

Лабораторна робота № 6

Вивчення системи переривань мікроконтролерів PSoC 6 фірми Cypress

Мета роботи: Ознайомитися з архітектурою переривань мікроконтролерів PSoC 6, джерелами і типами переривань, налаштуванням переривань, алгоритмом опрацювання переривань.

Теоретичні відомості

Розглянемо проект, в якому виконується генерація періодичних переривань за допомогою компоненти таймера/лічильника широтно-імпульсного модулятора (TCPWM - timer/counter pulse-width modulation) в режимі таймера/лічильника для мікроконтролера PSoC 6.

У цьому проекті налаштовується екземпляр компоненти TCPWM в режимі неперервного сумуючого лічильника для створення періодичного переривання. Процесор переходить у сплячий режим. Він прокидається кожного разу, коли відбувається переривання, і повертається в режим сну після обслуговування переривання. Обробник переривання просто перемикає світлодіод. Можна змінити період таймера, змінивши макрос `TIMER_PERIOD_MSEC` у файлі `main_cm4.c`. На рис. 6.1 показано схему PSoC Creator для цього прикладу проекту.

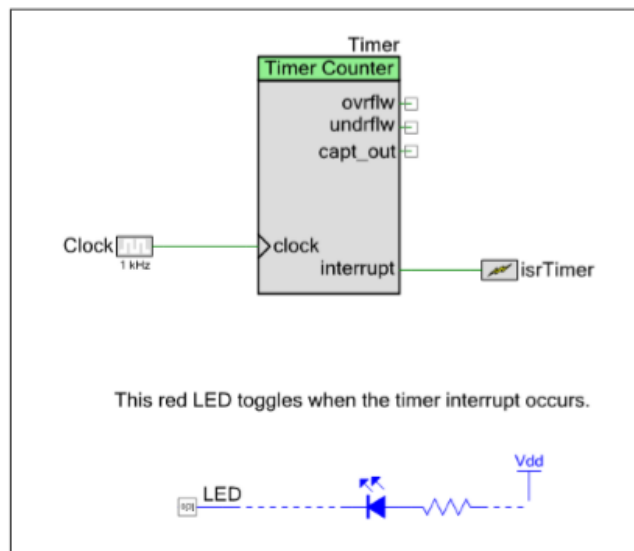


Рис. 6.1. TopDesign Schematic

Основні кроки, які потрібно виконати в цьому проекті:

1. під'єднати плату стенду CY8CKIT-062-BLE до USB – порту комп'ютера.
2. створити проект і запрограмувати його на мікроконтролері PSoC 6 (вибрати Debug > Program).
3. червоний світлодіод LED переключатиметься з періодом 1 сек.

4. змінити період таймера на 0.5 сек, змінивши макрос `TIMER_PERIOD_MSEC` в файлі `main_cm4.c`, запрограмувати мікроконтролер і переконатися, що світлодіод тепер засвічується з частотою 2 Гц.

Компоненти, що використовуються в цьому проекті:

- вивід LED, налаштований на цифровий вихід;
- екземпляр таймера 1 блоку TCPWM;
- периферійний дільник тактових імпульсів Clock 1;
- переривання `isrTimer` немає (він тільки генерує конфігурацію для переривання TCPWM).

Параметри.

Налаштування на рис. 6.2, рис. 6.3 та рис. 6.4 показують конфігурацію компонента з виділеними зміненими налаштуваннями.

