PIC-Based Multi-Channel PWM Signal Generation Method and Application to Motion Control of Six Feet Robot Toy

Chin-Pao Hung, Wei-Ging Liu, Hong-Jhe Su, Jia-Wei Chen, and Bo-Ming Chiu

***Abstract*—**The aim of this research considered in this paper is to show a novel multi-channel PWM (pulse width modulation) signal generation method for the multi-joint RC robot driving. Integrating the I/O pins of the microcontroller and the interrupt function of the built-in timer, the maximum PWM channel number is identical to the number of I/O pins and can be used to drive the multi-joint robot. Differ to traditional polling scheme; the multiple channel PWM signals are synchronous. Applying this novel scheme to the 18 joints RC robot control, without any other extra chips or components, the smoothing motion control demonstrated the feasibility of the proposed scheme. Also, a user friendly interface is developed to benefit the motion control planning. User planned the walking path and downloaded it from PC to PIC microcontroller via RS232 protocols. Then the PIC microcontroller runs the motion control independently.

***Keywords*—**microcontroller, PWM, PIC, robot toy, RC motor, servo control.

I. Introduction

RECENTLY, robotics becomes a charming research topic, including robotic arms [1]-[2], mobile robot for special function [3], humanoid robot [4-5] and robot toys[6]-[8]. Specially, the robots toys attracted many people wallowed in its design and control, including us. Servo motors are the mainly drive components of the multi-joint robot toys. Generally, the mechanism is driven by more than ten RC motors. To drive these motors moving or rotating smoothly, the key technology is to generate the driving signals simultaneously. However, this is not an easy task and generally the engineers used the polling scheme to generate the driving signals, as shown in Fig. 1. If the maximum duty cycle of the driving signal is 2ms of each channel, the total polling time of n channels is smaller than 2×n mini-seconds and people maybe didn't feel the time delay evidently. But the time delay will affect the motion smoothly and make the complex action hard to be realized. Moreover, the polling period occupied the CPU time and made the program lack of efficiency.

To solve or alleviate the time delay problem, the designer usually employed an extra driver IC or circuit to achieve the desire performance, such as CPLD/FPGA [9], and PWM IC [10]. Naturally, it increases the control cost and the circuit becomes larger and complex. In [9], the authors used a CPU and FPGA chip to obtain the biped robot control. The PWM signals are generated by FPGA. Of course, it has larger PCB size and applied to 55cm tall robot. For smaller size robot, however, it needs smaller control board to benefit it hid in robot mechanism. Although most of new microcontrollers have built in PWM modules to generate the PWM signal [11]-[12]. Generally, the PMW channel is less than ten channels and the PWM frequency is beyond the desired bandwidth of RC motor. Therefore, the build-in PWM module can not be applied to the control of multi-joint RC robot directly.

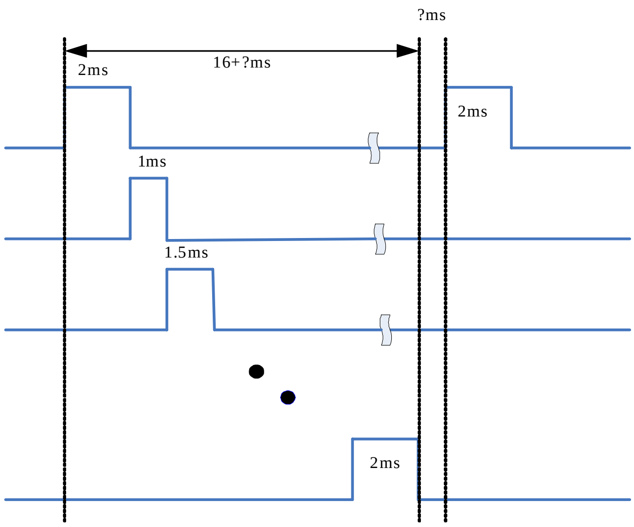


Fig. 1 Multiple PWM signals generation-polling scheme.

Manuscript received March 6, 2009: Revised version received April 4, 2009. This work was supported in part by the National Science Council, Taiwan, ROC under Grant NSC-97-2622-E-167-009-CC3, and Research Group of National Chin-Yi University of Technology.

C. P. Hung is now with the Department of Electrical Engineering of National Chin-Yi University of Technology, Taiping City, Taichung County, Taiwan, ROC. Phone: 886-4-23923881 fax: 886-4-23923881; E-mail: cbhong@ncut.edu.tw

W.G. Liu is with the Department of Electrical Engineering of National Chin-Yi University of Technology, Taiping City, Taichung County, Taiwan, ROC. E-mail: n0915n@yahoo.com.tw

H.J. Su is with the Department of Electrical Engineering of National Chin-Yi University of Technology, Taiping City, Taichung County, Taiwan, ROC. E-mail: whitedog175@yahoo.com.tw

J.W. Chen is with the Department of Electrical Engineering of National Chin-Yi University of Technology, Taiping City, Taichung County, Taiwan, ROC. E-mail: qoo_wei01@yahoo.com.tw

B.M. Chiu is with the Department of Electrical Engineering of National Chin-Yi University of Technology, Taiping City, Taichung County, Taiwan, ROC. E-mail:bread_416@yahoo.com.tw

To solve the multi-channel PWM signal generation problem as described above, a novel multi-channel digital PWM signal generation scheme is proposed in this paper. This novel scheme incorporates features that accommodate microcontroller based robot control with a variety of advantages such as without the necessary of extra circuit/IC, maximum PWM channels are identical to the IO pins, and the minimal control board size is easy obtained.

This paper mainly emphasizes the synchronized multiple channels PWM generation method and applied it to six feet RC robot control. To implement precise motion control needs more efforts on the integration of sensors and control theorem. Here, we just provided a friendly interface to plan the robot motion and the RC robot runs in open loop type control only.

