jako baza dla logiki realizującej funkcję pomiarów i generacji przebiegów, dlatego jest uważany za asynchroniczny względem zegara magistrali. Zegar licznika może być wyższy niż zegar magistrali i pozostać aktywny w trybach oszczędzania energii. Źródłem taktowania dla głównego licznika może być:

- MCGFLLCLK,
- MCGPLLCLK/2,
- OSCERCLK,
- MCGIRCLK (*low-power mode*),

Liczniki TPM mogą być taktowane również z zewnętrznego źródła (TPM_EXTCLK). TPM_CLKIN0 lub TPM_CLKIN1 mogą być wejściami tych sygnałów taktujących. Ustawiany wówczas TPMx_SC[CMOD], SIM_SOPT4[TPMxCLKSEL].

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
								TOF		CH5F	CH4F	CH3F	CH2F	CH1F	CH0F

Rysunek 5. Rejestr TPMx_STATUS.

Bity TPMx_STATUS:

- TOF - Flaga przepełnienia timera
- CHnF - flaga 'n' kanału. „0” - nie wystąpiło żadne zdarzenie kanału. „1” - wystąpiło zdarzenie na danym kanale.

Rejestr MOD określa do jakiej wartości liczy rejestr CNT. Komparator wykrywa stan przepełnienia, który występuje, gdy CNT próbuje policzyć poza wartość zapisaną w MOD. Licznik CNT jest modyfikowany zgodnie z wartością bitu CPWMS w TPMx_SC:

- Jeśli licznik jest skonfigurowany do zliczania w górę (CPWMS = 0), to CNT jest zerowane. Okres zliczania wynosi wówczas MOD+1.
- Jeśli licznik jest skonfigurowany do zliczania w górę / w dół (CPWMS = 1), wówczas kierunek jest ustawiany na zliczanie w dół, a wartość CNT jest zmniejszana. Odliczanie do zera skutkuje stanem nieomiaru, więc kierunek zliczania zmienia się w górę. Kolejne wejście zegara zwiększy CNT do jednego. Flaga TOF jest ustawiana w momencie zmiany wartości z MOD na MOD-1. Okres zliczania wynosi wówczas 2*MOD.

2.1. Wyliczanie czasu okresu TPM_CNT

Długość okresu zależy od częstotliwości zegara taktującego, jego preskalera, wartości modulo (MOD), a w przypadku trybu PWM również od sposobu przylegania do brzegów szczeliny czasowej i można ją opisać w następujący sposób:

$$TPM_{okres} = TPMxMOD \times \frac{\text{wartość preskalera}}{\text{częstotliwość zegara taktującego TPM}} \times \frac{1}{1 + TPM_SC_CPWMS}$$

Przykładowo, jeśli:

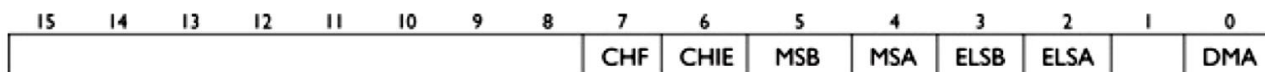
- zegar taktujący MCGFLLCLK = 48MHz (SIM_SOPT2_TPMSRC = 1, SIM_SOPT2_PLLFLLSEL = 0),
- preskaler wynosi 16 (TPMx_SC_PS = 0x4),
- wartość modulo jest ustawiona na 3000 (TPMx_MOD = 0x0BB8),

to długość okresu wyniesie:

$$TPM_{okres} = 3000 \times \frac{16}{48\,000\,000} \times \frac{1}{1 + 0} = 1ms$$

2.2. Kanały TPM

Każdy kanał TPM ma rejestr stanu i kontroli (TPMx_CnSC), pokazany na rysunku 6. Każdy kanał może być skonfigurowany do pracy w trybie przechwytywania wejścia lub generowania wyjścia, zgodnie z bitami wyboru trybu (MSB i MSA) i pól wyboru zbocza (ELSB i ELSA) w TPMx_CnSC.



Rysunek 6. Rejestr TPMx_CnSC.

2.3. Tryby pracy TPM

a) Input Capture

Korzystanie z trybu przechwytywania sygnału wejściowego sprawia, że pomiar czasu sygnałów wejściowych jest znacznie mniej podatny na opóźnienia w przetwarzaniu przerwań, ponieważ moduł kanału automatycznie rejestruje wartość licznika czasu bez potrzeby jakiegokolwiek interwencji oprogramowania.

