

第六章

脈波寬度調變

6.1 脈波寬度調變控制器 (PWM)

類比 (Analog) 信號的主要特色是連續可變的特性，因此不論是時間或是振幅皆有無限精密的解析度 (infinite resolution)，類比信號容易直接用於類比電路的控制信號，但是類比信號容易隨時間飄移不穩定，容易受到雜訊的影響，而且功率消耗大都是其缺點。為此數位控制因應而生，脈波寬度調變 (Pulse Width Modulation, PWM) 就是一種強有力的控制技術 (<http://goo.gl/yJmL6A>)，可以在不使用 ADC 轉換器的情況下取代類比信號的技術，只要調變頻率夠高 PWM 的輸出信號便可接近完全取代類比信號。PWM 的應用範疇極廣，包含信號量測、通訊、馬達控制、甚至簡單的聲音播放等等。

產生 PWM 信號有多種方式，有類比及數位不同的方式，一般常見類比的產生方式是藉助鉅齒波或三角波與類比信號經由比較器比較振幅大小藉以調整脈波寬度決定工作週期，至於調變的頻率則由鉅齒波或三角波的頻率決定。至於數位產生方式則利用計時／計數器決定 PWM 波形的調變頻率與工作週期 (Duty Cycle)。至於 PWM 波形如何轉換回復成類比信號，一般最簡單的方式就是透過 RC 或 LC 濾波低通電路去除載波調變頻率而得。一般而言調變頻率大約落在 1KHz ~ 200KHz (<http://goo.gl/yJmL6A>)，雖說調變頻率愈高，PWM 波形愈趨近於類比信號，但是高頻方波實際上也有相當多諧波的問題存在，因此一般 PWM 頻率視應用場合而定。

6.2 WT59F064 脈波調變寬度控制器

6.2.1 WT59F064 脈波調變寬度控制器特性

WT59F064 系統晶片提供兩組 PWM 控制器：PWM0 與 PWM1，每組各可輸出 4 個解析度達 12 位元的 PWM 信號，控制模組與 PWM 輸出腳位及 GPIO 腳位之間的關係如表 6.2.1-1 所示，WT59F064 中 PWM 的產生是

借助計時器完成的數位產生模式。其特性歸納如下：

- 計時時脈採用 MCU 系統時脈
- 12 位元解析度 PWM 輸出，共有 8 個 PWM 輸出。
- 當 MCU 時脈採 12MHz 時，PWM 輸出波形頻率範圍為 1.907 Hz ~ 6.000 MHz，當 MCU 時脈採 24MHz 時 PWM 輸出波形頻率範圍為 3.815 Hz ~ 12.000 MHz。
- PWM 波形的週期範圍 2 ~ 4096 個 PWM 工作時脈
- PWM 工作週期（Duty Cycle）範圍 0/4096 ~ 4095/4096

表 6.2.1-1 PWM 控制模組、PWM 輸出腳位與共用 GPIO 腳位

PWM 模組	PWM 輸出腳位	共用 GPIO 接腳腳位	其他共用腳位
PWM1 (位址：0x001F6000 ~ 0x001F63FF)	PWM[0]	GPIO_B10	UART2_TX
	PWM[1]	GPIO_B11	UART2_RX
	PWM[2]	GPIO_D2	
	PWM[3]	GPIO_D3	
PWM0 (位址：0x001F5C00 ~ 0x001F5FFF)	PWM[4]	GPIO_C13	I2S_LRCLK、TMR2_capi0
	PWM[5]	GPIO_B5	I2S_BCLK、TMR2_capi1
	PWM[6]	GPIO_B0	ADC8, I2S_DI
	PWM[7]	GPIO_B1	ADC9、I2S_DO、HDMI_CEC

6.2.2 WT59F064 PWM 暫存器與使用設定

表 6.2.2-1 列出 PWM1 模組中所有暫存器的索引位址與功能設定，而 PWM1 模組控制 PWM[0] ~ [3] 等四個 PWM 輸出，PWM_EN[7] ~ [4] 分別用來啟動 PWM[3] ~ PWM[0] 的功能。

表 6.2.2-1 WT59F064 PWM 暫存器與功能設定

索引位址	預設值	讀／寫	位元	名稱	功能設定
0x00	0	R/W	7:4	PWM_EN	1：啟動 PWM[x]功能 0：使 PWM[x]功能失效
0x04	0	R/W	1:0	PWM_BAS_CLK	PWM 基頻選擇 00：MCU clock