II. Synchronized Multi-Channel PWM Generation Scheme

A. Synchronized multi-channel PWM signal

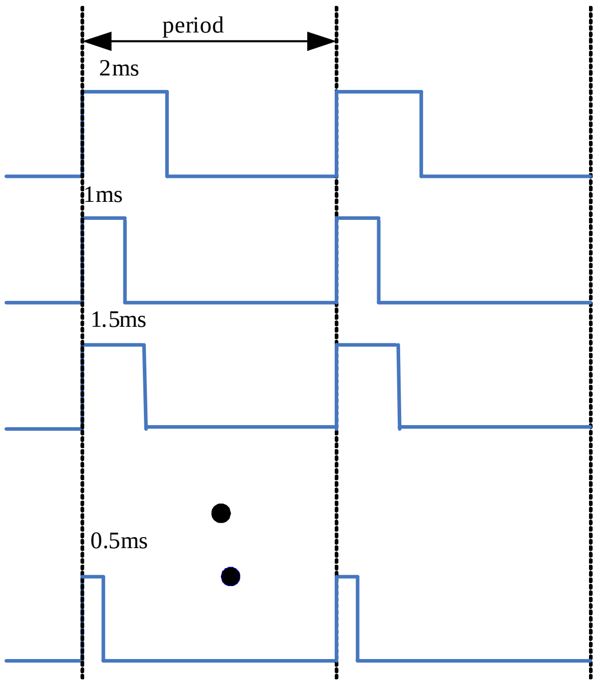


Fig. 2 Desired synchronized multi-channel PWM signals

Compared to the polling scheme of Fig. 1, the desired synchronized multi-channel PWM signals are shown as Fig. 2. It is especially suitable applied to the RC robot control and can alleviate the time delay problem evidently. Differ to the polling scheme of Fig. 1, all channels generate the pre-designed duty cycle simultaneously.

B. Digital multi-channel PWM signal generation

To generate the desired PWM signal of Fig. 2, a PIC-microcontroller based multi-channel PWM generation scheme is proposed in this paper. Omitting the oscillator circuit, power and the connectors, as shown in Fig. 3, only a microcontroller with the built-in timer interrupt function, the microcontroller can produce multi-channel PWM signals and

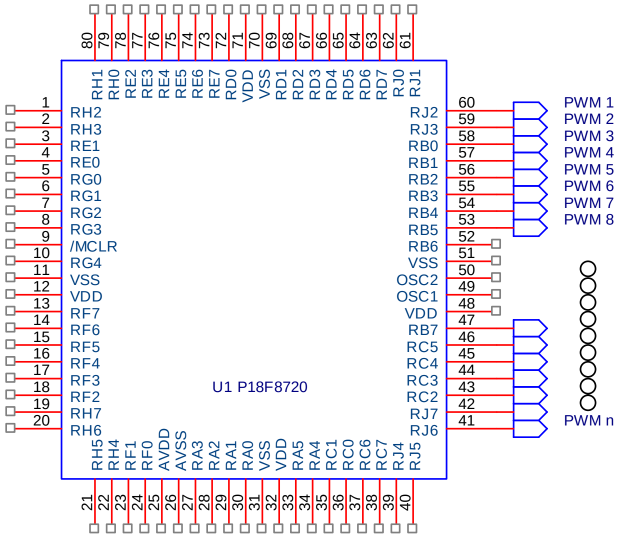


Fig. 3 Schematic diagram of microcontroller based PWM signal generation scheme

the maximum channels are identical to the number of IO pins. The design processes are illustrated as follows.

Firstly, configure the timer to generate a preset periodic clock as the interrupt time base and preset the PWM I/O pins at high level. Taking the 16 bits timer of PIC18F8720 microcontroller as example [11], we set the timer register with an initial value of 65536-***count***. The register is incremented for every machine cycle (4 oscillator cycles for PIC18F8720) and every ***count*** clock the TF flag will generate an interrupt request to CPU. Since a machine cycle consists of 4 oscillator periods, the count rate is 1/4 of oscillator frequency. Assuming the time base is 0.125ms and the oscillator frequency is 20MHz, the initial value of the timer register is set as 65536-OSC_freq/4*0.000125 and the interrupt frequency is 8KHz, as shown in Fig.4.

Secondly, design a corresponding interrupt subroutine to control the duty cycle of each PWM channel. In the interrupt subroutine, an interrupt counter is incremented once for every interrupt happened and can be used for the be

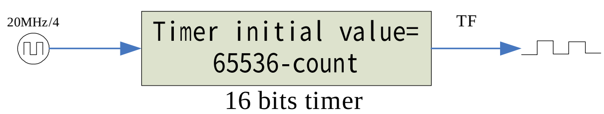


Fig. 4 Timer operation diagram

duration calculation of each pin's high level output. Assuming the duty cycle of each PWM channel is calculated in the beginning of every interrupt cycle. Comparing the preset duty cycle of each channel and the time of duration, the corresponding pin will be reset as zero if the equal condition is satisfied. Therefore, the duty cycle of each channel is under control. Also, based on the precise time base, a preset interrupt counter bound is used to tune the PWM period. Once the upper bound of interrupt counter is achieved, the interrupt counter will be reset as zero and all the pins' outputs are reset at high level.

Program 1 is the partial code of interrupt subroutine to illustrate above operation. In program 1, interrupt_count ×time_base denotes the duration time of each PWM cycle. Comparing the duration time with the preset duty cycle of each channel *set_PWM_duty[i]*, if the equal condition is satisfied, the corresponding IO pin will be reset as LOW. The final *if* instruction is used to tune the PWM period (frequency). Once the equal condition is satisfied, the interrupt_count will be reset as 0 and all the PWM IO pins reset as LOW. The PWM period is count_upper_bound × time_base. The detailed programming flowchart is shown as Fig. 5, including main program and interrupt subroutine. Under this operation, the main program runs the user program loop still and with the higher efficiency.

```
void interrupt_h_function(void)    //interrupt function
{
……….;
……….;
interrupt_count=interrupt_count+1;//interrupt_count
//incremented
if(interrupt_ count ==set_PWM_duty[0])//Is the duty cycle
//of channel 1 achieved?
PWM_1=0;//reset the corresponding IO pin 1 as /LOW
if(interrupt_ count ==set_PWM_duty[1]) //Is the duty cycle
//of channel 2 achieved?
PWM_2=0;//reset the corresponding IO pin 2 as //LOW
if(interrupt_ count ==set_PWM_duty[2]) //Is the duty cycle
//of channel 3 achieved?
PWM_3=0;//reset the corresponding IO pin 3 as LOW
   ……….
if(interrupt_count==set_PWM_duty[n]) //Is the duty cycle
//of channel n achieved?
PWM_n=0;//reset the corresponding IO pin 3 as LOW

if(interrupt_count==count_upper_bound)    //Does    the
//interrupt_ count upper bound achieved?
{  interrupt_count=0;//reset the interrupt_count value
   PORTD=0XFF; //reset the corresponding IO pins
   PORTJ=0XFF;
   PORTF=0XFF;
   PORTB=0XFF;
}
   ……………..;
   ……………..;
}
```

