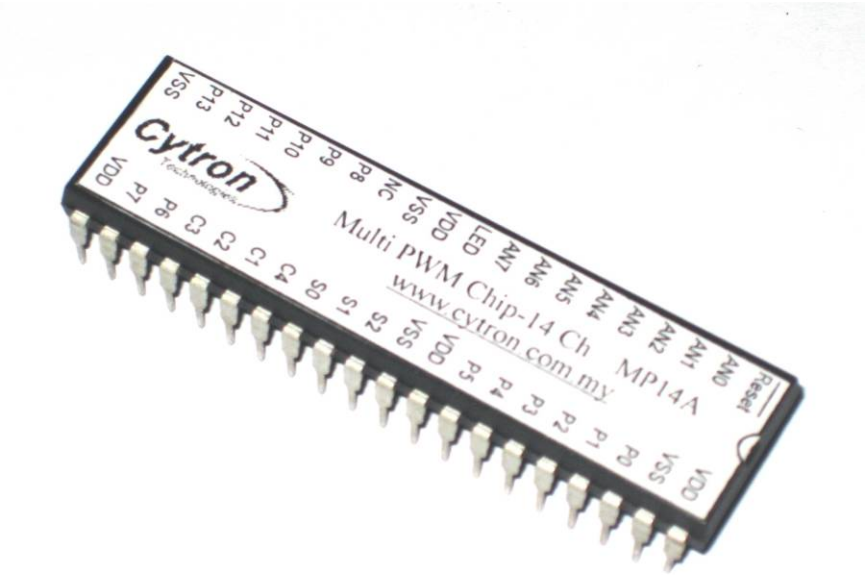
MP14A
Multi PWM chip 14 Channels



User's Manual

V1.0

April 2009

Index

1. INTRODUCTION AND OVERVIEW

1.1 Introduction

PWM is a way of digitally encoding analog signal levels. Through the use of high-resolution counters, the *duty cycle* of a square wave is modulated to encode a specific analog signal level. The PWM signal is still digital because, at any given instant of time, the full DC supply is either fully on or fully off. The voltage or current source is supplied to the analog load by means of a repeating series of on and off pulses. The *on-time* is the time during which the DC supply is applied to the load, and the *off-time* is the periods during which that supply is switched off.

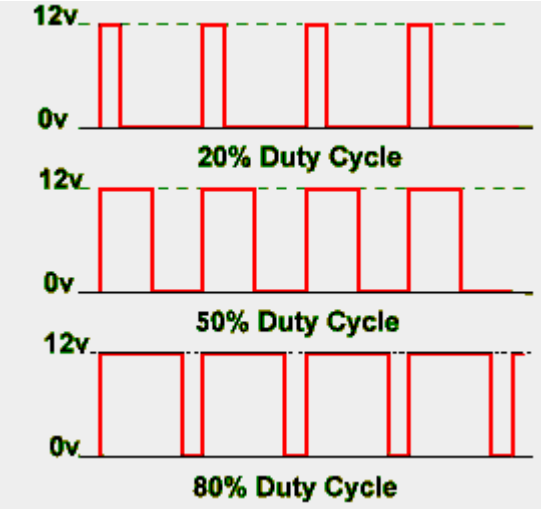
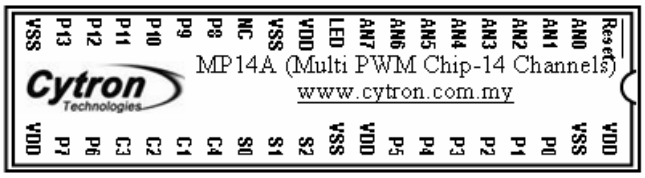


Figure above shows three different PWM signals. The first graph in figure above shows a PWM output at a 20% duty cycle. That is, the signal is 'on' for 20% of the period and 'off' the other 80%. The second and third graph in figure above show PWM outputs at 50% and 80% duty cycle, respectively. These three PWM outputs encode three different analog signal values, at 20%, 50%, and 80% of the full strength. For example, if a system is powered by a 12V power supply, filtering the output signal from the system with a 50% duty cycle PWM signal will give an average output voltage of 6V.

1.2 Overview of MP14A



Cytron Technologies has design multi PWM Chips for PWM output usage. Although there are many microcontrollers include on-chip PWM controllers, but the number of PWM output is limited. For example, PIC16C67 includes only two PWM outputs. For multi PWM chip, it offers more than 2 PWM output channels. Multi PWM chip, MP14A has 14 PWM output channels with 8 bit resolution. User may use this multi PWM chip to provide PWM signal to the device that need PWM control.

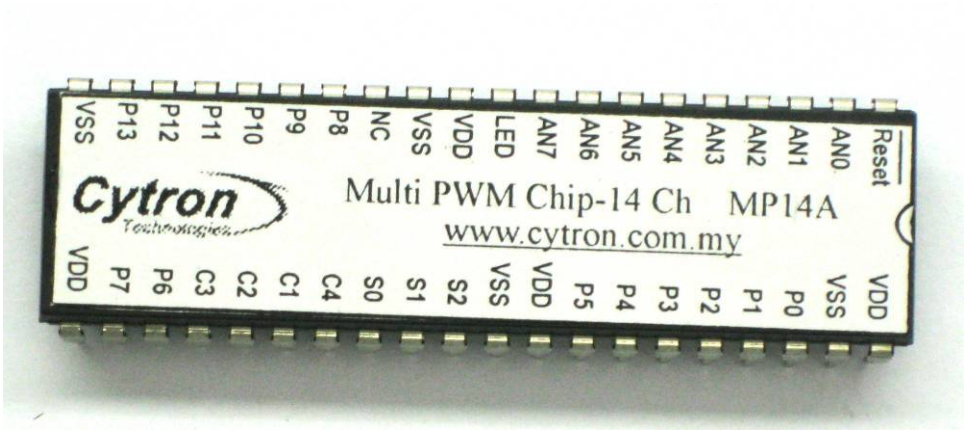
MP14A is available in 40 pin packages. The PWM Frequency of the MP14A is 1.4 KHz. There are analog input pins, PWM output pins, serial communication pins and mode selection pins on MP14A. Analog input pins (AN0-AN7) can connect to analog output from sensor or analog device to control PWM output on MP14A. 14 channels of PWM outputs (P0-P13) can provide PWM signal to the device that need PWM control. MP14A provide 3 types of communication method to communicate with microcontroller unit (MCU), which are UART, SPI and IIC. MP14A also can communicate with computer through UART. The mode selection pins are use to choose mode in MP14A.

For Mode selection, there are 8 modes available for Multi PWM chip. Pin S2 is MSB and pin S0 is LSB.

Mode Selection			Descriptions
S2	**S1**	**S0**	
0	0	0	14 channels PWM output control by 8 ADC input channels.
0	0	1	14 channels PWM output control by microcontroller in UART with baud rate 9600
0	1	0	14 channels PWM output control by microcontroller in UART with baud rate 115200
0	1	1	14 channels PWM output control by ASCII code from computer in UART with baud rate 9600
1	0	0	14 channels PWM output control by ASCII code from computer in UART with baud rate 115200
1	0	1	14 channels PWM output control by microcontroller in IIC
1	1	0	14 channels PWM output control by microcontroller in SPI without SS enable
1	1	1	14 channels PWM output control by microcontroller in SPI with SS enable

2. PACKAGING LIST

Please check the parts and components according to the packing list. If there are any parts missing, please contact us at sales@cytron.com.my immediately.