Podczas gdy kanał jest w trybie przechwytywania sygnału wejściowego, sygnał wejściowy jest monitorowany pod kątem określonej zmiany (zbocze narastające, zbocze opadające lub jedno i drugie). Kiedy następuje zadany warunek, wartość rejestru TPMx_CNT jest przechwytywana w polu VAL rejestru CnV danego kanału, a flaga kanału CHF w TPMx_CnSC jest ustawiana na „1”. Jeśli pole włączające przerwanie kanału (CHIE) wynosi „1”, wtedy TPMx wygeneruje przerwanie.

ISR musi skopiować przechwyconą wartość z rejestru TPMx_CnV kanału i użyć jej. ISR musi również wyczyścić flagę kanału, wpisując do niej „1”.

Dla trybu przechwytywania danych wejściowych konfigurowujemy TPMx_CnSC w następujący sposób:

- Wybierz tryb przechwytywania wejścia, ustawiając bity MSB i MSA na 0:0. CPWMS musi również wynosić „0”.
- Wybierz, które zbocza wyzwalają przechwytywanie, używając ELSnB:ELSnA: 0:1 dla zboczy narastających, 1:0 dla zboczy opadających, 1:1 dla zboczy narastających i opadających.
- Jeśli jest używane, włącz przerwanie, ustawiając CHIE na „1”.

b) Output Compare

Każdy kanał TPM ma układ logiczny, który może wykryć, kiedy licznik CNT modułu TPM osiągnie określoną wartość. W odpowiedzi kanał może zmienić swój sygnał wyjściowy lub wyzwolić przerwanie. Sygnał może zmienić wartość, gdy rejestr TPMx_CNT przepełni się, nastąpi niedomiar lub zrówna się do wartości zapisanej w TPMx_CnV.

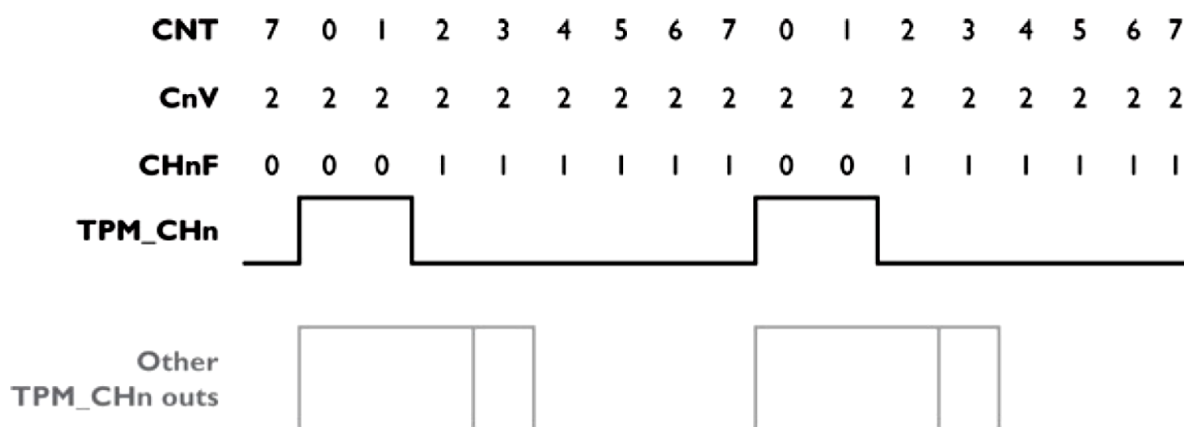
Możliwe są do stworzenia różne kombinacje sygnałów wyjściowych, w tym pojedynczy impuls o określonej szerokości lub opóźnieniu, czy ciągły strumień impulsów o określonym cyklu pracy. Flaga kanału CHF zostanie ustawiona, gdy TPMx_CNT osiągnie wartość TPMx_CnV. Jeśli CHIE jest ustawione, TPMx wygeneruje przerwanie.