Program 1:Interrupt subroutine

C. PWM resolution

Most microcontrollers accommodate built-in PWM module with fixed resolution such as 8 bits or 10bits [12]. After the configuration, the resolution is fixed. But in the proposed scheme as described above, the resolution is variable and depends on the value of count_upper_bound. Setting the value as 1023, the duty values are between 0 and 1023, resolution is denoted as 1/1024, and is similar to 10 bits PWM resolution. Arbitrarily setting the count_ upper_bound will obtain different PWM resolution. Each channel can be set a different count_upper_bount to obtain different resolution. It is more flexible than traditional scheme.

D. Maximum PWM frequency evaluation

As described above, the PWM period is count_upper_bound ×time_base. However, the period is subject to the CPU

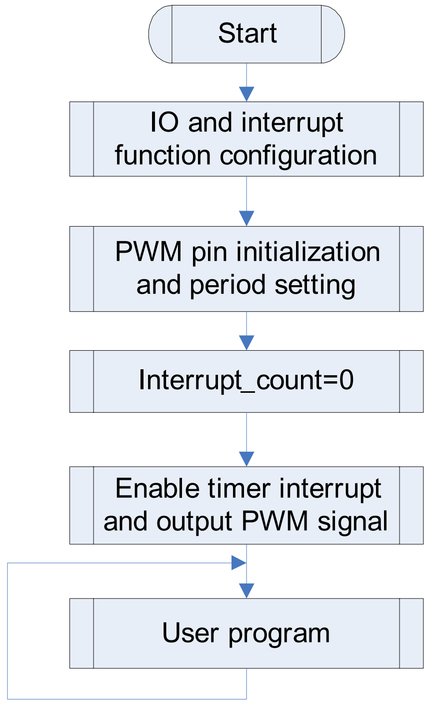


Fig. 5(a) Main program flowchart

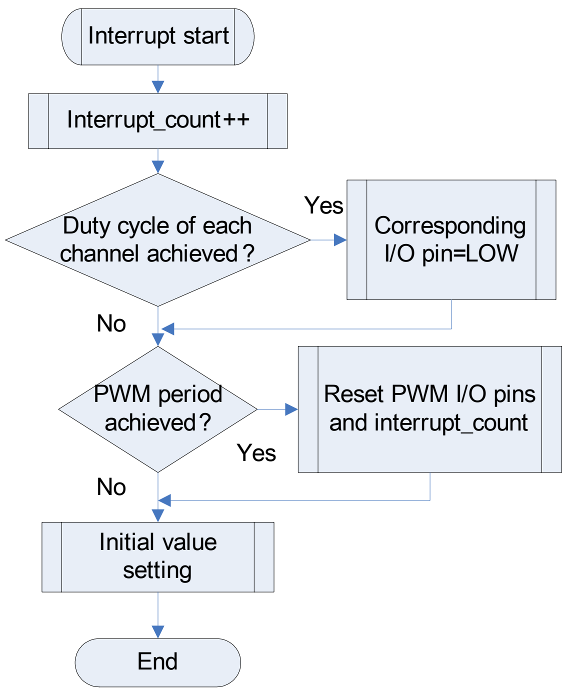


Fig. 5(b) Interrupt subroutine flowchart performance, the PWM channel number and the PWM resolution. In 18 PWM channels application, the executing time of Program 1 is less than 50 μ s using PIC18f8720 microcontroller under 20 MHz operation oscillator. Assuming the count_upper_bound and time_base is set as 1000 and 0.01ms respectively; the PWM frequency is nearly 100Hz. The PWM frequency is expressed in equation (1). The user can set the PWM frequency arbitrarily to satisfy the desired performance and the only limitation is the time_base must be larger than the executing time of Program 1.

$$\text{PWM freq.} = \frac{1}{count\_upper\_bound \times time\_base} \quad (1)$$

III. Multi-Channel PWM Signal Output

To implement the proposed PWM generation scheme, we designed an 18-channel PWM generation circuit to demonstrate the feasibility and advantages. The control kernel just like mentioned above is PIC18f8720 microcontroller. The system oscillator frequency is 20MHz (maximum operation frequency 40MHz). Some experimental results are shown and illustrated as follows.

A. 10 bits resolution and 0.01ms time base

10 bits means the count_upper_bound of Program 1 is 1023. Using equation (1) will obtain the PWM frequency is nearly 100 Hz. Figure 6 shows the PWM signals of channel 1~8 . All the channels output PWM signals synchronously.

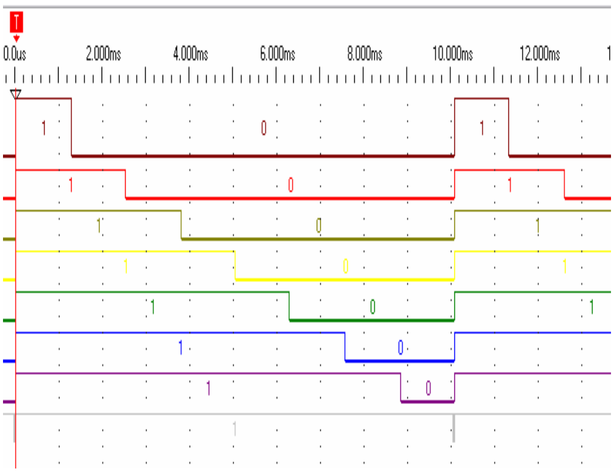


Fig. 6 18 channels 100Hz PWM signals output

B. Hybrid frequency/resolution PWM signals output s

Here hybrid means each channel can be set with different resolution or frequency. Figure 7 shows the proposed scheme with the ability of arbitrary setting the resolution and frequency. In the 18 channels, odd ones output nearly 100Hz and resolution 0~99, even ones output nearly 200Hz with 8 bits resolution. That is in the interrupt subroutine with two count_upper_bound values of 100 and 200. Noted the time base is the same as above 0.01ms. In this application, the resolution is more flexible than a tradition digital PWM scheme in which