- 1 x Multi PWM chip – 14 channels (MP14A)

3. PRODUCT SPECIFICATION AND LIMITATIONS

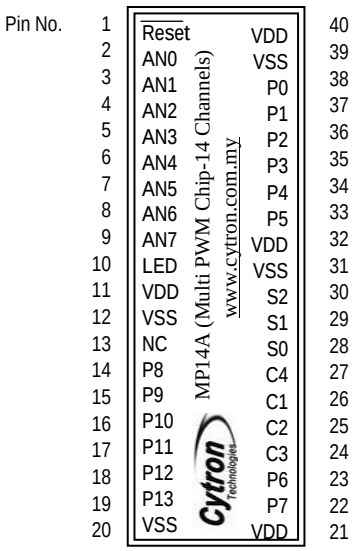
No	Parameters	Max	Typical	Min	Unit
1	Input voltage	5.5	5	2.0	V
2	Output voltage	5.5	5	2.0	V
3	Current- Sink/ Source	-	25	-	mA
4	PWM frequency (8bits)	-	1.4K	-	Hz

Product Feature

No	Feature	Channel	Output/Input of the channel	PWM Feedback
1	PWM output	14	Output = 0-5V	N/A
2	Analog Input*	8	Input = 0-5V	No
3	UART 9600*	1	Input value to control PWM output = 0-255 or 0x00-0xFF	Yes
4	UART 115200*			Yes
5	UART 9600 (ASCII)*		ASCII input to control PWM output = 00-FF	Yes
6	UART 115200 (ASCII)*			Yes
7	IIC*	1	Input value to control PWM output = 0-255 or 0x00-0xFF	Yes
8	SPI with SS enable*	1	Input value to control PWM output = 0-255 or 0x00-0xFF	Yes
9	SPI without SS enable*	1	Input value to control PWM output = 0-255 or 0x00-0xFF	Yes

*select either one mode in a time.

4. CHIP LAYOUT



Pin	Pin No.	Name	Description
Reset	1	Chip Reset	Master Clear (RESET) input. This pin is an active low reset pin to the Multi PWM chip.
VDD	11,21,32,40	5V	Positive supply for logic and I/O pins
VSS	12,20,31,39	Ground	Ground references for logic and I/O pins
AN0	2	Analog input 0	AN0 will control PWM output 0 and PWM output 8.
AN1	3	Analog input 1	AN1 will control PWM output 1 and PWM output 9.
AN2	4	Analog input 2	AN2 will control PWM output 2 and PWM output 10.
AN3	5	Analog input 3	AN3 will control PWM output 3 and PWM output 11.
AN4	6	Analog input 4	AN4 will control PWM output 4 and PWM output 12.
AN5	7	Analog input 5	AN5 will control PWM output 5 and PWM output 13.
AN6	8	Analog input 6	AN6 will control PWM output 6.
AN7	9	Analog input 7	AN7 will control PWM output 7.
P0	38	PWM output 0	P0 is PWM output 0. Maximum PWM value for P0 is 255.
P1	37	PWM output 1	P1 is PWM output 1. Maximum PWM value for P1 is 255.
P2	36	PWM output 2	P2 is PWM output 2. Maximum PWM value for P2 is 255.
P3	35	PWM output 3	P3 is PWM output 3. Maximum PWM value for P3 is 255.
P4	34	PWM output 4	P4 is PWM output 4. Maximum PWM value for P4 is 255.
P5	33	PWM output 5	P5 is PWM output 5. Maximum PWM value for P5 is 255.
P6	23	PWM output 6	P6 is PWM output 6. Maximum PWM value for P6 is 255.
P7	22	PWM output 7	P7 is PWM output 7. Maximum PWM value for P7 is 255.
P8	14	PWM output 8	P8 is PWM output 8. Maximum PWM value for P8 is 255.
P9	15	PWM output 9	P9 is PWM output 9. Maximum PWM value for P9 is 255.
P10	16	PWM output 10	P10 is PWM output 10. Maximum PWM value for P10 is 255.
P11	17	PWM output 11	P11 is PWM output 11. Maximum PWM value for P11 is 255.

P12	18	PWM output 12	P12 is PWM output 12. Maximum PWM value for P13 is 255.
P13	19	PWM output 13	P13 is PWM output 13. Maximum PWM value for P13 is 255.
S2	30	Mode select 2	S2S1S0 is mode selection pin. S2 is MSB for mode selection. Select either 1 or 0 for S2 to select mode for MP14A. Mode selection (S2S1S0) 000 = 8 ADC input channels to control 14 channels PWM output 001 = UART 9600 010 = UART 115200 011 = UART 9600, communicate with computer in ASCII mode 100 = UART 115200, communicate with computer in ASCII mode 101 = IIC (SS/ADD as Address selection pin) 110 = SPI w/o SS 111 = SPI w SS (SS/ADD as SS pin, active low)
S1	29	Mode select 1	S2S1S0 is mode selection pin. S1 is mode select 1. Select either 1 or 0 for S1 to select mode for MP14A. Mode selection (S2S1S0) 000 = 8 ADC input channels to control 14 channels PWM output 001 = UART 9600 010 = UART 115200 011 = UART 9600, communicate with computer in ASCII mode 100 = UART 115200, communicate with computer in ASCII mode 101 = IIC (use Pin C4, SPI SS pin/ IIC address select as Address selection pin) 110 = SPI w/o SS 111 = SPI w SS (use Pin C4, SPI SS pin/ IIC address select as SS pin, active low)
S0	28	Mode select 0	S2S1S0 is mode selection pin. S0 is LSB for mode selection. Select either 1 or 0 for S0 to select mode for MP14A. Mode selection (S2S1S0) 000 = 8 ADC input channels to control 14 channels PWM output 001 = UART 9600 010 = UART 115200 011 = UART 9600, communicate with computer in ASCII mode 100 = UART 115200, communicate with computer in ASCII mode 101 = IIC (use Pin C4, SPI SS pin/ IIC address select as Address selection pin) 110 = SPI w/o SS 111 = SPI w SS (use Pin C4, SPI SS pin/ IIC address select as SS pin, active low)
LED	10	Status LED	Status LED for PWM. LED will turn ON when PWM chip is function.
NC	13	No connect	Do not connect this pin. Leave this pin disconnected.

Communication pins

Pin	Pin No.	Name	Description
C1	26	RX/SDI/SDA	This pin is shared for UART, SPI and IIC communication. RX = UART Receive SDI = SPI data in SDA = IIC data I/O
C2	25	TX/SDO/SCL	This pin is shared for UART, SPI and IIC communication. TX = UART Transmit SDO = SPI data out SCL = Synchronous serial clock input/output for IIC mode
C3	24	SCK	This pin is for SPI communication. Synchronous serial clock input/output for SPI mode.
C4	27	SPI SS pin/ IIC address select	This pin is shared for SPI and IIC communication. SPI slave select input/IIC slave address

5. INSTALLATION AND GETTING STARTED

This section will explain about hardware installation and how to getting start with Multi PWM chip.

5.1 Connection of PWM chip

PWM chip is available to use with or without microcontroller unit (MCU). Figure 5-1 below show example of basic installation hardware for Multi PWM chip 14 channels. S2, S1 and S0 are mode selection pin. These mode selections pins are connect directly to switch or connect to I/O pins of microcontroller unit to select mode of PWM chip. Figure 5-2 is example circuit connection to manually reset and select mode for PWM chip. S2, S1 and S0 are connected to switch to select mode for PWM chip. Figure 5-3 is example circuit connection by using I/O pins of microcontroller unit (MCU) to reset and select mode for PWM chip.