Najbardziej podstawowym trybem pracy wyjścia jest tryb porównania wyjścia. W takim przypadku kanał będzie odpowiadał za każdym razem, gdy CNT osiągnie CnV (na początku cyklu). Wyjście można skonfigurować tak, aby było ustawione na jeden lub zerowane przy każdym dopasowaniu. Można to wykorzystać do wytworzenia impulsu o ustalonym czasie trwania. Po pierwszym dopasowaniu wyjście pozostaje w nowym stanie do momentu, gdy oprogramowanie wyraźnie go zmieni. Wyjście można również skonfigurować tak, aby przełączało się po każdym dopasowaniu (*toggle mode*), w którym to przypadku sygnał wyjściowy jest falą prostokątną z przesunięciem czasowym (opóźnieniem fazy) określonym przez CnV.

c) PWM

Każdy kanał modułu TPM może generować sygnał PWM o cyklu pracy CnV/MOD. Polaryzację sygnału wyjściowego można wybrać jako wysoką (*high-true*) lub niską (*low-true*). Ponadto impulsy kanałów można wyrównać do zbocza (EPWM) lub do środków (CPWM).

Rysunek 7 przedstawia przykład trybu PWM wyrównanego zboczem, z MOD=7 i CnV=2. Licznik CNT zlicza w górę od zera do MOD, więc jego okres zliczania wynosi $(MOD + 1) / f_{count}$.



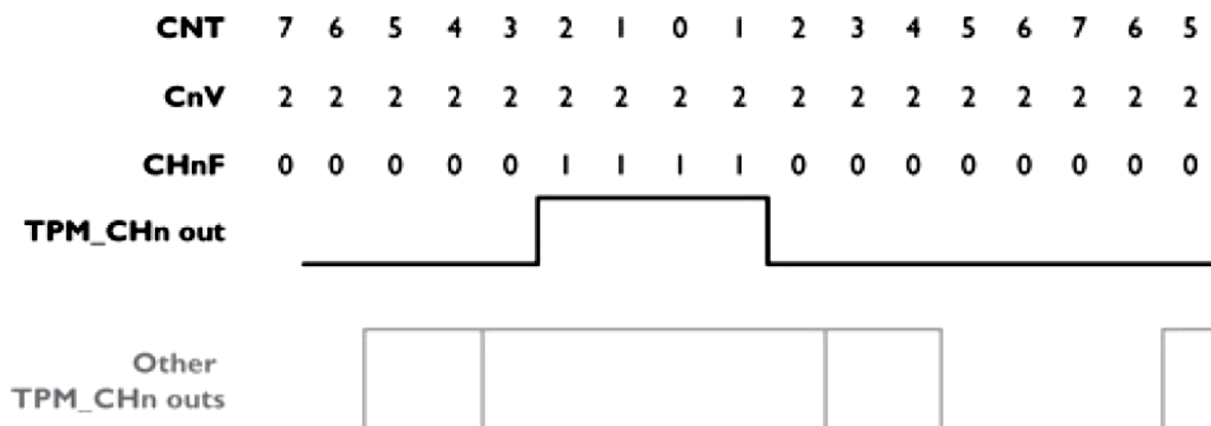
Rysunek 7. Działanie TPM w trybie PWM Edge-aligned (*high-true*).

Wyjście każdego kanału jest inicjalizowane (np. '1') za każdym razem, gdy licznik CNT się przepełni (np. 7 -> 0). CNT liczy i zrównuje się z CnV, w którym to momencie sygnał wyjściowy kanału TPM_CHn jest zmieniany (np. '0'). CNT kontynuuje liczenie i ostatecznie zrównuje się z wartością MOD i następuje przepełnienie. Cykl się powtarza. Szerokość impulsu sygnału wynikowego jest proporcjonalna do CnV, a cykl pracy to CnV/MOD. Zauważ, że początkowa krawędź wyjścia kanału jest wyrównana z momentem przepełnienia licznika CNT. Jeśli dodatkowe kanały są włączone, wszystkie będą miały zbocza narastające jednocześnie (w pozostałych kanałach możemy ustawić inne wartości CnV, ale MOD jest wspólny).

Rysunek 8 przedstawia przykład wyśrodkowanego trybu PWM. Licznik CNT na przemian zlicza w dół i w górę, więc jego okres zliczania wynosi $(2 * MOD) / f_{count}$.