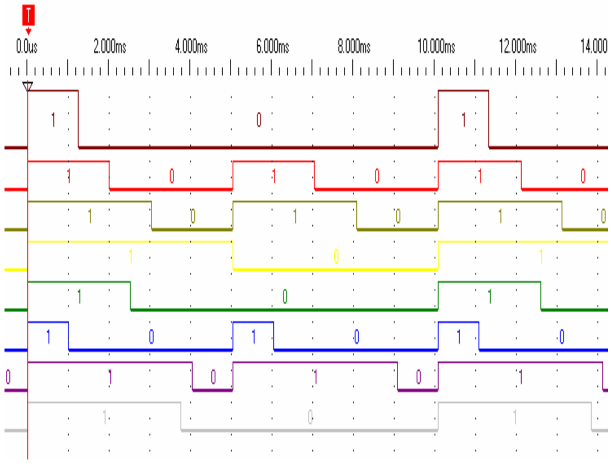


Fig. 7 Hybrid PWM signals

resolution is limited to 2 to the power of n.

In fact, hybrid PWM signals can be achieved easily by modifying Program 1 with multiple interrupt counters and upper bound values such as Program 2.

```
void interrupt_h_function(void)    //interrupt function
{ …
interrupt_count1++;//interrupt_count1 incremented
interrupt_count2++;//interrupt_count2 incremented
if(interrupt_ count1 ==set_PWM_duty[0])//Is the duty
//cycle of channel 1 achieved?
  PWM_1=0; //reset the corresponding IO pin 1 as LOW
if(interrupt_ count2 ==set_PWM_duty[1]) //Is the duty
//cycleof channel 2 achieved?
  PWM_2=0; //reset the corresponding IO pin 2 as LOW
if(interrupt_ count1 ==set_PWM_duty[2]) //Is the duty
//cycleof channel 3 achieved?
  PWM_3=0; //reset the corresponding IO pin 3 as LOW
  ………
………
if(interrupt_ count2 ==set_PWM_duty[n]) //Is the duty
//cycleof channel n achieved?
  PWM_n=0;//reset the corresponding IO pin 3 as LOW

if(interrupt_count1==count_upper_bound1 )//Is the
//interrupt_count1 upper bound achieved?
{  interrupt_count1=0;//reset the interrupt_count1 value
   PORTD=0XFF;  //reset the corresponding IO pins
   PORTJ=0XFF;
}
if(interrupt_count2==count_upper_bound2)  //Does the
//interrupt_count2 upper bound achieved?
{  interrupt_count2=0;//reset the interrupt_count2 value
   PORTF=0XFF;  //reset the corresponding IO pins
   PORTB=0XFF;
}
   ……………..;
   ……………..;
}
```

Program 2. Interrupt subroutine with hybrid PWM output

IV. Robot toys servo control application

A. Brief description of RC servo motor

Generally, robot toys consist of more than ten RC servo motors. The RC servo motor, as shown in Fig. 8, is driven by pulse width modulation signal. For example, the pulse width input from 0.5ms to 2.0ms will respond to the axis output 0 degrees to 180 degrees, as shown in Fig. 9. The designer controls the RC servo motors of robot toys to achieve the special motion indirectly. If the driving signals are shown as in Fig. 1, the delay phenomenon will be evident on the high joints robot motion. Here, we design a six feet 18 degrees robot to demonstrate the excellent performance of the proposed PWM generation scheme.

Fig. 8 RC motor photo

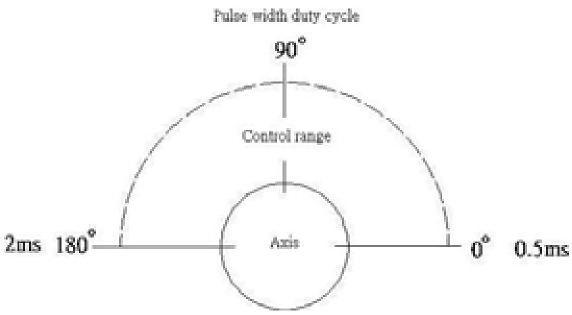


Fig. 9 Relationship of pulse width and axis output of RC motor

B. Application to six feet robot control

The six feet robot is shown in Figure 10. Each foot is driven by three RC servo motors and six feet with 18 degrees. To drive the robot motion smoothly and dexterously, it needs to be synchronized with multi-channel PWM signals. Using the proposed scheme, the block diagram of control PCB is shown as Fig. 11, it main contains the PIC18f8720 controller, programming connector, oscillator, reset button, and no other extra IC. The control PCB, with a small, clear, and simple view, is shown as Fig. 12. Its real size is 4cm*6cm only and the maximum PWM channels **n** is identical to the I/O pins, 64 channels. Here we just used 18 channels. The small size control board benefits it to hide in the robot toys.