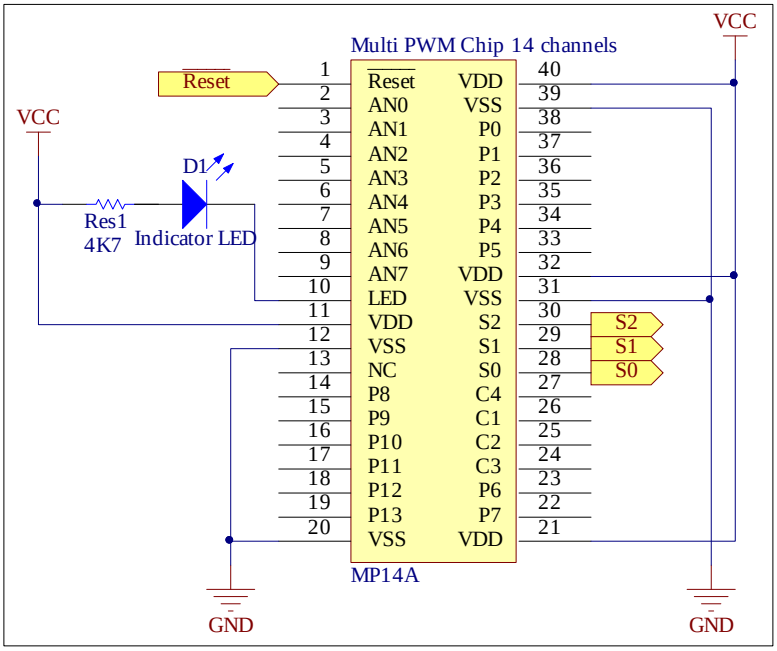


Figure 5-1 Basic connection of PWM chip

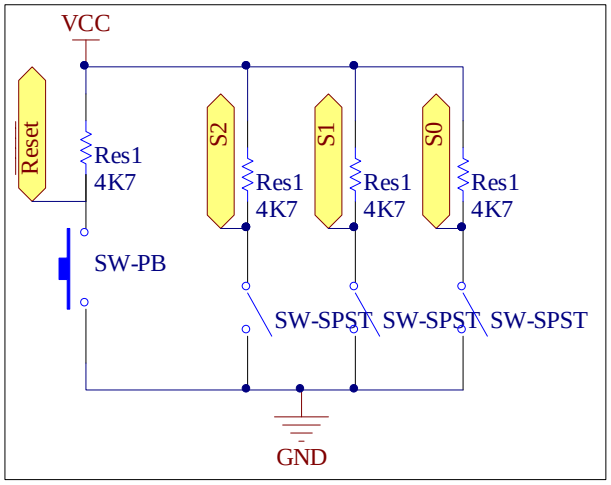


Figure 5-2 Example circuit connection to manually reset and select mode for PWM chip.

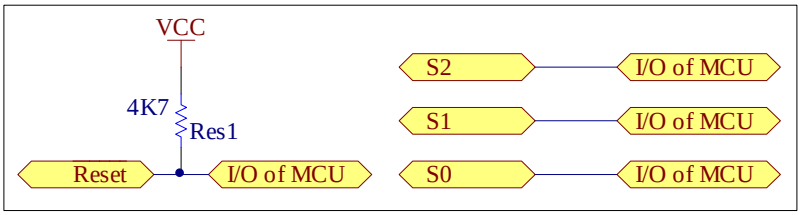


Figure 5-3 Example circuit connection to reset and select mode for PWM chip by using I/O of microcontroller (MCU).

5.2 PWM mode and Protocols

There are 8 modes available for Multi PWM chip. S2, S1 and S0 pin are used to select mode for PWM chip. Modes for PWM chip are:

1. 000 = 14 channels PWM output control by 8 ADC input channels.
2. 001 = 14 channels PWM output control by microcontroller in UART with baud rate 9600
3. 010 = 14 channels PWM output control by microcontroller in UART with baud rate 115200
4. 011 = 14 channels PWM output control by ASCII code from computer in UART with baud rate 9600
5. 100 = 14 channels PWM output control by ASCII code from computer in UART with baud rate 115200
6. 101 = 14 channels PWM output control by microcontroller in IIC
7. 110 = 14 channels PWM output control by microcontroller in SPI without SS enable
8. 111 = 14 channels PWM output control by microcontroller in SPI with SS enable

This section will explain about PWM modes and the communication protocol for UART, SPI and IIC.

a) Mode1 : 8 ADC input channels to control 14 channels PWM output

This mode will use analog input pin (AN0-AN7) to control PWM output pin (P0-P13). For this mode, multi PWM chip can use without microcontroller (MCU).

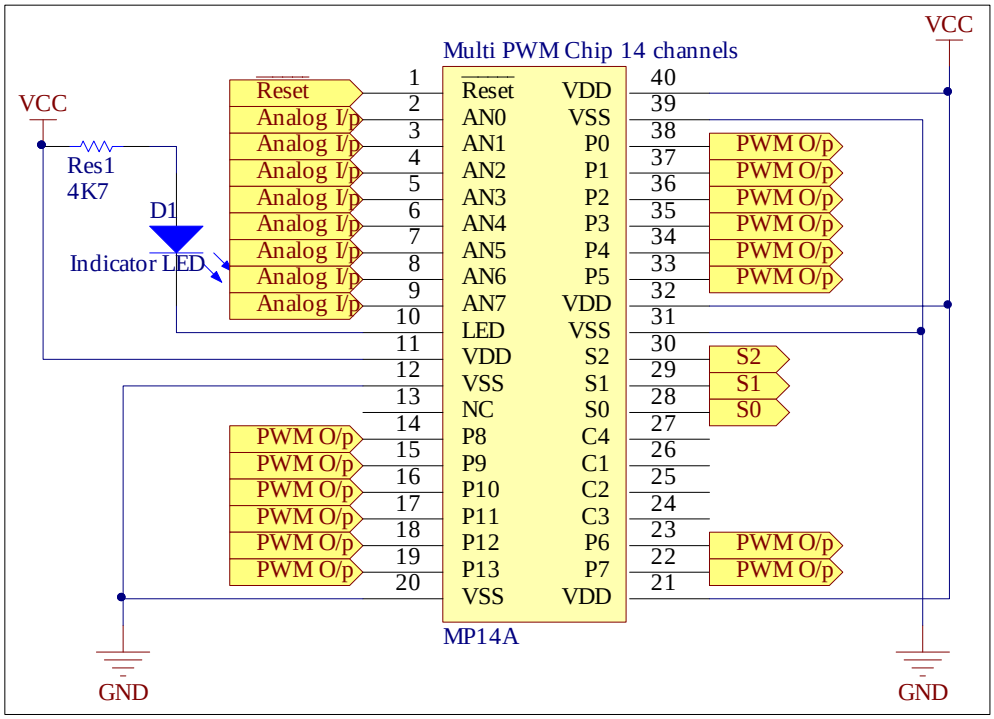


Figure 5-4a Example circuit connection of PWM Chip for mode 1

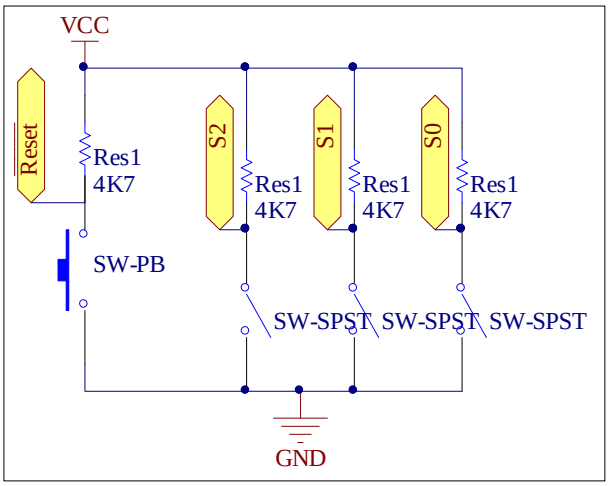


Figure 5-4b Example circuit connection to manually reset and select mode for PWM chip.

Figure 5-4 is example circuit connection for manually reset and select mode for PWM chip. S2, S1 and S0 are connecting to switch for PWM chip mode selection. To operate in analog mode, S2, S1 and S0 pin need to set to low (provide the pins with 0V or digital value '0').

