

# Re: Microchip PIC programming question

---

*Source:* <http://sci.tech-archive.net/Archive/sci.electronics.design/2005-07/msg01212.html>

---

- *From:* "Roger Hamlett" <[rogerspamignored@xxxxxxxxxxxxxxxxxxxxxx](mailto:rogerspamignored@xxxxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Sat, 09 Jul 2005 23:15:31 GMT
- 

<liamfoxtrot@xxxxxxx> wrote in message  
[news:1120943853.498031.131800@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:1120943853.498031.131800@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx)

> Hi,  
>  
>  
> I am learning how to program a PIC for fun and creating a project for  
> my car. I have downloaded the microchip.com's simulator and am looking  
> forward to trying out some code. So I figure start small and build my  
> way on up. I am not looking for someone to tell me exactly how to do ,  
> just a few pointers and links would be greatly appreciated.

>  
>  
> I am thinking about getting a PIC16F676 after I know what I am doing a  
> little better.

> The first chunk I want to do is:

- > Input A: 30mv – 75mv
- > Input B: 2.5V – 6.0V
- > Vref: > 6.0V

You are going to have to divide your second input voltage to bring it below the supply voltage of the chip. The same for the Vref. The first input will need amplification to give acceptable resolution.

> Calculate: Ratio A/B

> Output C: PWM % at  $A/B * 32$  (would vary from 16% – 96% at extremes)

> The second chunk I want to do is:

- > Load a 10 point table of min to max ratios and corresponding
- > calibration points from 0–100mv

## Re: Microchip PIC programming question

- >
- > Based on the calculated Ratio A/B, calculate the corresponding mv using
- > a linear extrapolation between the two closest points on the curve.
- >
- >
- > Output D: create another PWM at the calculated mv
- >
- >
- > The third chunk I want to do is:
- >
- >
- > Input E: sensor reading 0–100mv
- >
- >
- > Output F: create another PWM that uses fuzzy logic to get the Input E
- > to match Output D. Increasing Output F will increase Input E.
- >
- >
- > I broke it up into these three pieces since each one had an output that
- > became the input for the next section. I wasn't sure it would all fit
- > on one PIC16F676, but it should all fit on three of them.
- >
- >
- > Thanks in advance for any suggestions, links, tutorials or the like
- > that you think will help me with my project.
- >
- > Thanks
- > Andy

This should fit into a single 676 \_easily\_. I have a board I did some while ago, that takes readings from a Ph sensor, temperature sensor, and ammonia sensor, linearises the ammonia reading based on two calibration points, and a logarithmic curve fit (a lot of arithmetic), outputting a new reading every second. At the same time it performs a servo loop controlling a heater with one of the PWM outputs, and regulating this based on the temperature sensor reading. Finally it converts the Ph sensor voltage into a Ph value. This only uses 85% of the code space in an 676. Realistically, the code would be much quicker to write, using a language like 'C', where in the versions for the PIC, most of the work to perform basic I/O, and the code needed to retrieve tables etc., is already written for you. It is down to whether you really want to learn the PIC assembler or not. There are at least three quite well known C compilers available for the PIC.

Best Wishes

.

Re: Microchip PIC programming question

- **References:**

- ◆ **Microchip PIC programming question**

- ◆ *From: liamfoxtrot@xxxxxxxxxx*

- Prev by Date: **Re: Digital panel meters with VCC < 5V?**
- Next by Date: **Re: Microchip PIC programming question**
- Previous by thread: **Re: Microchip PIC programming question**
- Next by thread: **Re: Microchip PIC programming question**
- Index(es):
  - ◆ **Date**
  - ◆ **Thread**