To fit the RC motor control requirement, as shown in Fig.9, we designed the interrupt time base to be 0.125ms and the

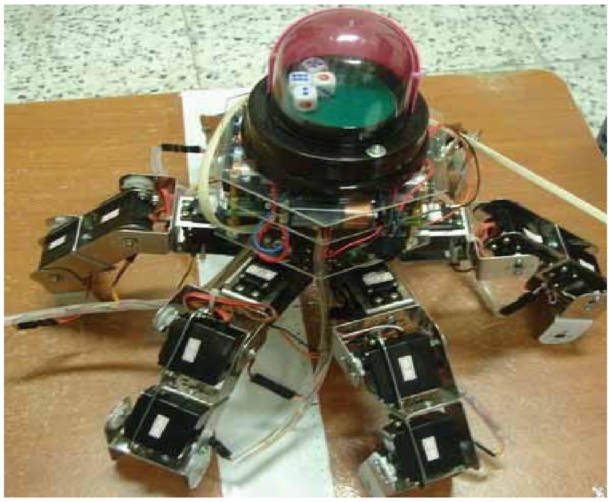


Fig. 10 Six feet robot toys

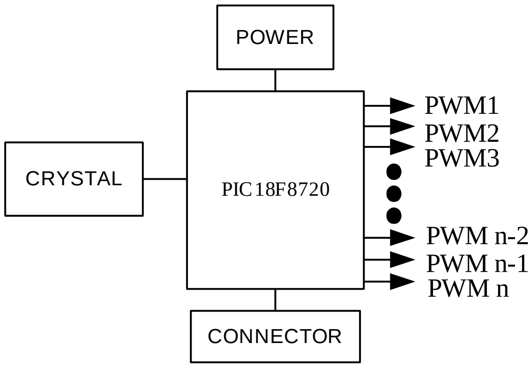


Fig. 11 Block diagram of control PCB

Fig. 12 photo of Control PCB

count_upper_bound 160, i.e. PWM frequency is nearly 50Hz. The output PWM signals of channel 1 to 8 are shown in Fig. 13. Here we just showed 8 channels of the 18 channels. It drives the RC robot with smooth motion and without the time delay feeling. The reader maybe doubted such resolution drives the RC motor of Fig. 8 only stop on 12 points. Of course, operating the PIC microcontroller on a higher oscillator frequency and take smaller interrupt time base can solve this question and achieve higher rotating resolution. Noted rotating resolution is different to the PWM resolution. It is subject to the bandwidth of RC motor command, such as 0.5ms to 2m pulse width. Using

0.125ms as time base, the RC motor rotating resolution is only 12.

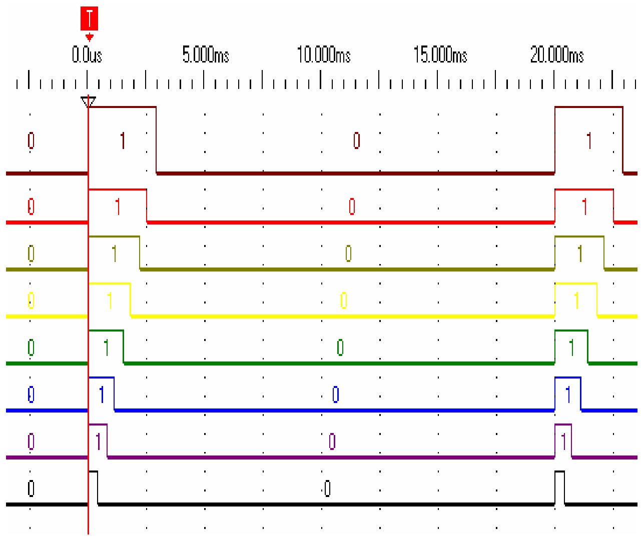


Fig. 13 18 channels 50Hz PWM signals output

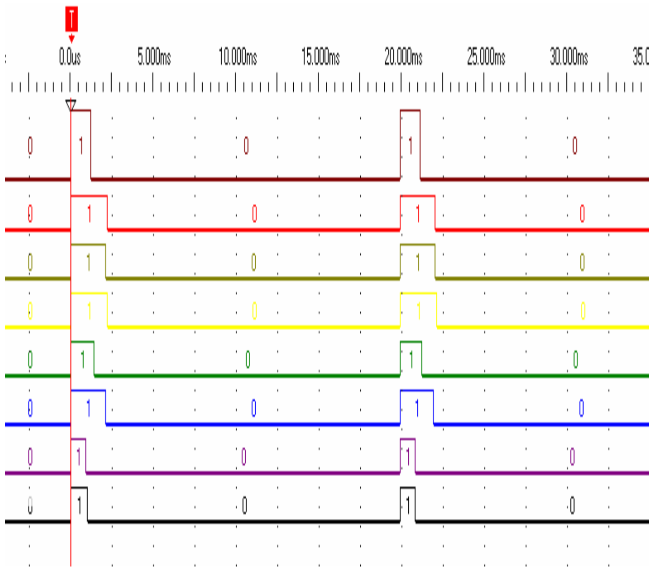


Fig. 14 50 Hz PWM output with 12.5 μs time base

C. Velocity control scheme

As described above, using 10 MHz crystal oscillator and turn on the quadruple HS+PLL mode. In this situation, we set the time base 12.5 μs and PWM period 20 *ms*, then the PWM resolution is nearly 1600 and the rotating resolution of RC motor is improved to 120. Figure 14 shows the 8 channels PWM signals with arbitrary duty cycle. Noted the RC motor in this application just receive 0.5ms to 2ms, i.e. the count value of duty cycle is 40 to 160.

Generally, the RC motor lacks of velocity control function. However, with the smaller time base, the velocity control of RC motor can be come true. In the PWM period, by software computation we can adjust the duty cycle output gradually to obtain the velocity control purpose. We will present this scheme in other paper.

V. Motion Control of RC Robot and Walking Path Planning Interface

A. Forward motion of six feet RC robot

Simulating the insect motion, Fig. 15 shows the schematic diagram of forward motion. Encoded the motor as Fig.16 and defined the home position is each joint closed to floor with zero degree. For forward motion, each joint's angle in degrees can be list as Table 1. It needs only seven steps to move the robot walking forward. More steps can obtain suppler and complex moving but needs much time to plan the joints moving timing. Such as Table 2, list the joint angle in degrees for fancy moving with 20 steps.