No	Mode Select Pin	Digital Value / Voltage
1	S2	0 / 0V
2	S1	0 / 0V
3	S0	0 / 0V

From schematic in Figure 5-4a, 8 ADC input channel on multi PWM chip (MP14A) will control 14 channels PWM output. Some ADC input channel will control 2 PWM output channels at the same time. For example ADC channel0 will control PWM0 and PWM8. PWM value for PWM8 is equal to 255 minus value of PWM0. Below is 8 ADC input channels and their PWM output channels.

1. ADC channel 0 control PWM0 and PWM8. (PWM8 = 255-PWM0)
2. ADC channel 1 control PWM1 and PWM9. (PWM9 = 255-PWM1)
3. ADC channel 2 control PWM2 and PWM10. (PWM10 = 255-PWM2)
4. ADC channel 3 control PWM3 and PWM11. (PWM11 = 255-PWM3)
5. ADC channel 4 control PWM4 and PWM12. (PWM12 = 255-PWM4)
6. ADC channel 5 control PWM5 and PWM13. (PWM13 = 255-PWM5)
7. ADC channel 6 control PWM6
8. ADC channel 7 control PWM7

Example for this mode:

Without using microcontroller (MCU), user may use analog input device to control PWM output. For example, potentiometer can be connected to ADC channel 7, AN7 to give analog value to Multi PWM chip and control the speed of DC brush motor with the PWM output from PWM7.

b) Mode 2 : UART 9600

Mode 3 : UART 115200

Mode 2 and mode 3 is UART mode. The different between mode 2 and mode 3 is their baudrate. User may choose either baudrate 9600 or baudrate 115200 with mode selection pins. To select the mode with baudrate 9600, S2 and S1 pin are set to low (provide the pins with 0V or digital value '0') and S0 pin is set to high (provide the pins with 5V or digital value '1'). To select mode with baudrate 115200, S2 and S0 pin are set to low (provide the pins with 0V or digital value '0') and S1 pin is set to high (provide the pins with 5V or digital value '1').

No	Mode Select Pin	Digital Value / Voltage	
		Baudrate 9600	Baudrate 115200
1	S2	0 / 0V	0 / 0V
2	S1	0 / 0V	1 / 5V
3	S0	1 / 5V	0 / 0V

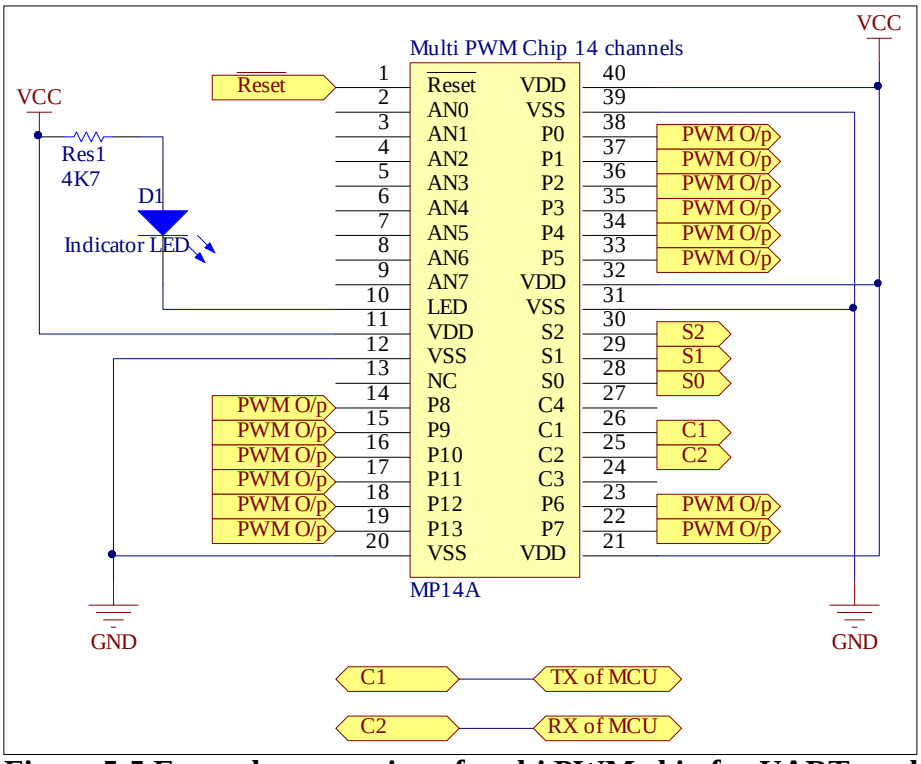


Figure 5-5 Example connection of multi PWM chip for UART mode.

Figure 5-5 above is example connection of multi PWM chip for UART mode. S2, S1 and S0 pin is connect to I/O pins of MCU. C1, which is UART Receive (RX) pin of multi PWM chip is connect to UART Transmit (TX) pin of MCU. C2, which is UART Transmit (TX) pin of multi PWM chip is connect to UART Receive (RX) pin of MCU.

```
//=================================================
// define input or ouput of board here
//=================================================

#define button1          PORTBbits.RB0
#define select1          LATBbits.LATB3
#define select2          LATBbits.LATB2
#define select3          LATBbits.LATB1
#define spi_cs_iic       LATBbits.LATB4
#define reset            LATBbits.LATB7
```

Example to define S2, S1, S0 pin which connected to a PIC microcontroller, PIC18F4520.

Protocols for data send to PWM chip

To control the PWM output of Multi PWM chip, microcontroller unit (MCU) need to send data through UART to PWM chip. To control 1 channel of PWM output on MP14A, 2 bytes of data need to send to PWM chip. The data are:

i. data1 = channel of PWM (0x00-0x0d)
ii. data2 = value of PWM (0-255 OR 0x00 – 0xFF)

Data1 send from MCU to MP14A is to determine the PWM output channel of Multi PWM chip that want to change the PWM output value. For example, to change PWM value of PWM output 11, 1 byte of data, '0x0b' needs send from MCU as data1 to PWM chip. Data2 is PWM value that want to set on PWM output channel that determined by data1. For example, to set PWM value on PWM output 11 to 128, 1 byte of data, '0x80' needs send from MCU as data2 to PWM chip.

To read back PWM value, there are also 2 bytes of data need to send to Multi PWM chip, which are:

i. data1 = 'r' or 'R'
ii. data2 = channel of PWM output that want to read(0x00-0x0d)

Data1 send from MCU to MP14A is to set Multi PWM chip to read mode. While data2 is the channel of PWM output that want to read. For example, to read PWM value of PWM output 11, 1 byte of data, 'r' or 'R' needs to send from MCU as data1 to PWM chip and follow by data2, '0x0b', which is the channel of PWM output that want to read. After that 1 byte of data, which is the PWM value of PWM output 11 will send from Multi PWM chip to MCU.

Below is example function to send and read UART data for MCU, PIC18F4520.

```
//send uart data
void send_serial(unsigned char data)
{
        while (PIR1bits.TXIF == 0);
        TXREG = data;
}
```

```
//read uart data
unsigned char read_serial(void)
{
        unsigned char temp;
        while(PIR1bits.RCIF==0);
        temp=RCREG;
        return temp;
}
```

Below is an example to send data to PWM chip using microcontroller PIC18F4520 and read the PWM value.

```
send_serial(0x01);
send_serial(0x7f);
send_serial(0x02);
send_serial(0x8f);
```

Send channel of PWM to activate PWM1.
Send PWM value. PWM value for PWM1 is 127.

```
send_serial('r');
send_serial(0x06);
data1=read_serial();
```

Send 'r' to read PWM value.
Send channel of PWM.
Read PWM value

c) Mode 4 : UART ASCII, 9600

Mode 5 : UART ASCII, 115200

Mode 4 and mode 5 are also UART mode. But these 2 modes are for UART communication between PWM chip and computer/laptop. Computer/laptop can send data to PWM chip to control PWM output channels. Same like previous UART, this UART communication available in 2 different baudrate, which are 9600 and 115200. User may choose either baudrate 9600 or baudrate 115200 with mode selection pin. To select mode with baudrate 9600, S2 pin is set to low (provide the pin with 0V or digital value '0') and S1 and S0 pin are set to high (provide the pins with 5V or digital value '1'). To select mode with baudrate 115200, S2 pin is set to high (provide the pin with 5V or digital value '1') and S1 and S0 pin are set to low (provide the pins with 0V or digital value '0').