Licznik CNT jest inicjalizowany na wartość MOD, co również inicjalizuje wyjście kanału (np. '0'). CNT następnie odlicza w dół i ostatecznie wyrównuje się z wartością CnV, w którym to momencie wyjście kanału jest przełączane (np. '0' -> '1'). CNT kontynuuje liczenie i osiąga zero, a w tym momencie zmienia kierunek liczenia i zaczyna zliczać w górę. CNT osiąga wartość CnV, a wyjście kanału jest przełączane (np. '1' -> '0'). CNT kontynuuje, aż osiągnie wartość MOD, po czym zmienia kierunek zliczania i zaczyna odliczanie. Cykl się powtarza. Szerokość impulsu

sygnału wynikowego jest proporcjonalna do CnV, a cykl pracy wynosi $(2 * CnV) / (2 * MOD)$
 $= CnV / MOD$.



Rysunek 8. Działanie TPM w trybie PWM Center-aligned (high-true).

Wybór trybu pracy kanału odbywa się w rejestrze kontrolno-statusowym TPMx_CnSC, według poniższej tabeli:

CPWMS	MSnB:MSnA	ELSnB:ELSnA	Mode	Configuration
X	00	00	None	Channel disabled
X	01/10/11	00	Software compare	Pin not used for LPTPM
0	00	01	Input capture	Capture on Rising Edge Only
		10		Capture on Falling Edge Only
		11		Capture on Rising or Falling Edge
	01	01	Output compare	Toggle Output on match
		10		Clear Output on match
		11		Set Output on match
	10	10	Edge-aligned PWM	High-true pulses (clear Output on match, set Output on reload)
		X1		Low-true pulses (set Output on match, clear Output on reload)
	11	10	Output compare	Pulse Output low on match
		X1		Pulse Output high on match
1	10	10	Center-aligned PWM	High-true pulses (clear Output on match-up, set Output on match-down)
		X1		Low-true pulses (set Output on match-up, clear Output on match-down)

Tabela 1. Tryby pracy TPM i ich konfiguracja za pomocą bitów rejestru TPMx_CnSC.

2.4. Wejścia wyzwalające TPM

Wyzwolenie zliczania może odbyć się na parę sposobów:

- programowo - wyzwalanie sprzętowe zablokowane (TPMx_CONF[CSOT=0]), licznik główny włączony (TPMx_SC[CMOD=1] {2 dla zegara zewnętrznego})
- sprzętowo - wyzwalanie sprzętowe odblokowane (TPMx_CONF->CSOT=1), licznik główny włączony (TPMx_SC[CMOD=1] {2 dla zegara zewnętrznego}).

Źródłem wyzwalania sprzętowego może być (zbocze narastające):

- sygnał zewnętrzny, podawany na wejście EXTRG_IN,
- CMP0,
- PITx,
- TPMx (przepełnienie),
- RTC,
- LPTMR.

Źródło wyzwalania sprzętowego wybiera się w rejestrze TPMx_CONF[TRGSEL].

3. Przygotowanie do laboratorium

ToDo

1.1 Podłącz do płytki FRDM-KL05 wyświetlacz LCD zgodnie ze wskazówkami z pierwszej instrukcji (jeżeli nie działa, pamiętaj o sprawdzeniu adresu w pliku *lcd1602.c*).

1.2 Ze strony laboratorium pobierz archiwum z projektem.

1.3 Rozpakuj plik do nowo utworzonego folderu, a następnie otwórz projekt (*demo.uvprojx*).

1.4 Projekt podzielony jest na foldery 'src' i 'inc'. Projekt składa się z plików bibliotecznych z pierwszego laboratorium oraz plików:

- *pit.c* - zawiera inicjalizację PIT i funkcje pomocnicze,
- *tpm.c* - zawiera funkcje związane z timerami.

1.5 Zbuduj projekt i załaduj program na płytkę FRDM-KL05. Po prawidłowym wykonaniu powyższych czynności układ powinien zachowywać się w następujący sposób:

- dioda RGB - po sekwencji powitalnej (krótkie zaświecenie każdego z kolorów) dioda niebieska (PTB10) miga z częstotliwością około 5Hz,
- wyświetlacz LCD - wyświetla "Slider = 50",
- slider TSI - dotknięcie płytki dotykowej powoduje zmianę wartości wyświetlanej na LCD oraz zmianę częstotliwości migania niebieskiej diody.