»» 微處理器應用與實作：C 語言與 Andes MCU 系列

索引位址	預設值	讀／寫	位元	名稱	功能設定
					01：MCU clock / 2 10：MCU clock / 3 11：MCU clock / 12
0x08	0	R/W	6:0	PWM_CLK[0] Select	PWM[0]工作時脈選取=PWM 基頻 / (PWM_CLK[0]+1)
0x0C	0	R/W	6:0	PWM_CLK[1] Select	PWM[1]工作時脈選取 (設定同上)
0x10	0	R/W	6:0	PWM_CLK[2] Select	PWM[2]工作時脈選取 (設定同上)
0x14	0	R/W	6:0	PWM_CLK[3] Select	PWM[3]工作時脈選取 (設定同上)
0x18	200	R/W	11:0	PWM[0]	PWM[0]工作週期時脈個數選 取，假設 PWM[0]的週期是 4096 個 PWM 工作時脈，工 作週期=PWM[0][11:0] / 4096
0x1C	200	R/W	11:0	PWM[1]	PWM[1]工作週期時脈個數選 取，假設 PWM[1]的週期是 4096 個 7PWM 工作時脈，工 作週期=PWM[1][11:0] / 4096
0x20	200	R/W	11:0	PWM[2]	PWM[2]工作週期時脈個數選 取，假設 PWM[2]的週期是 4096 個 7PWM 工作時脈，工 作週期=PWM[2][11:0] / 4096
0x24	200	R/W	11:0	PWM[3]	PWM[3]工作週期時脈個數選 取，假設 PWM[3]的週期是 4096 個 7PWM 工作時脈，工 作週期=PWM[3][11:0] / 4096
0x28	3FF	R/W	11:0	PERIOD_[0]	PWM [0]的週期選擇 =(PERIOD_[0] [11:0] +1)* PWM[0]工作時脈週期
0x2C	3FF	R/W	11:0	PERIOD_[1]	PWM [1]的週期選擇 =(PERIOD_[1] [11:0] +1)* PWM[1]工作時脈週期
0x30	3FF	R/W	11:0	PERIOD_[2]	PWM [2]的週期選擇 =(PERIOD_[2] [11:0] +1)* PWM[2]工作時脈週期
0x34	3FF	R/W	11:0	PERIOD_[3]	PWM [3]的週期選擇 =(PERIOD_[3] [11:0] +1)* PWM[3]工作時脈週期

(WT59F064 Flash Memory Type 32-bit Microcontroller Data Sheet)

PWM_BAS_CLK 用來設定整個 PWM1 模組的基本時脈(PWM 基頻), 再由 PWM_CLK[0] ~ PWM_CLK[3] 各別設定 PWM[0] ~ PWM[3] 的工作時脈 (=PWM 基頻/(PWM_CLK[x]+1))。PERIOD_[x] 暫存器用來選定 PWM 輸出波形的頻率 (或週期), 換句話說, PWM 輸出波形週期 = 工作時脈週期 * (PERIOD_[x]+1) 。PWM[x] 決定每個輸出的 PWM 信號中的工作週期 (Duty Cycle), 工作週期的設定也是以設定每個 PWM 波形中工作時的脈波個數決定之, 換句話說, 工作週期 = PWM[x] / (PERIOD_[x]+1) 。要規劃使用 WT59F064 PWM 時首先必須要確認 MCU 時脈, 依序是 PWM_BAS_CLK、PWM_CLK[x]、PERIOD_[x] 及 PWM[x] 等暫存器的使用設定, 最後是 PWM_EN 設定啟動 PWM 的功能。此外設定需要注意以下幾點:

- 若 PERIOD_[x]=0, 則其 PWM 輸出波形永遠維持高電位。
- PERIOD_[x] 不為 0 時, 若 PWM[x]=0 則 PWM 輸出波形永遠維持低電位。
- 若 PWM[x] > PERIOD_[x], 則其 PWM 輸出波形永遠維持高電位。