No	Mode Select Pin	Digital Value / Voltage	
		Baudrate 9600	Baudrate 115200
1	S2	0 / 0V	1/ 5V
2	S1	1 / 5V	0 / 0V
3	S0	1 / 5V	0 / 0V

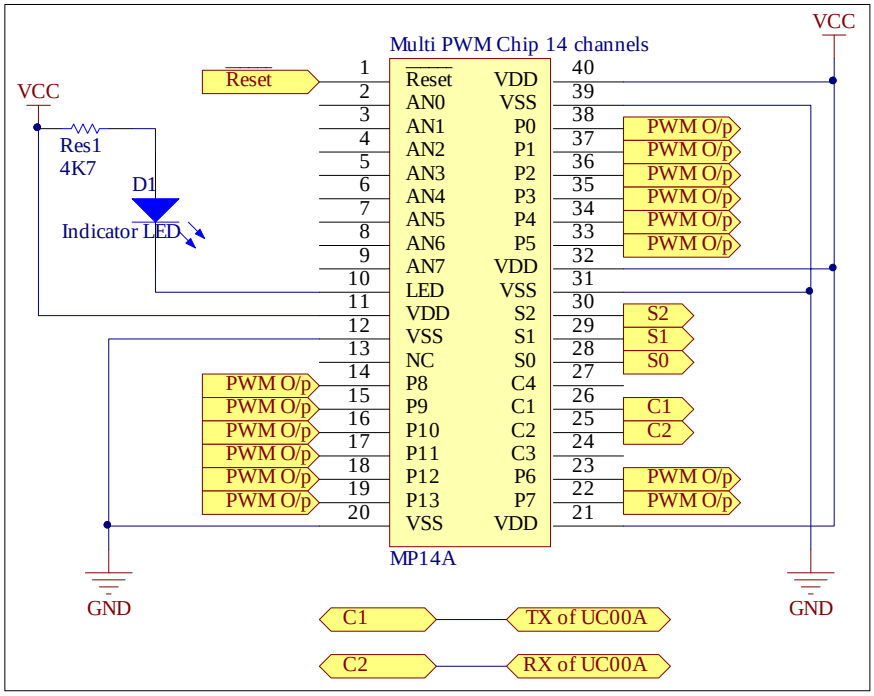


Figure 5-6 Example circuit connection to computer USB port through USB to UART converter,UC00A.

Figure 5-6 above is example connection of multi PWM chip. S2, S1 and S0 are connecting to switch for PWM chip mode selection. C1, which is UART Receive (RX) pin of multi PWM chip is connect to UART Transmit (TX) pin of UC00A. C2, which is UART Transmit (TX) pin of multi PWM chip is connect to UART Receive (RX) pin of UC00A. For more details of UC00A, please refer:
http://www.cytron.com.my/listProductCategory.asp?cid=82#3090

Protocols for data send to PWM chip

Computer/laptop can send ASCII data through to PWM chip to control PWM output channels. The ASCII data is data from keyboard on computer/laptop. There are 3 bytes of data need to send to Multi PWM chip:

i. data1 = channel of PWM (0-d in hexadecimal)
ii. data2 = value of PWM (0-f)
iii. data3 = value of PWM (0-f)

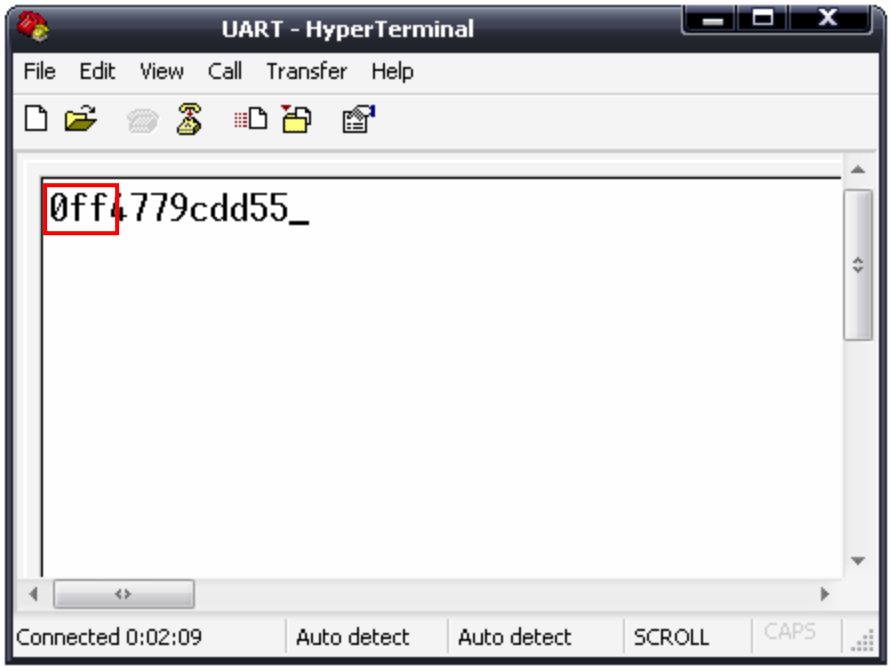
Data1 is to determine the PWM output channel of Multi PWM chip that want to change the PWM output value. For example, to change PWM value of PWM output 11, ASCII data, 'b' need to send to PWM chip through HyperTerminal. Data2 and data3 are PWM value that want to set on PWM output channel that determined by data1. Data2 is the upper nibble of the PWM value while data3 is the lower nibble of PWM value. The combination of data2 and data3 will form a bits data to set the PWM value of PWM output channel that determined by data1. For example:

i. data1 = b, data2 = f, data3 = f, output value of PWM channel 11 = 255 (0xff).
ii. data1 = 6, data2 = 2, data3 = f, output value of PWM channel 6 = 47(0x2f).

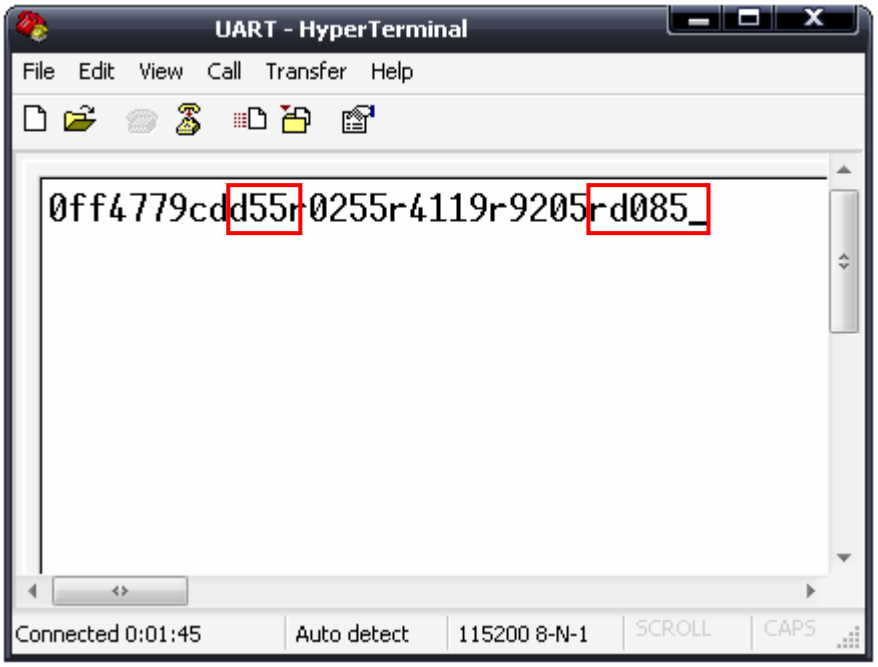
To read back PWM value, there are 2 bytes of data need to send to Multi PWM chip, which are:

i. data1 = 'r' or 'R'
ii. data2 = channel of PWM output that want to read(0x00-0x0d)

Send command 'r' or 'R' and followed with PWM channel to PWM chip to read value of PWM channels. If user give wrong data, '**.**' will feedback the error. Below are examples to send data from computer/laptop to PWM chip, all ASCII data in examples below are data from keyboard.