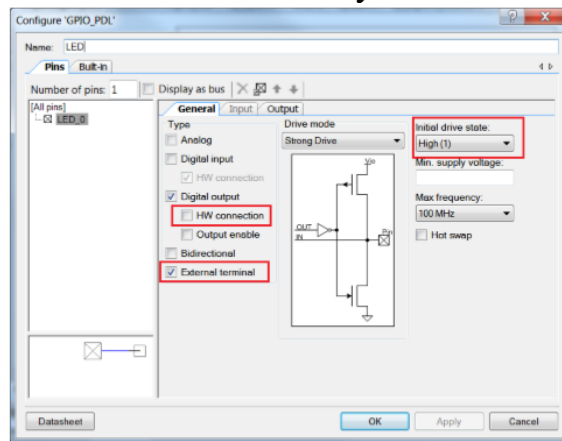


Рис. 6.2. Конфігурація компонента цифрового виводу, налаштованого на вихід

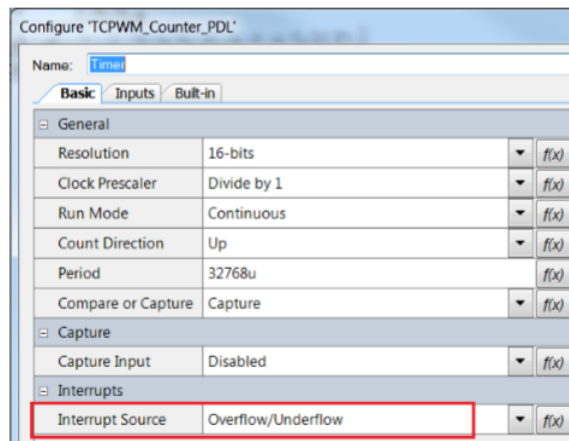


Рис. 6.3. Конфігурація лічильника таймера (TCPWM)

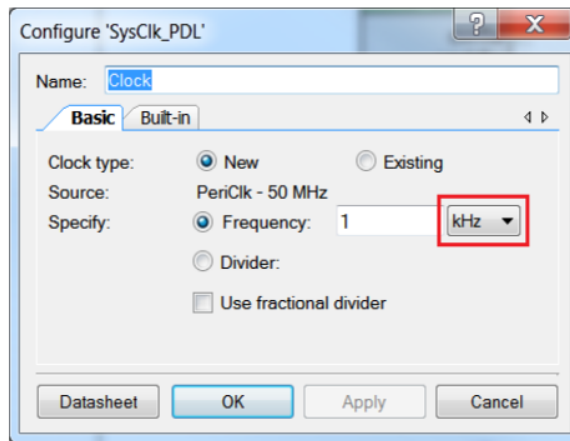


Рис. 6.4. Конфігурація компоненти тактових імпульсів
На рис. 6.5. приведено призначення виводів для проекту.

	Name	Port	Pin	Lock
<input checked="" type="checkbox"/>	LED	P0[3]	E3	<input checked="" type="checkbox"/>

Рис. 6.5. Призначення виводів

На рис. 6.6. зображено призначення переривань для цього проекту. PSoC Creator призначає переривання для ядра CM4 за замовчуванням.

CE22016...pt.cydwr						
Instance Name	Interrupt Number	ARM CM0+ Enable	ARM CM0+ Priority (1 - 3)	ARM CM0+ Vector (3 - 29)	ARM CM4 Enable	ARM CM4 Priority (0 - 7)
isrTimer	90	<input type="checkbox"/>	--	--	<input checked="" type="checkbox"/>	7

Рис. 6.6. Призначення переривань

Алгоритм програми:

```
#include "project.h"
```

```
/* Макроси для налаштування функціональності проекту */
```

```
#define TIMER_PERIOD_MSEC 1000U /* Період таймера в мсек */
```