表 6.2.2-2 是假設 MCU 系統時脈使用 24MHz 方波。

表 6.2.2-2 PWM 基頻設定與最高、最低 PWM 輸出頻率表

PWM_BAS_CLK	PWM 基頻	PWM 輸出頻率 (min./max.)
0x00	24MHz	45.78Hz/12MHz
0x01	12MHz	22.89Hz/6MHz
0x10	8 MHz	15.26Hz/4MHz
0x11	2 MHz	3.81Hz/1MHz

欲達最高頻率輸出 PWM_CLK[x] 設為 0, PERIOD_[x] 設為 1, 達最低頻率輸出 PWM_CLK[x] 設 0x7F, PERIOD_[x] 設為 0xFFFF, 因此可得表 6.2.2-2。

6.2.3 WT59F064 PWM 實作

此實作單元旨在熟悉 PWM 的設定使用，以便可以應用在不同場合，例如一些機器人的手臂運動控制或舵機控制，在 ADP-WT59F064 平台上蜂鳴器接在 GPIO_D2 輸出腳位上，因此在這個實驗就以 PWM[2] 輸出波形驅動蜂鳴器做為實驗範例，本單元提供兩個實作專案：一個是簡單的 ADP-WT59F064-PWM 專案送出固定頻率的 PWM 波形使蜂鳴器產生急促的警示聲；另一個 ADP-WT59F064-PWM-TONE 專案則是送出不同音符的聲音，如同音樂課練習發音，首先練習簡單的 ADP-WT59F064-PWM 專案：

● ADP-WT59F064-PWM 實作

表 6.2.1-1 顯示與 GPIO_D2 共用腳位是 PWM[2]，因此本實驗完全使用 PWM[2] 進行實作，ADP-WT59F064-PWM 專案中完全以 main.c 完成，一開始進入 main() 時先切換 GPIO_D2 輸出接腳為特殊功能腳位，提供給 PWM[2] 使用，將 MCU 時脈除以 12 作為 PWM 模組的基頻 (2MHz)，並將 PWM_CLK2 設為 0 使得送入 PWM 模組的基頻就是 PWM 真正的工作時脈，為得到警示的聲音因此將 PWM 輸出頻率設為 4KHz，所以 PERIOD_2 設為 $(2M/4K - 1) = 499$ ，為了將 PWM 的工作週期設為 50% 就必須將 PWM2 設為 $500/2 = 250$ 。以下程式碼有詳細說明，設定完畢後記得啟動 PWM[2] 的輸出，因此 PWM_EN 設為 0x40，為了達到警示效果因此必須有短暫停止輸出，所以在 while 迴圈以 $PWM_EN \wedge 0x40$ 切換輸出狀態。

```
//切換 GPIO_D2 腳位為 PWM[2]的輸出腳位
```

```
GPIO_ACT_PD = 0xFFFB;
```

```
//因為系統時脈=24MHz，因此由表 6.2.2-2 可知將 PWM_BAS_CLK 設為 3
```

```
//可以得到 PWM 模組的基頻 24MHz /12 = 2MHz
```

```
PWM_BAS_CLK = 0x3;
```

```
//當 PWM_CLK2 設為 0 時，表示 PWM[2]的工作時脈= 2MHz/(0+1) = 2MHz
```

```

PWM_CLK2 = 0;
//PWM[2]輸出波形的週期為  $1/2M * (499+1) = 250$  微秒，也就是 4KHz 的波形
PERIOD_2 = 499;
//輸出高準位的時間設為 250 個工作時脈，因此等於 duty cycle=  $(250/500) = 50\%$ 
PWM2 = 250;
//啟動 PWM[2]輸出
PWM_EN = 0x40;