'0ff' are data send to PWM chip to set PWM output0 to 255. '0' is data1 which is representing of PWM output0 and 'f' 'f' is value of PWM for PWM0 channel. '477', '9cd', 'd55' are others data send to PWM chip to set PWM value on PWM output4, PWM output9 and PWM output13.



From figure above, PWM output 13 ('d' in hexadecimal) is first set to 0x55, which is same to 85 in decimal. After that, command 'r' is send to read value of PWM for PWM output 13 ('d' in hexadecimal). In this example, PWM value for PWM output 13 is display in decimal, which is 85 that same with 55 in hexadecimal.

d) Mode 6 : IIC (C4 as Address selection pin)

Another mode in PWM chip is Inter-Integrated Circuit IC (IIC) mode. IIC is using in PWM chip for communication between PWM chip and microcontroller unit (MCU). To choose IIC mode S2 and S0 are set to high (provide the pins with 5V or digital value '1') and S1 is set to low (provide the pins with 0V or digital value '0').

No	Mode Select Pin	Digital Value / Voltage
1	S2	1 / 5V
2	S1	0 / 0V
3	S0	1 / 5V

Each IIC compatible hardware slave comes with a predefined device address. Each slave is responsible for monitoring and responding only to its own address. Below are salve addresses for IIC communication in PWM chip.

No	Address Select Pin	Digital Value / Voltage	Slave Address
1	SS/ADD	0 / 0V	0x10
2	SS/ADD	1 / 5V	0x20

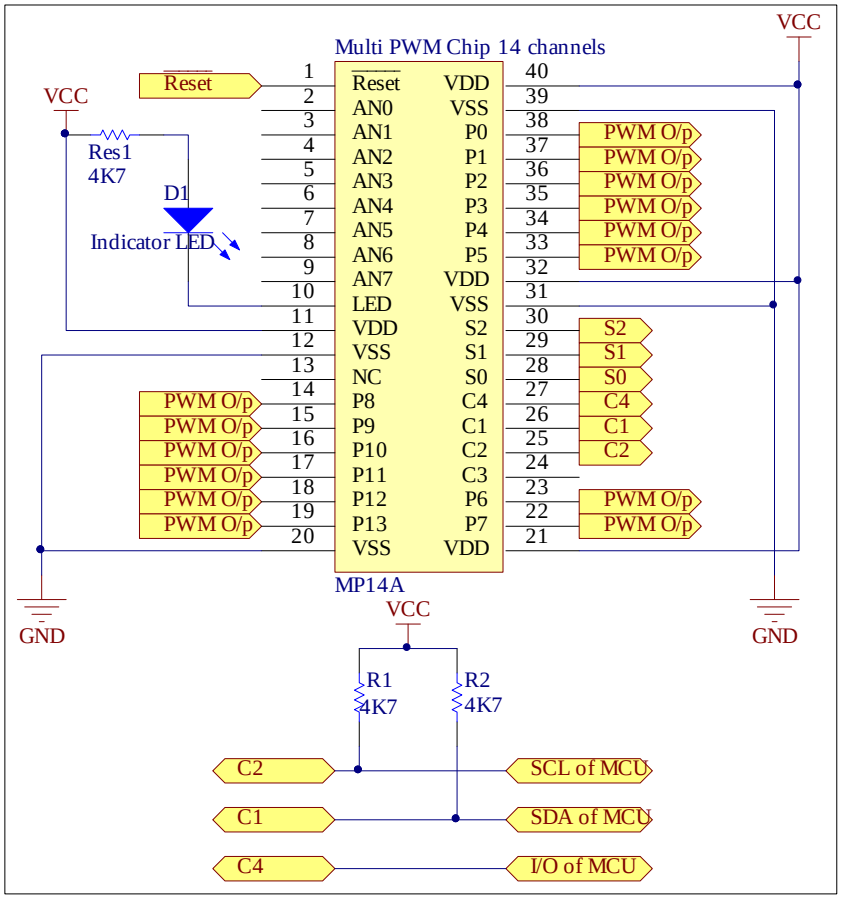


Figure 5–7 Example circuit connection to microcontroller (MCU) in IIC mode.

Figure 5-7 above is example circuit connection to microcontroller (MCU) in IIC mode. S2, S1, and S0 pin are connecting to I/O pins of MCU for mode selection. For communication between master and slave, C1, which is SDA pin of PWM chip is connect to SDA pin of MCU. C2, which is SCL pin of PWM chip is connect to SCL pin of MCU. C4 is connecting to I/O pin of MCU to set slave address. User also can set the slave address of MP14A by pull high the address pin C4 to 5V, it will set the address of MP14A to 0x20; while pull low the address pin C4, it will set the address of MP14A to 0x10.

Protocol for data send to PWM chip

Microcontroller (MCU) as a master in IIC mode, need to send data to MP14A to control the PWM output channels. To control 1 channel of PWM output on MP14A, 2 bytes of data need to send to PWM chip with correct IIC address (0X10 or 0X20). The data are:

i. data1 = channel of PWM (0x00-0x0d)
ii. data2 = value of PWM (0 - 255 OR 0x00 - 0xFF)

Data1 send from MCU to MP14A is to determine the PWM output channel of Multi PWM chip that want to change the PWM output value. 14 channels of PWM are represented in hexadecimal number (0x00 – 0x0d). For example, to change PWM value of PWM output 11, 1 byte of data, '0x0b' needs send from MCU as data1 to PWM chip. Data2 is PWM value that want to set on PWM output channel that determined by data1. For example, to set PWM value on PWM output 11 to 128, 1 byte of data, '0x80' needs send from MCU as data2 to PWM chip.

To read back PWM value, there are also 2 bytes of data need to send to Multi PWM chip, which are:

i. data1 = 'r' or 'R'
ii. data2 = channel of PWM output that want to read(0x00-0x0d)

Example of IIC function for Master write data to Multi PWM chip, the MCU use as master is PIC18F4520.

```
//iic start and send add (write mode)
//=============================================================================
void PWMwrite_IIC(unsigned char add, unsigned char channel, unsigned char value)
{
        //send IIC address
        add=add<<1;
        add=add&0b11111110;             //write mode

        PIR1bits.SSPIF=0;               //clear interrupt flag bit
        SSPCON2bits.SEN=1;              //start
        while(PIR1bits.SSPIF==0);       //wait
        PIR1bits.SSPIF=0;               //clear interrupt flag bit
        SSPBUF=add;                     //send add
        while(PIR1bits.SSPIF==0);       //wait
        PIR1bits.SSPIF=0;               //clear interrupt flag bit

        //send PWM channel
        SSPBUF=channel;                 //send data
        while(PIR1bits.SSPIF==0);       //wait
        PIR1bits.SSPIF=0;               //clear interrupt flag bit

        //send PWM value
        SSPBUF=value;                   //send data
        while(PIR1bits.SSPIF==0);       //wait
        PIR1bits.SSPIF=0;               //clear interrupt flag bit

        //stop IIC
        SSPCON2bits.PEN=1;              //stop
        while(PIR1bits.SSPIF==0);       //wait
        PIR1bits.SSPIF=0;               //clear interrupt flag bit
}
```