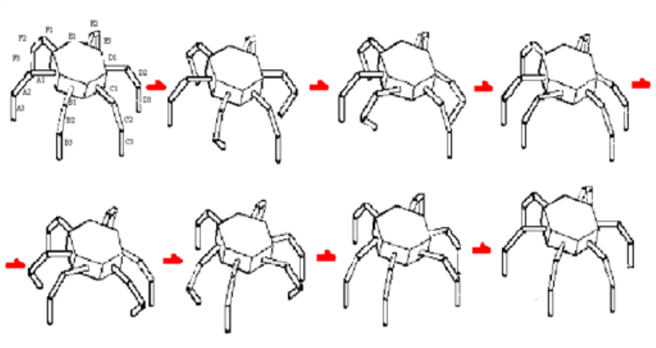


Fig. 15 Schematic diagram for forward

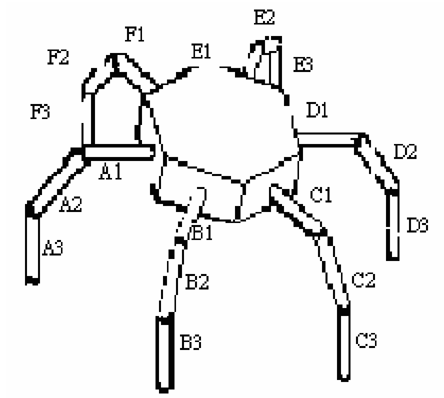


Fig. 16 Motor number of each joint

Table 1. Joint angles in degrees for forward motion

STEP	1	2	3	4	5	6	7
A1	0	0	0	0	0	-45	0
A2	0	0	0	0	0	0	0
A3	0	0	0	0	45	0	45
B1	0	0	45	0	0	0	0
B2	0	0	0	0	0	0	0
B3	0	-45	0	45	0	0	0
C1	0	0	0	0	0	-45	0
C2	0	0	0	0	0	0	0
C3	0	0	0	0	45	0	45
D1	0	0	45	0	0	0	0
D2	0	0	0	0	0	0	0
D3	0	-45	0	45	0	0	0
E1	0	0	0	0	0	-45	0

E2	0	0	0	0	0	0	0
E3	0	0	0	0	45	0	45
F1	0	0	45	0	0	0	0
F2	0	0	0	0	0	0	0
F3	0	-45	0	45	0	0	0

Table 2. Fancy motion(continue)

STEP	8	9	10	11	12	13	14
A1	0	0	0	0	0	45	0
A2	0	90	-90	0	90	0	-90
A3	0	-45	45	0	90	0	-90
B1	0	0	0	0	0	0	0
B2	0	90	-90	0	0	0	0
B3	0	-45	45	0	0	0	0
C1	0	0	0	0	0	45	0
C2	-90	90	-90	0	90	0	-90
C3	-90	-45	45	0	90	0	-90
D1	0	0	0	0	0	0	0
D2	0	90	-90	0	0	0	0
D3	0	-45	45	0	0	0	0
E1	0	0	0	0	0	45	0
E2	0	90	-90	0	90	0	-90
E3	0	-45	45	0	90	0	-90
F1	0	0	0	0	0	0	0
F2	-90	90	-90	0	0	0	0
F3	-90	-45	45	0	0	0	0

Table 2. Fancy motion(continue)

STEP	15	16	17	18	19	20
A1	-45	0	0	0	0	0
A2	0	0	0	0	0	0
A3	0	0	0	0	0	0
B1	0	0	-45	0	45	0
B2	0	90	0	-90	0	0
B3	0	90	0	-90	0	0
C1	-45	0	0	0	0	-45
C2	0	0	0	0	0	0
C3	0	0	0	0	0	0
D1	0	0	-45	0	45	0
D2	0	90	0	-90	0	0
D3	0	90	0	-90	0	0
E1	-45	0	0	0	0	0
E2	0	0	0	0	0	0
E3	0	0	0	0	0	0
F1	0	0	-45	0	45	0
F2	0	90	0	-90	0	0
F3	0	90	0	-90	0	0

B. *Experimental system for walking planning*

The developed experimental system for man-machine interface is shown as Fig. 17. Figure 18 is the schematic diagram of Fig. 17. It mainly contains a personal computer, RS232 communication port, control board (alternate type PCB) and six feet robot. The man-machine interface program is designed using Visual Basic language.

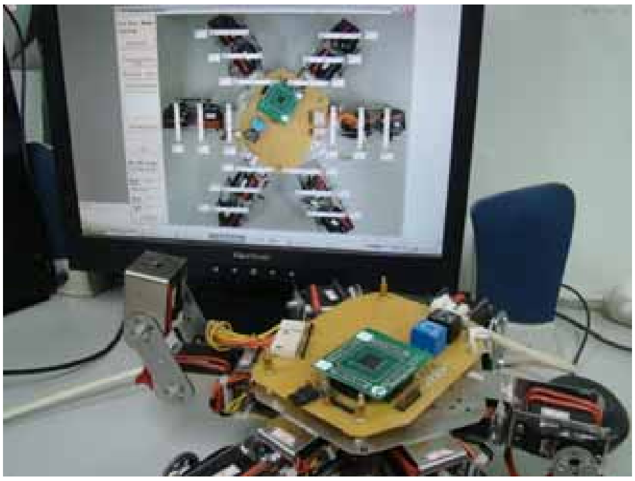


Fig. 17 Man machine interface development system

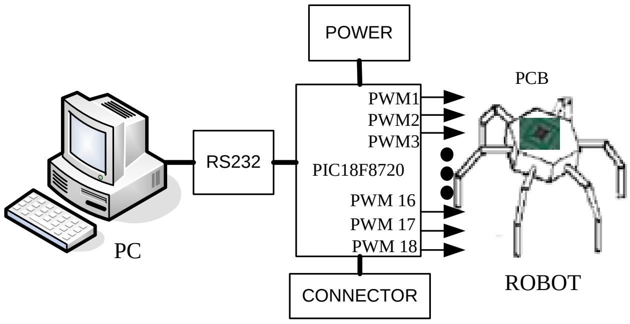


Fig.18 Schematic diagram of man-machine interface system

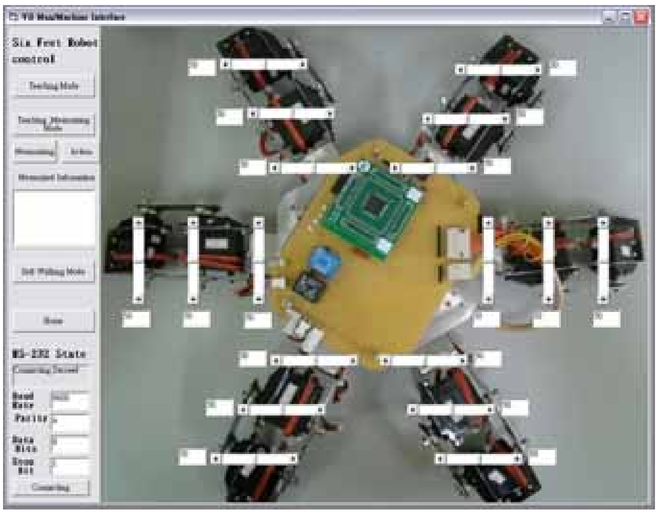
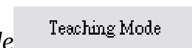