```

以下是匯入專案執行步驟：

- Step 1.** 將滑鼠移入專案瀏覽 (Project Viewer) 子視窗中點擊滑鼠右鍵後，選「 import 」進行匯入 ADP-WT59F064-PWM 專案。
- Step 2.** 打開 General 選擇「 Existing Projects into Workspace 」後，按 Next >。
- Step 3.** 因為匯入的是專案壓縮檔因此點擊「 Select archive file 」後再打開點擊「 Browse 」選項選擇要匯入的專案。
- Step 4.** 點選 ADP-WT59F064-PWM.zip 這個專案壓縮檔後，按開啟舊檔(O)。
- Step 5.** 開啟專案壓縮檔後，可見到專案檔的完整名稱 ADP-WT59F064-PWM，在專案檔名稱左方勾選者表示要匯入此專案，然後按點擊下方「 Finish 」選項。
- Step 6.** 透過 Build Project 編譯專案檔裡的程式碼，可藉此檢查程式碼是否有錯。先以滑鼠左鍵點選「 ADP-WT59F064-PWM 」使其出現反白，然後按滑鼠右鍵後再點選「 Build Project 」進行編譯連結的工作。
- Step 7.** 把 Step 6. 中建置完成的 Debug\output\ADP-WT59F064-PWM.bin 檔透過燒錄程式 ISP_WT59F064.exe 燒錄到晶片中。在專案瀏覽子視窗中對 ADP-WT59F064-PWM 按右鍵出現下拉式選單，再點擊「 Flash Burner 」以開啟燒錄視窗。
- Step 8.** 燒錄程式視窗中可以見到燒錄程式碼檔名，要燒錄到晶片中的程式碼存放路徑及檔案名稱，燒錄前請再次確認是否正確。透過點擊「 Auto 」一次完成整個燒錄過程如，當出現「 Verify Successful 」，表示燒錄成功。

»» 微處理器應用與實作：C 語言與 Andes MCU 系列

Step 9. 程式執行時可以聽到蜂鳴器發出尖銳的警示聲。圖 6.2.3-1 中手指的即是執行本實驗發出警示音的蜂鳴器。

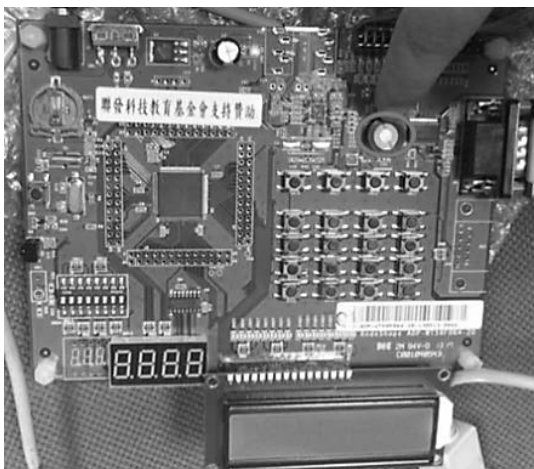


圖 6.2.3-1 蜂鳴器發出急促的警示聲

● ADP-WT59F064-PWM-TONE 實作

上一個實作是以固定週期輸出 PWM 波形，因此蜂鳴器發出固定頻率的警示聲，此一實作則以音符的頻率依序從低頻開始發聲，原則上音符頻率如表 6.2.3-1，本實驗 PWM 的工作時脈的設定與上一個一致為 2MHz，所以要發出以下音符的頻率，其 PERIOD_2 的設定就是以 2MHz 除以各音符的頻率，但是有些設定值已超過 12 位元 (4095)，因此再將音降 8 度，設定如程式碼中的定義：

```
#define Do (unsigned int) (2*1000000/262/2)
#define Re (unsigned int) 2*1000000/294/2
#define Mi (unsigned int) 2*1000000/330/2
#define Fa (unsigned int) 2*1000000/349/2
#define Sol (unsigned int) 2*1000000/392/2
#define La_H (unsigned int) 2*1000000/440/2
```



```
#define Si_H (unsigned int) 2*1000000/494/2
#define Do_H (unsigned int) 2*1000000/(262*2)/2
```

在 `main()` 中只負責設定要播放的音符頻率及時間長短，真正設定 PWM[2] 的任務則交由 `playtone (tempo, duration)` 執行，其設定的方式與 ADP-WT59F064-PWM 專案的方式一致不再重複說明。

表 6.2.3-1 音符頻率

Do	262Hz
Re	294Hz
Mi	330Hz
Fa	349Hz
Sol	392Hz
La_H	440Hz
Si_H	494Hz
Do_H	524Hz

6.3 ADP-XC5-for-N801-S 脈波寬度調變控制器 (PWM)

6.3.1 ADP-XC5-for-N801-S PWM 特性

ADP-XC5-for-N801-S 提供一個 PWM 模組內含兩個 PWM 輸出通道，各自由不同的暫存器控制設定其輸出，每個通道使用的基本時脈由 PLL/DLL 控制暫存器 1 中 PWMCLKDIV 決定，大約為 $5\text{MHz}/(\text{PWMCLKDIV}+1)$ 。因為 PWMCLKDIV 重置預設初值為 0，因此在不改初值的情形下送達 PWM 模組基頻時脈大約是 5MHz 方波，此基頻時脈送達 PWM 模組後由兩個通道中的 CTRL 暫存器器各自決定其 PWM 工作時脈的頻率，CTRL 暫存器用來設定除頻計數器的值，這是一個 6 位元的計數器，送到 PWM 模組的基頻時脈經過此計數器的除頻後得到 PWM 的工作時脈。