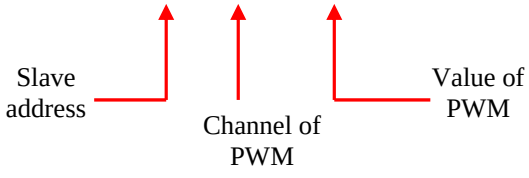
Example of IIC function for Master read data from Multi PWM chip, the MCU use as master is PIC18F4520.

```
unsigned char PWMRead_IIC(unsigned char add)
{
        unsigned char pwm_value;

        //send IIc address
        add=add<<1;
        add=add|0b00000001;             //read mode

        PIR1bits.SSPIF=0;               //clear interrupt flag bit
        SSPCON2bits.SEN=1;              //start
        while(PIR1bits.SSPIF==0);       //wait
        PIR1bits.SSPIF=0;               //clear interrupt flag bit

        SSPBUF=add;                     //send add
        while(PIR1bits.SSPIF==0);       //wait
        PIR1bits.SSPIF=0;               //clear interrupt flag bit

        //read IIC data
        SSPCON2bits.RCEN=1;             //enable receive mode
        while(PIR1bits.SSPIF==0);       //wait
        PIR1bits.SSPIF=0;               //clear interrupt flag bit
        pwm_value=SSPBUF;               //read data

        SSPCON2bits.ACKDT=1;            //send not ack
        SSPCON2bits.ACKEN=1;            //start sending not ack
        while(PIR1bits.SSPIF==0);       //wait
        PIR1bits.SSPIF=0;               //clear interrupt flag bit

        //stop IIC;
        SSPCON2bits.PEN=1;              //stop
        while(PIR1bits.SSPIF==0);       //wait
        PIR1bits.SSPIF=0;               //clear interrupt flag bit

        //return PWM vlaue
        return pwm_value;

}
```

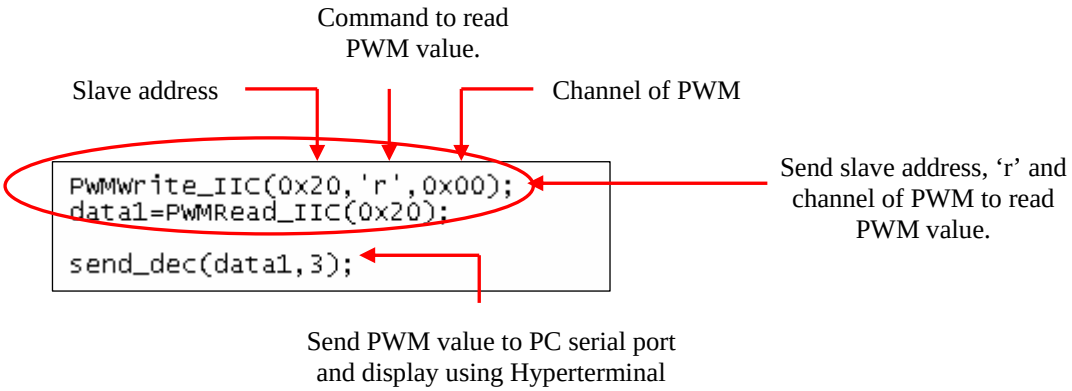
Below is an example to send data to PWM chip using microcontroller PIC18F4520 and read back the PWM value.

a. Example to write data to Multi PWM chip:

```
PWMWrite_IIC(0x20,0x00,0xf5);
PWMWrite_IIC(0x20,0x01,0xe5);
PWMWrite_IIC(0x20,0x02,0xd5);
PWMWrite_IIC(0x20,0x03,0xff);
```



b. Example to read PWM value from Multi PWM chip



e) Mode 7 : SPI without SS

Mode 8 : SPI with SS (SS/ADD as SS pin, active low)

SPI is another mode in PWM chip. They are 2 types of SPI mode. 1st is SPI without SS enable and 2nd is SPI with SS enable. SS is slave select input for SPI. To choose SPI without SS enable, S2 and S1 are set to high (provide the pins with 5V or digital value '1') and S0 is set to low (provide the pins with 0V or digital value '0'). To choose SPI with SS, all selection pin (S2S1S0) are set to high (provide the pins with 5V or digital value '1').

No	Mode Select Pin	Digital Value / Voltage	
		Without SS (mode 7)	With SS (mode 8)
1	S2	1 / 5V	1 / 5V
2	S1	1 / 5V	1 / 5V
3	S0	0 / 0V	1 / 5V

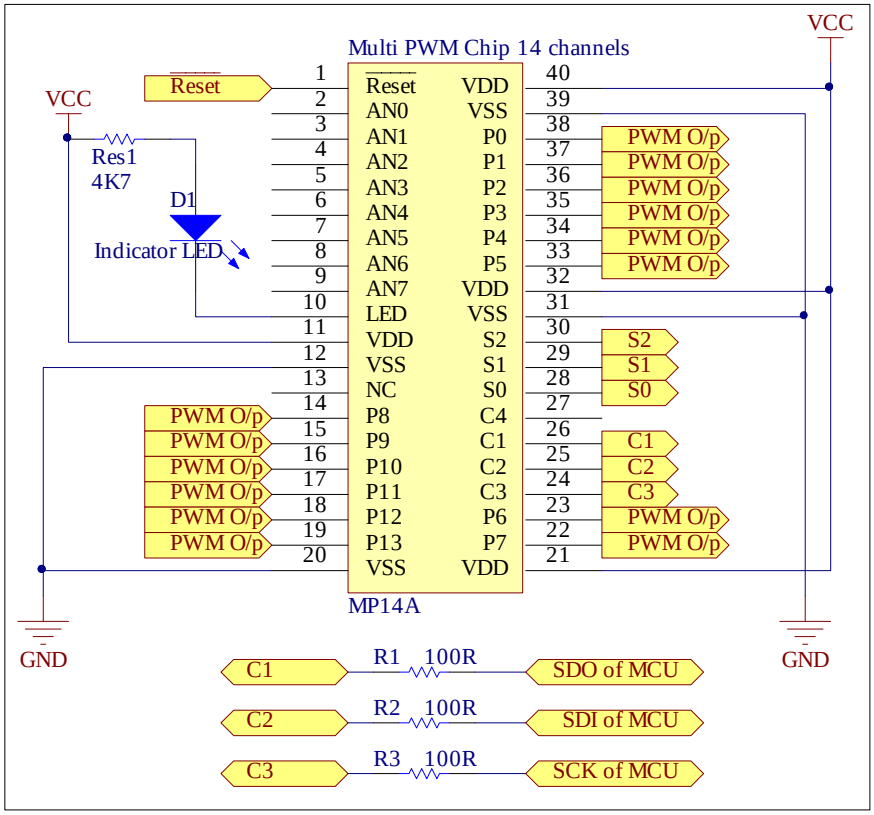


Figure 5-8 Example circuit connection to microcontroller (MCU) in SPI mode without SS enable