Fig. 19 Man machine interface windows

C. *Walking planning program design*

The man-machine interface is shown as Fig. 19. In Fig. 19, each motor mapped to a horizontal or vertical scroll bar. Moving the scroll bar the mapped RC motor will move relative degrees. The function of the developed interface is illustrated as follows.

- *Teaching mode* 

Press the Teaching Mode button, system will enter the teaching mode. The user can move the scroll bar and the relative position command of RC motor will transfer from PC to PIC via RS232 communication port. Observing the moving angle of relative motor, the user can record the joint angles in degrees for every moving step of each joint and list such as Table 1. Noted the scroll bar in the leftest denotes the rotating angle is 0%, i.e. 0 degree. And in the middle denotes the rotating angle is 50%, i.e. 90 degrees, and so on.

- *Teaching/Memorizing mode* Teaching_Memorizing Mode

Press the Teaching-Memorizing Mode, system turns off Teaching Mode and switched to memorizing mode. Press the Memorizing button, system will record the entire teaching angles of each joint. Press Action button will reappear the memorized motion angles. All the memorized progress will list in Memorized Information window.

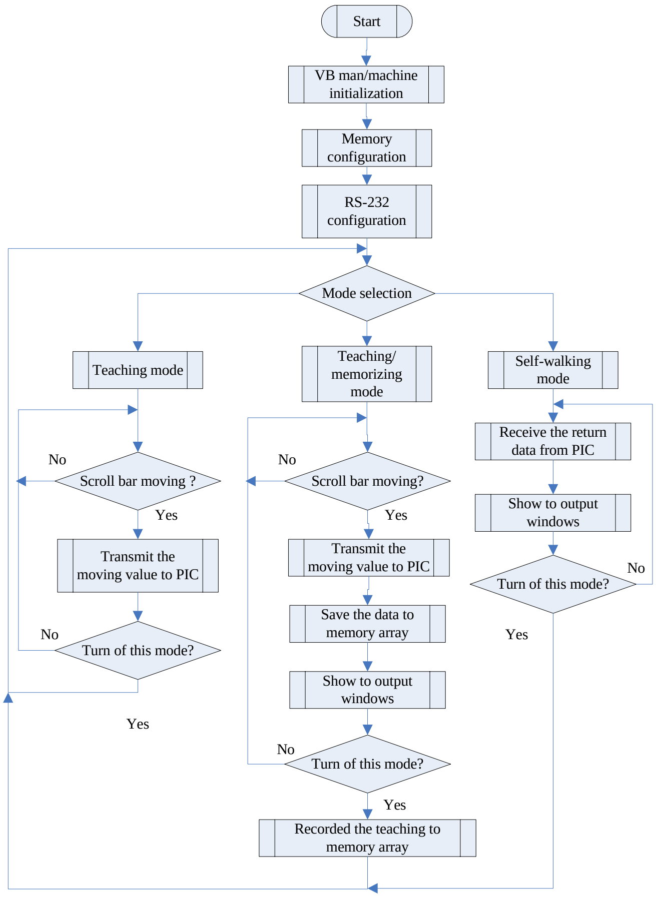


Fig.20. main program flowchart

- *Self-Walking mode* Self-Walking Mode *and Home mode* Home.

Press the Self-Walking Mode button will execute all the recorded motion and Home button return to the original state, each motor closed to floor.

D. PC program design

To achieve the function as described above, the PC end program was designed using Visual Basic. The program flowchart is shown as Fig. 20.

E. PIC program design

PIC received the command from PC, it mainly contains three parts; main program, RS-232 interrupt subroutine and multi-channel PWM signal output interrupt subroutine. The main program flowchart is shown as Fig. 21(a), RS-232 interrupt subroutine is shown as Fig. 21(b), and the PWM signal output interrupt subroutine is same as Fig. 5(b).