N801-S SoC PWM 特性如下：

- 2 個 PWM 通道，使用相同的基頻時脈，但各自彼此獨立工作輸出。
- 每個 PWM 通道各自擁有一個除頻用計數器 (6 位元) 及 PWM 頻率及工作週期的控制暫存器，彼此獨立工作輸出。

»» 微處理器應用與實作：C 語言與 Andes MCU 系列

- 10 位元的 PWM 輸出頻率設定。
- 10 位元的工作週期設定。

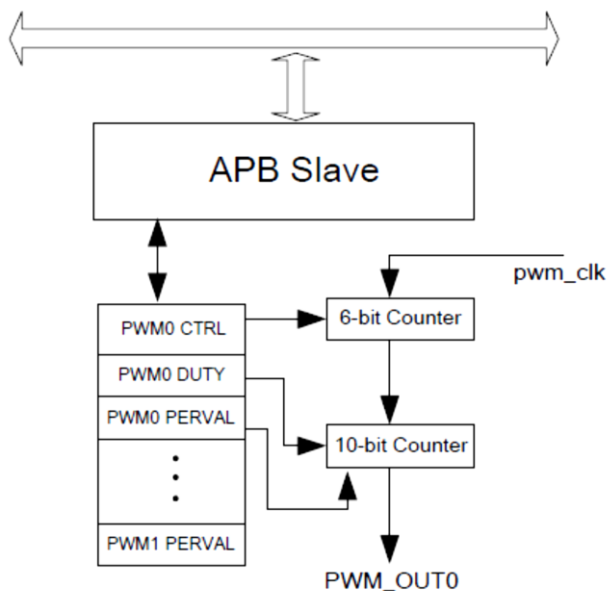


圖 6.3.1-1 PWM 方塊圖 (取自 AndeShape_AG101P_Datasheet)

6.3.2 ADP-XC5-for-N801-S PWM 暫存器說明與使用設定

N801-S SoC 中的 PWM 占用 0x00F11000 ~ 0x00F110FF 的位址空間，並且與 GPIO[30]、GPIO[31] 共用輸出入腳位，使用前須先透過 Multi-Function Port Setting Register (MFPSR) 選擇設定。

表 6.3.2-1 N801-S SoC PWM 暫存器位址空間

PWM 模組	PWM 輸出腳位	共用 GPIO 接腳腳位
PWM(位址： 0x00F11000 ~ 0x00F110FF)	PWM[1]	GPIO[31]
	PWM[0]	GPIO[30]

(透過 Multi-Function Port Setting Register (MFPSR) 選擇)

表 6.3.2-2 N801-S SoC PWM 暫存器索引位址與設定說明

索引位址	暫存器名稱	位元	R/W	功能說明	預設初值
0x00	PWM0CR	5:0	R/W	Prescale : PWM0 Control Registers for first clock output , 將基頻時脈 (pwm_clk) 除此設定值 (PWM0CR[5:0]+1) 即可得 PWM 的工作時脈 PSCLK_PWM0	0x0
0x04	PWM0DCR	10	R/W	FDCYCLE : PWM Full Duty Cycle 0 ; 由 DCYCLE 決定 1 ; PWM0 輸出永遠為”1”	
		9:0	R/W	DCYCLE : PWM0 Duty Cycle Registers for first clock output PWM0DCR[10]=1 時 PWM_OUT0 永遠維持高電位 , PWM0DCR[10]=0 時 , PWM0DCR[9:0] 值決定維持 PWM_OUT0 維持高電位的工作時脈個數	0x0
0x08	PWM0PCR	9:0	R/W	PWM PERVAL : PWM0 Period Control Register for first clock output PWM0PCR[9:0]+1 即 PWM_OUT0 輸出波形的週期	0x04
0x10	PWM1CR	5:0	R/W	PWM1 Control Registers for first clock output , 將基頻時脈 (pwm_clk) 除此設定值 (PWM1CR[5:0]+1) 即可得 PWM 的工作時脈 PSCLK_PWM1	0x0
0x14	PWM1DCR	10	R/W	FDCYCLE : PWM Full Duty Cycle 0 ; 由 DCYCLE 決定 1 ; PWM1 輸出永遠為「 1 」	
		9:0	R/W	DCYCLE : PWM1Duty Cycle Registers for second clock output PWMD0CR[10]=1 時 PWM_OUT1 永遠維持高電位 , PWMD0CR[10]=0 時 , PWM1DCR[9:0] 值決定維持 PWM_OUT1 維持高電位的工作時脈個數	0x0
0x18	PWM1PCR	31:22	R/W	PWM PERVAL : PWM1 Period Control Register for first clock output , 設定同 PWM0 PCR	0x04