Generator zbudowany jest w oparciu o moduł PIT (laboratorium 3), którego inicjalizacja następuje z użyciem funkcji *PIT_Init_Generator*, a następnie zmiana generowanej częstotliwości poprzez funkcję *PIT_SetTSV*. Przerwanie od kanału 0 modułu PIT przełącza pin PTB10 (dioda niebieska).

Główna pętla programu składa się z dwóch zadań, które uruchamiane są w oparciu o zegar systemowy (SysTick Timer - laboratorium 3). Pierwsze polega na odczycie wartości z płytki dotykowej i na jej podstawie aktualizacji generowanej częstotliwości. Drugie zadanie to odświeżanie wyświetlanych wartości na LCD.

4. Pomiar czasu w trybie *Input Capture*

ToDo

2.1 Otwórz plik *tpm.c* i wykonaj następujące etapy inicjalizacji timera:

2.1.1 Podłącz zegar do modułu TPM1 i wybierz MCGFLLCLK jako jego źródło

2.1.2 Podłącz zegar do portu A, który będzie wykorzystany jako wejście pomiarowe. Podłącz pin 0 portu A do timera TPM1 - wybierz odpowiednią alternatywę na multiplekserze i zapisz numer kanału TPM1.

2.1.3 Ustaw preskaler na wartość podzielnika zegara równą 128.

2.1.4 Wybierz zliczanie licznika na narastającym zboczu zegara.

2.1.5 Ustaw bity: CPWMS, MSA, MSB, ELSA, ELSE, tak aby kanał licznika TPM1 podłączony do pinu 0 portu A pracował w trybie "input capture" z reakcją na oba zbocza (patrz Tabela 1).

2.1.6 Włącz przerwanie od zadanego kanału timera.

2.1.7 Aktywuj miganie diody czerwonej w obsłudze przerwania od TPM1.

2.1.8 Na samym końcu obsługi przerwania od TPM1 wymaż flagę z odpowiedniego kanału.

2.2 Otwórz plik *main.c* i dołącz plik nagłówkowy TPM oraz włącz inicjalizację TPM1.

2.3 Zbuduj projekt i załaduj program na płytkę FRDM-KL05. Po prawidłowym wykonaniu powyższych czynności układ powinien zachowywać się w następujący sposób:

- podłącz kabelkiem wyjście generatora (PTB10) z wejściem układu pomiarowego (PTA0)
- w zależności na jaką polaryzację się natrafi diody niebieska z czerwoną zacząć migać jednocześnie tworząc kolor fioletowy lub naprzemiennie
- sprawdź czy podczas zmiany częstotliwości diody zostają w synchronizacji

2.4 W pliku *tpm.c* w obsłudze przerwania od TPM1 włącz zapisywanie wartości licznika do zmiennej lokalnej.

2.5 W pliku *main.c* w funkcjach do wyświetlania (*lcd_static* i *lcd_update*) włącz wypisywanie wartości licznika na LCD.

2.6 Zbuduj projekt i załaduj program na płytkę FRDM-KL05. Po prawidłowym wykonaniu powyższych czynności układ powinien zachowywać się w następujący sposób:

- z ciągle podłączonym kabelkiem (PTB10 - PTA0) - na wyświetlaczu LCD pojawi się wartość odczytana z rejestru CnV (Channel Value).
- proszę policzyć zmierzoną częstotliwość dla skrajnych wartości slidera (2 i 96) na podstawie inicjalizacji timera TPM1 i wartości odczytanych z licznika, a następnie zaproponować funkcję wyświetlającą na LCD zmierzoną częstotliwość w Hz.

5. Ściemnianie diody LED w trybie PWM

ToDo

3.1 Otwórz plik *tpm.c* i wykonaj następujące etapy inicjalizacji timera:

3.1.1 Podłącz zegar do modułu TPM0 i wybierz MCGFLLCLK jako jego źródło

3.1.2 Podłącz zegar do portu B, który będzie wykorzystany jako wyjście PWM.

Podłącz pin 9 portu B do timera TPM0 - wybierz odpowiednią alternatywę na multiplekserze i zapisz numer kanału TPM0.