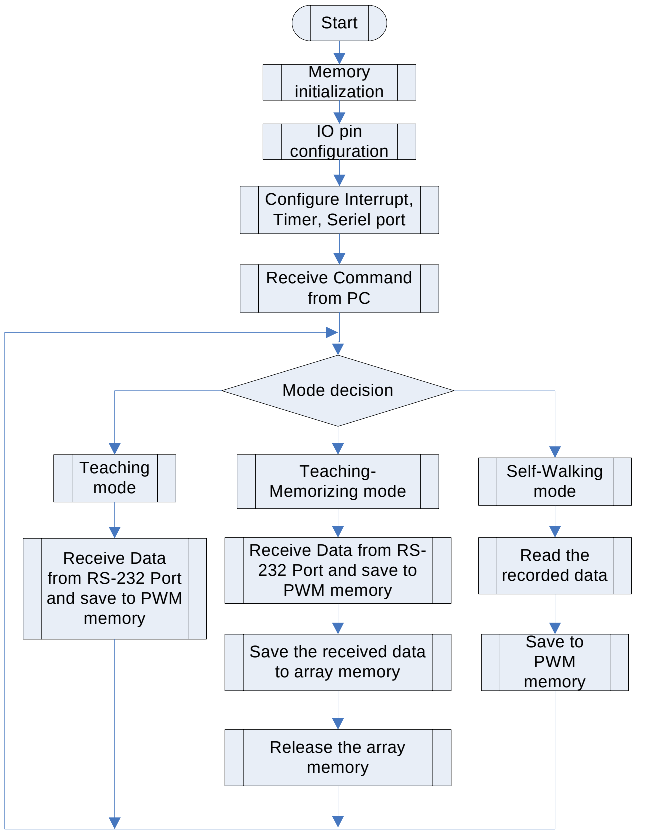


Fig. 21(a) Main program flowchart of PIC microcontroller

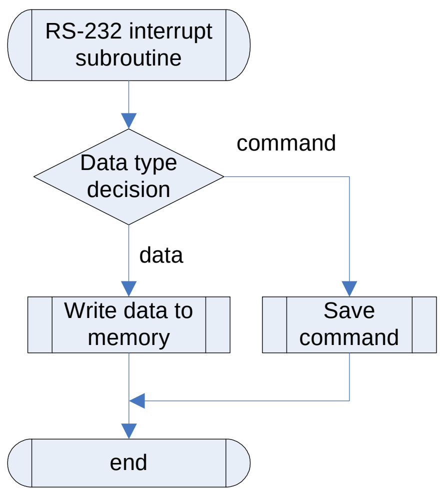


Fig. 21(b) RS-232 interrupt subroutine flowchart of PIC microcontroller

VI. Conclusion

A novel multi-channel PWM signal generation scheme is proposed in this paper. Using any microcontroller can generate synchronized multi-channel PWM signals and the max channel number is identical to the I/O pins' number. Since without any special IC added, the control circuit is easy to simplify and minimize immensely. Simultaneously, hybrid PWM signals are easily obtained just though software modification, and the cost is lower than traditional scheme. Also, a friendly man machine interface program is developed to benefit the motion control and planning in this paper. The proposed scheme is rigorously verified by experimental tests. Applied to RC robot control, this also demonstrated it outperforms a traditional polling method. Integrating more sensors to implement biologically inspired robot locomotion is our future work.

Acknowledgment

The authors would gratefully acknowledge the constructive suggestions and comments of the reviewers and the editors. And thanks the financial support from National Science Council, Taiwan, ROC, under grant NSC-97-2622-E-167-009-CC3 and the Academic Research Group of Chin-Yi University of Technology.

References

[1] K. Daeinabi and M. Teshnehlab, "Steam tracking of intelligent arc welding robot", *Proc. 6th WSEAS Int. Conf. System Theory & Scientific Computation,* 2006, pp. 161–166.

[2] M. C. Popescu, I. Borcosi, O. Olaru, L. Popescu, and F. Grofu, "Simulation hybrid fuzzy control of SCARA robot", *Proc. 3th WSEAS Int. Conf. Applied and Theoretical Mechanics,* 2007, pp. 175–180.

[3] A. S. D. LOS R., M. GARDUÑO G, and A. GONZÁLEZ L, "4Stell-robot: a climbing mobile robot for gas containers inspection", *Proc. 3th WSEAS/IASME Int. Conf. Dynamical Systems and Control,* 2007, pp. 200–205.

[4] J. Lenarcic and M. Stanisic, "A humanoid shoulder complex and the humanoid pointing kinematics", *IEEE Trans. Robot. Autom.*, vol. 19, no.3, pp. 499–506, Jun. 2003.

[5] Kenji KANEKO, Kensuke HARADA, Fumio KANEHIRO, Go MIYAMORI, and Kazuhiko AKACHI, "Humanoid robot HRP-3", *IEEE/RSJ Int. Conf. Intelligent Robots and Systems,* Nice, France, 2008.

[6] M. Friedrich, S. Alexander, and B. Luc, **"**Passive compliance for a RC servo-controlled bouncing robot", *Advanced Robotics*, vol. 20, no. 8, pp. 953–961, 2006.

[7] J. Yamaguchi, A. Takanishi, and I. Kato, "Development of a biped walking robot compensating for three-axis moment by trunk motion", *Proc. IEEE Int. Conf. Intelligent Robots and Systems,* 1993, pp. 561–566.

[8] I. Yamagucbi, E. Soga, S. Inoue, and A. Takanishi, "Development of a bipedal humanoid robot control method of whole body cooperative dynamic biped walking", *Proc. IEEE Int. Conf. Robotics and Automation,* 1999, pp. 368–374.

[9] S. N. Oh, Kab-Il Kim, and S. Lim, "Motion control of biped robots using a single-chip drive", *Proc. IEEE Int. Conf.e Robotics and Automation*, vol. 2, 2003, pp.2461–2465.

[10] O. Dal Y. and O. Richard J., "Which PWM motor-control IC is best for your application", *Motion System Design*, vol.47, no. 7, pp. 22–30,2005.

[11] Microchip,"PIC18F86520/8520/6620/8620 /6720 /8720 Datasheet Search", 2004.

[12] Han-Way Hunag, PIC microcontroller: an introduction to software and hardware interfacing, Thomson, 2005.

Chin-Pao Hung was born in Maioli, Taiwan, in 1964. He received the B.S and M.S. degrees in automatic control engineering from Feng Chia University, Taichung, Taiwan.

He is currently an Assistant Professor with the Department of Electrical Engineering, National Chin-Yi University of Technology. He has published about 15 papers in a wide research field. His research interests include robot manipulator, humanoid robots, robot toys, artificial neural network control, and fault diagnosis system.

Mr. Hung was the recipient of the Excellent Paper Award from the 3th Intelligent Living Technology Conference 2008 in Taiwan (ILT2008) and the Excellent Teacher Award from the National Chin-Yi University of Technology in 2009.

Wei-Ging Liu was born in Changhua, Taiwan in 1986. He is currently a graduated student of Department of Electrical Engineering, National Chin-Yi University of Technology. His research interests include microcontroller application, robot arm design, servo system and humanoid robots design.

Hong-Jhe Su was born in Taichung, Taiwan in 1985. He is currently a graduated student of Department of Electrical Engineering, National Chin-Yi University of Technology. His research interests include artificial neural network, software design, and servo control system.

Jia-Wei Chen was born in Taichung, Taiwan in 1980. He is currently a graduated student of Department of Electrical Engineering, National Chin-Yi University of Technology. His research interests include mechanism design and robotics.

Bo-Ming Chiu was born in Maioli, Taiwan in 1987. He is currently a graduated student of Department of Electrical Engineering, National Chin-Yi University of Technology. His research interests include robot toys design and embedded control system design.