Функція: `void TimerInterruptHandler(void)`. Ця функція виконується, коли спрацьовує переривання по таймеру. Вона виконує просто переключення ON/OFF світлодіода.

```
void TimerInterruptHandler(void)
{
    /* Очистка переривання підрахунку терміналу */
    Cy_TCPWM_ClearInterrupt(Timer_HW, Timer_CNT_NUM, CY_TCPWM_INT_ON_TC);

    /* Переключення світлодіоду LED */
    Cy_GPIO_Inv(LED_Red_PORT, LED_Red_NUM);
}

/* Ініціалізація таблиці векторів переривань за допомогою
 * обробника переривань по таймеру,
 * адреса і призначення пріоритету.
 */
```

```

Cy_SysInt_Init(&isrTimer_cfg, TimerInterruptHandler);
NVIC_EnableIRQ(isrTimer_cfg.intrSrc);
/* Дозвіл переривань ядра */
__enable_irq();
/* Дозвіл глобальних переривань */

/* Запуск компоненти TCPWM в режимі таймера/лічильника. The return value of the
 * Значення, що повертається:
 * функція вказує чи дійсні аргументи чи ні.
 * Тут не використовується.
 */
(void)Cy_TCPWM_Counter_Init(Timer_HW, Timer_CNT_NUM, &Timer_config);
Cy_TCPWM_Enable_Multiple(Timer_HW, Timer_CNT_MASK);
/* Дозвіл екземпляра лічильника */

/* Встановлення періоду таймера в мілісекундах.
 * Для підрахунку N циклів, період повинен бути встановлений на N-1.
 */
Cy_TCPWM_Counter_SetPeriod(Timer_HW, Timer_CNT_NUM, TIMER_PERIOD_MSEC - 1);

/* Програмний запуск екземпляра лічильника.
 * Це необхідно виконати, якщо жодний інший апаратний вхідний сигнал не
 підключений до компоненти,
 * щоб діяти як джерело запуску.
 */
Cy_TCPWM_TriggerStart(Timer_HW, Timer_CNT_MASK);

for(;;)
{
    /* Переведення процесора в сплячий режим для економії енергії. Put the CPU
into sleep mode to save power. CPU wakes up whenever
    * Процесор прокидається кожного разу, коли відбувається переривання по
таймеру
    * і знову переходить в режим сну після обслуговування переривання ISR.
    */
    Cy_SysPm_Sleep(CY_SYSPM_WAIT_FOR_INTERRUPT);
}

```

Функції, що використовуються в цьому проекті.

```

void Cy_TCPWM_ClearInterrupt(TCPWM_Type *base, uint32_t
cntNum, uint32_t source)

```

Ця функція очищує активне джерело переривань.

Параметри:

Base	– вказівник на екземпляр TCPWM;
cntNum	– номер екземпляру лічильника в вибраному TCPWM;
source	- джерело для очистки.

```

void NVIC_ClearPendingIRQ (IRQn_Type IRQn);

```

Ця функція видаляє стан очікування заданого переривання IRQn. IRQn не може бути від'ємним числом.

Параметри:

[in] IRQn Interrupt number

Зауваження:

- реєстри, які керують станом переривань: SETPEND, CLRPEND.
- переривання може мати статус очікування, хоча воно є неактивним.

```
void NVIC_EnableIRQ (IRQn_Type IRQn);
```

Ця функція дозволяє вказане переривання IRQn для конкретного пристрою. IRQn не може бути від'ємним числом.

Параметри:

[in] IRQn Interrupt number

Зауваження:

- реєстри, які керують дозволом та заборонаю переривань: SETENA, CLRENA.

- кількість підтримуваних переривань залежить від реалізації проекту кристалу і може бути зчитана з реєстру типів переривань (ICTR - Interrupt Controller Type Register) з дискретністю 32:

ICTR[4:0]=

- 0 – підтримується 32 переривання;
- 1 – підтримується 64 переривання.
- 2 – ...

Опис літератури:

CE220169 – PSoC 6 MCU: Periodic Interrupt using TCPWM. Електронний ресурс. Режим доступу: <https://www.cypress.com/file//385846/download>

Завдання.

1. Реалізувати проект засвічування світлодіодів LED8 та LED9 стенду PSoC 6 BLE PIONER KIT. з використанням переривань по таймеру та переривань GPIO, до якого під'єднано перемикач SW_2. Алгоритм засвічування світлодіодів: при включенні живлення стенду світлодіод LED8 блимає з частотою 1 Гц 8 разів. Після цього процесор переходить в режим "глибокого сну". З цього режиму його виводить натискання на кнопку SW_2. При настанні переривання починає блимати 4 рази світлодіод LED9 з частотою 2 Гц.

Періоди блимання задаються з допомогою переривань таймера TCPWM.