»» 微處理器應用與實作：C 語言與 Andes MCU 系列

使用設定依表 6.3.2-2 的說明可完成設定，但是當 PWM0PCR=0 時，PWM_OIUT0 輸出維持高電位除非 PWM0DCR=0，因為此時 PWM_OIUT0 輸出維持低電位。

表 6.3.2-3 PWM 工作平率設定與最高、最低 PWM 輸出頻率

pwm_clk	PWM0CR[5:0]	PWM 輸出頻率 (min./max.)
5 MHz	0	4.882KHz/2.5MHz
5 MHz	31	152.59Hz/156.25KHz

6.3.3 ADP-XC5-for-N801-S PWM 實作

● N8_PWM 實作

本實作旨於熟悉 PWM 控制器的設定與使用，使其應用於各式應用領域。本專案由 main.c pwm.c uartio.c 等組合而成，其中 uartio.c 主要是設定 UART 的初始化，工作原理及設定方式請參閱第五章，pwm.c 執行 pwm_init() 進行 PWM0 的初始化，設定 Prescale 的值可決定 PSCLK_PWM0 的頻率，設定 FDCYCLE=0 及 PWM0DCR[9:0] 決定工作週期共有多少個 PSCLK_PWM0 脈波，設定 PERVAL 的值以決定 PWM0 輸出波形的週期。閱讀程式碼時請注意 pwm_init() 中程式註解說明目前此版本 VEP 提供的 PWM 元件，有關 Prescale、FDCycle、Dcycle、PERVAL 的位置與表 6.3.2-2 不完全相符，

```
#define PWM0CR (*((volatile unsigned int *) (PWM_BASE+0x00)))/Control Registers
#define PWM0DCR (*((volatile unsigned int *) (PWM_BASE+0x04)))/Duty Cycle
#define PWM0PCR (*((volatile unsigned int *) (PWM_BASE+0x08)))/Period Control
void pwm_init(void)
{
    PWM0CR = 0x5 << 26; //Bit 31~26 are used to decide prescale value
    PWM0PCR = 512 << 22; //Bit 31~22 are used to decide PWM period
    //Bit 31~22 are used to decide PWM duty cycle, bit 21 is used to decide full duty
    PWM0DCR = 128 << 22;
}
```

實作步驟採匯入專案目錄方式進行請參考第四章，圖 6.3.3-1 是專案執行時 VEP 上 PWM 元件顯示的訊息，此圖顯示 PWM0 Perval 設為 512，Duty Cycle 設為上述程式碼的內容。

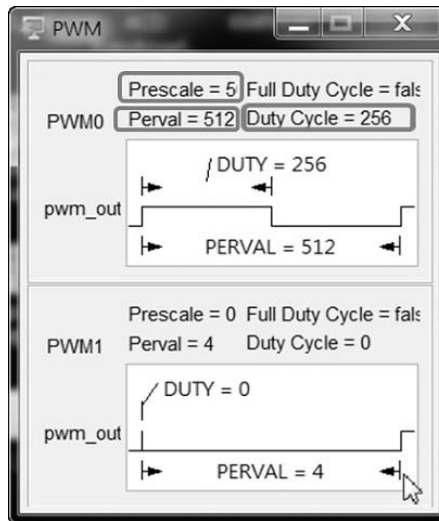


圖 6.3.3-1 PWM 模擬元件顯示 PWM 輸出波形