3.1.3 Ustaw preskaler na wartość podzielnika zegara równą 128.

3.1.4 Wybierz zliczanie licznika na narastającym zboczu zegara.

3.1.5 Ustaw wartość rejestru modulo na wartość jaką maksymalnie możesz uzyskać na swoim sliderze.

3.1.6 Ustaw bity: CPWMS, MSB, ELSA, tak aby kanał licznika TPM0 podłączony do pinu 9 portu B pracował w trybie "PWM" z generacją impulsów "low-true" i "edge-aligned" (patrz Tabela 1).

3.1.7 Ustaw początkową wartość licznika.

3.1.8 W funkcji ustawiającej nową wartość licznika (*TPM0_SetVal*) wybierz właściwy kanał.

3.2 Proszę wyłączyć inicjalizację modułów PIT i TPM1 w pliku *main.c* (zakomentować funkcje *PIT_Init_Generator* i *TPM1_Init_InputCapture*).

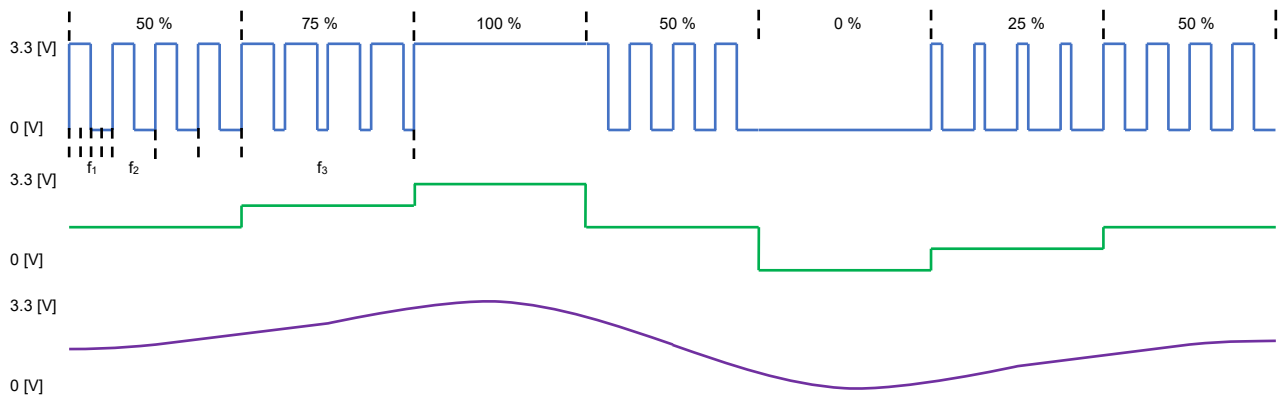
3.3 W pliku *main.c* włącz inicjalizację modułu TPM0 (*TPM0_Init_PWM*) oraz aktywuj w pętli głównej przepisywanie wartości odczytanej ze slidera do rejestru CnV modułu TPM0.

3.4 Zbuduj projekt i załaduj program na płytkę FRDM-KL05. Po prawidłowym wykonaniu powyższych czynności układ powinien zachowywać się w następujący sposób:

- dotknięcie płytki dotykowej powoduje zmianę wartości wyświetlanej na LCD oraz zmianę jasności świecenia zielonej diody LED.
- proszę powtórnie zainicjalizować układ pomiarowy (TPM1) i podłączyć kabelkiem jego wejście (PTA0) do wyjścia PWM (PTB9) w celu odczytania zawartości rejestru.
- proszę zmodyfikować układ pomiarowy tak, aby mierzył częstotliwość i wypełnienie badanych sygnałów.

6. Odtwarzacz muzyki w trybie PWM

Idea tego zadania polega na generowaniu fali prostokątnej składającej się z impulsów o stałym okresie, ale zmiennym wypełnieniu reprezentującym wartość próbki sygnału. Taki przebieg podany na głośnik ulega filtracji o charakterze dolno-przepustowym lub pasmowo-przepustowym uśredniając wartość napięcia, co przypomina działanie wzmacniacza klasy D. Zamiana sygnału cyfrowego na analogowy odbywa się w sposób przedstawiony na rysunku 9.



Rysunek 9. Przebiegi kolejno sygnałów: niebieski - PWM o częstotliwości f_2 i wypełnieniu zmienianym z częstotliwością f_3 , zielony - wartość średnia napięcia na wyjściu, fioletowy - wartość średnia po filtrze.