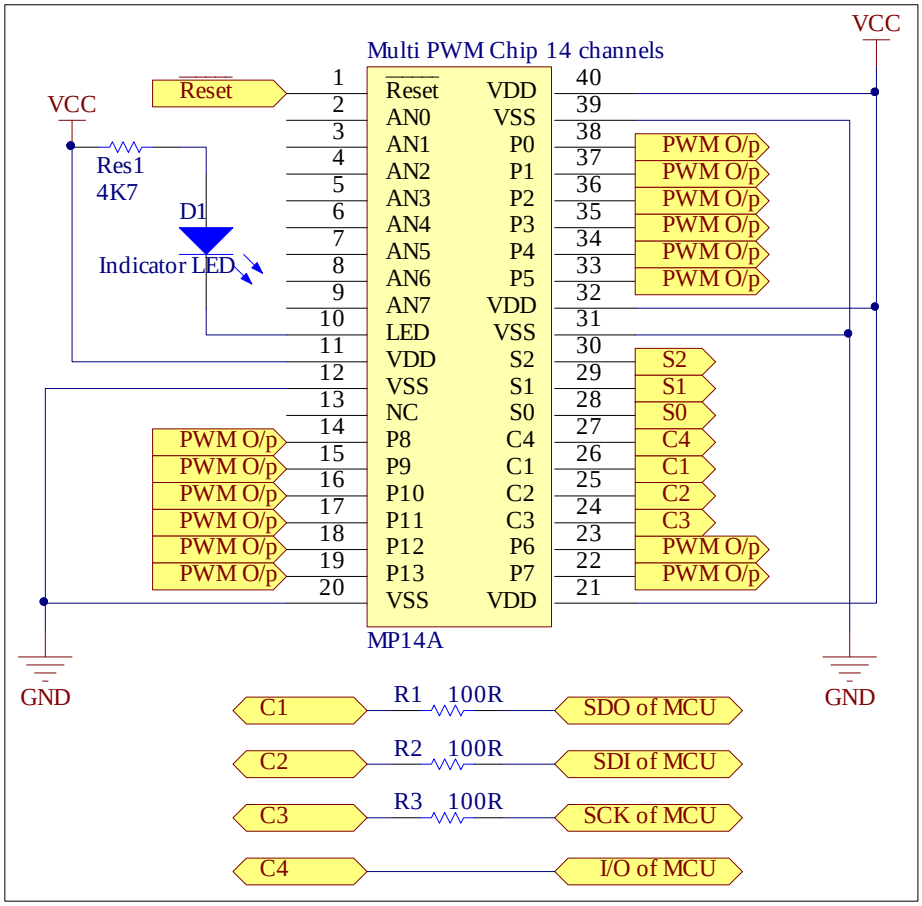


Figure 5-9 Example circuit connection to microcontroller (MCU) in SPI mode with SS enable

Figure 5-8 is example circuit connection to microcontroller (MCU) in SPI mode without SS enable. Figure 5-9 is example circuit connection to microcontroller (MCU) in SPI mode with SS enable. S2S1S0 pins are connecting to I/O pins of MCU for mode selection. For both modes, C1, which is SDI pin of PWM chip is connect to SDO pin of MCU. C2, which is SDO pin of PWM chip is connect to SDI pin of MCU. C3, which is SCK pin of PWM chip is connect to SCK pin of MCU. For SPI mode with SS enable (figure 5-9), C4, which is SS pin for SPI is connect to I/O pin of MCU. This SS pin is active low. In SPI SS enable mode, user need to enable the SPI communication in Multi PWM chip by provide the SS pin with 0V or digital value '0' before send data to Multi PWM chip.

Protocol for data send to PWM chip

For both SPI modes, (with SS enable and without SS enable), there are 2 bytes of data need to send from microcontroller unit (MCU) to PWM chip. The data are:

i. data1 = channel of PWM (0x00-0x0d)
ii. data2 = value of PWM (0 - 255 OR 0x00 - 0xFF)

Data1 send from MCU to MP14A is to determine the PWM output channel of Multi PWM chip that want to change the PWM output value. 14 channels of PWM are represented in hexadecimal number (0x00 – 0x0d). For example, to change PWM value of PWM output 11, 1 byte of data, '0x0b' needs send from MCU as data1 to PWM chip. Data2 is PWM value that want to set on PWM output channel that determined by data1. For example, to set PWM value on PWM output 11 to 128, 1 byte of data, '0x80' needs send from MCU as data2 to PWM chip.

To read back PWM value, there are also 2 bytes of data need to send to Multi PWM chip, which are:

i. data1 = 'r' or 'R'
ii. data2 = channel of PWM output that want to read(0x00-0x0d)

Below is example function to send SPI data for MCU, PIC18F4520.

```
void spi_send(unsigned char data)
{
        unsigned char buffer;
        // Send the data
        SSPBUF = data;

        // Wait for the SPI module to finish sending / receiving.
        while(SSPSTATbits.BF == 0);

        // Read the received data
        buffer = SSPBUF;
}
```

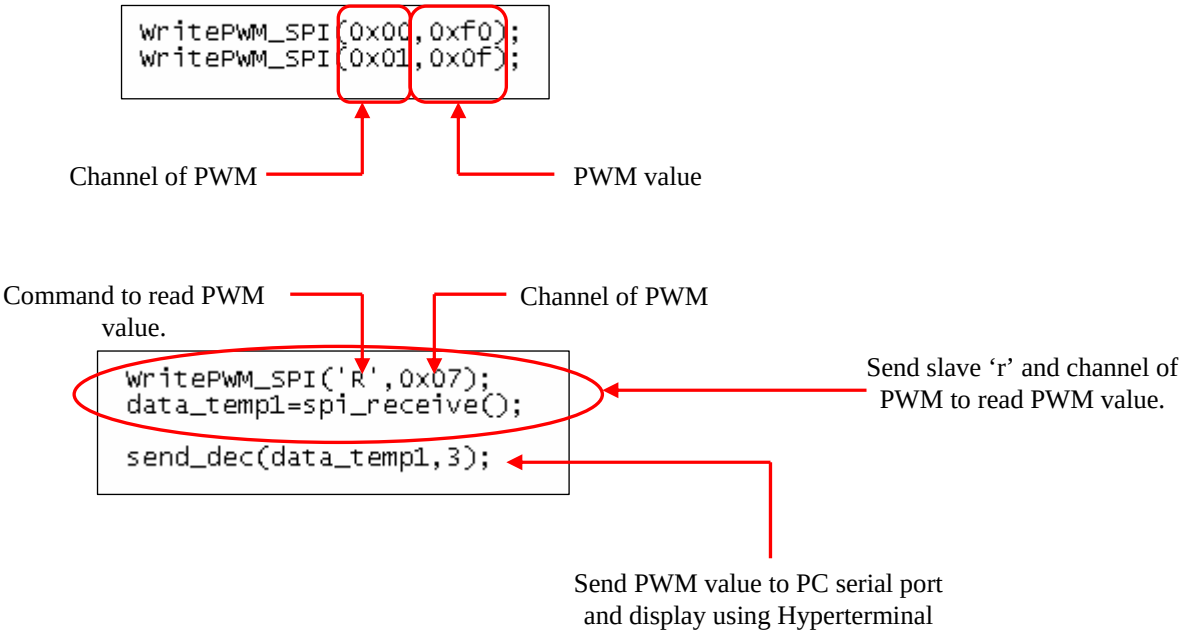
Below is example function to receive SPI data for MCU, PIC18F4520.

```
unsigned char spi_receive(void)
{
        // Send the dummy data
        SSPBUF = 0xff;

        // Wait for the SPI module to finish sending / receiving.
        while(SSPSTATbits.BF == 0);

        // Return the received data.
        return SSPBUF;
}
```

Below is example function to send and read data to PWM chip using microcontroller PIC18F4520.

```
void WritePWM_SPI(unsigned char channel, unsigned char value)
{
        spi_send(channel);
        spi_send(value);
}
```

Below is sample program to send and read back the PWM value for SPI mode.



Note1: For SPI communication, microcontroller which use as master might generate noise in SPI initialization, user is advised to reset PWM chip and set the PWM chip to SPI mode AFTER SPI initialization in master

Prepared by
Cytron Technologies Sdn. Bhd.
19, Jalan Kebudayaan 1A,
Taman Universiti,
81300 Skudai,
Johor, Malaysia.

Tel: +607-521 3178
Fax: +607-521 1861

URL: www.cytron.com.my
Email: support@cytron.com.my
sales@cytron.com.my