Sygnał PWM o częstotliwości f_2 generowany jest przy pomocy licznika ustawionego na częstotliwość f_1 . Wypełnienie przebiegu jest zmieniane z częstotliwością f_3 . Dlatego też wartość średnia napięcia za filtrem również będzie miała częstotliwość f_3 . Im więcej okresów sygnału PWM „zmieścimy” w jednym okresie sygnału wyjściowego tym lepszy efekt osiągniemy.

Ćwiczenie przedstawia sposób wykorzystania modulacji PWM do generowania dźwięku przy pomocy kodowania PCM (*Pulse-Code Modulation*). Dźwięk zapisany jest przy pomocy próbek w rozdzielczości 8 bitów. Częstotliwość próbkowania wynosi 4.08 [kHz]. Takie parametry zostały wybrane ze względu na ograniczoną ilość pamięci do zapamiętania utworu. Dla 1 sekundy utworu ilość próbek będzie wynosiła 4080.

Mając zadany główny zegar taktujący moduł TPM możemy dostosować rejestry tak, aby otrzymać przybliżoną częstotliwość próbkowania.

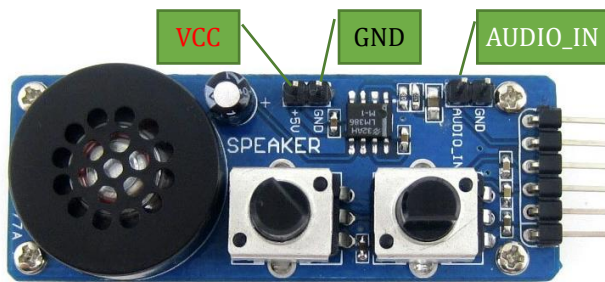
$$41.9\text{MHz} \div 4 \div (255 + 1) \div 10 \approx 4.1\text{kHz}$$

gdzie:

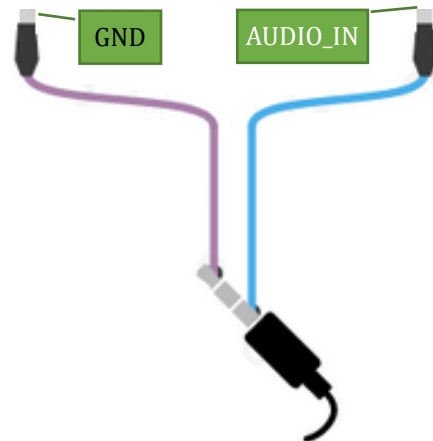
- PS = 4 – preskaler w rejestrze TPMx_SC
- CMOD = 255 – ponieważ próbki dźwięku są 8 bitowe
- UPSAMPLING = 10 – wykorzystanie dodatkowego kanału pozwala na zwiększenie jakości generowanego sygnału ($f_2 / f_3 = 10$)

Zadanie wymaga podłączenia do płytki FRDM-KL05Z odbiornika dźwięku. Moduł głośniczka proszę podłączyć w następujący sposób:

FRDM-KL05	Waveshare Analog Test Board
PTB7	AUDIO_IN
+3.3V	VCC
GND	GND



Rysunek 10. Widok na płytkę z głośniczkiem.



Rysunek 11. Sposób podłączenia słuchawek do FRDM-KL05Z.

W przypadku braku głośniczka można użyć analogowych słuchawek.

ToDo

- 4.1 Otwórz plik *main.c* i zamień plik nagłówkowy „*tpm.h*” na „*tpm_pcm.h*”.
- 4.2 Proszę wyłączyć inicjalizację modułu TPM0 z poprzednich zadań w pliku *main.c* (zakomentować funkcje *TPM0_Init_PWM*), oraz zablokować wywoływanie funkcji *TPM0_SetVal*
- 4.3 W pliku *main.c* włącz inicjalizację modułu TPM0 w nowej roli (*TPM0_Init_PCM*) oraz aktywuj w pętli głównej włączanie melodii za pomocą funkcji *TPM0_PCM_Play*.
- 4.4 Zbuduj projekt i załaduj program na płytkę FRDM-KL05. Po prawidłowym wykonaniu powyższych czynności układ powinien zachowywać się w następujący sposób:
 - po resecie następuje jednorazowe odtworzenie fragmentu piosenki,
 - dotknięcie płytki dotykowej wywołuje odtworzenie fragmentu piosenki.