

Data acquisition with buffering using 8 pin PIC microcontroller

There are many different data-converter chips or micro-controllers in the market that can do the analog/digital conversion, but the idea of using a low price, 8-pin microcontroller that saves space and money and also does the buffering and more, sounds interesting. Look at figure 1, it shows a PIC12F675 which is an eight pin microcontroller and used as a smart A/D with five communication lines.

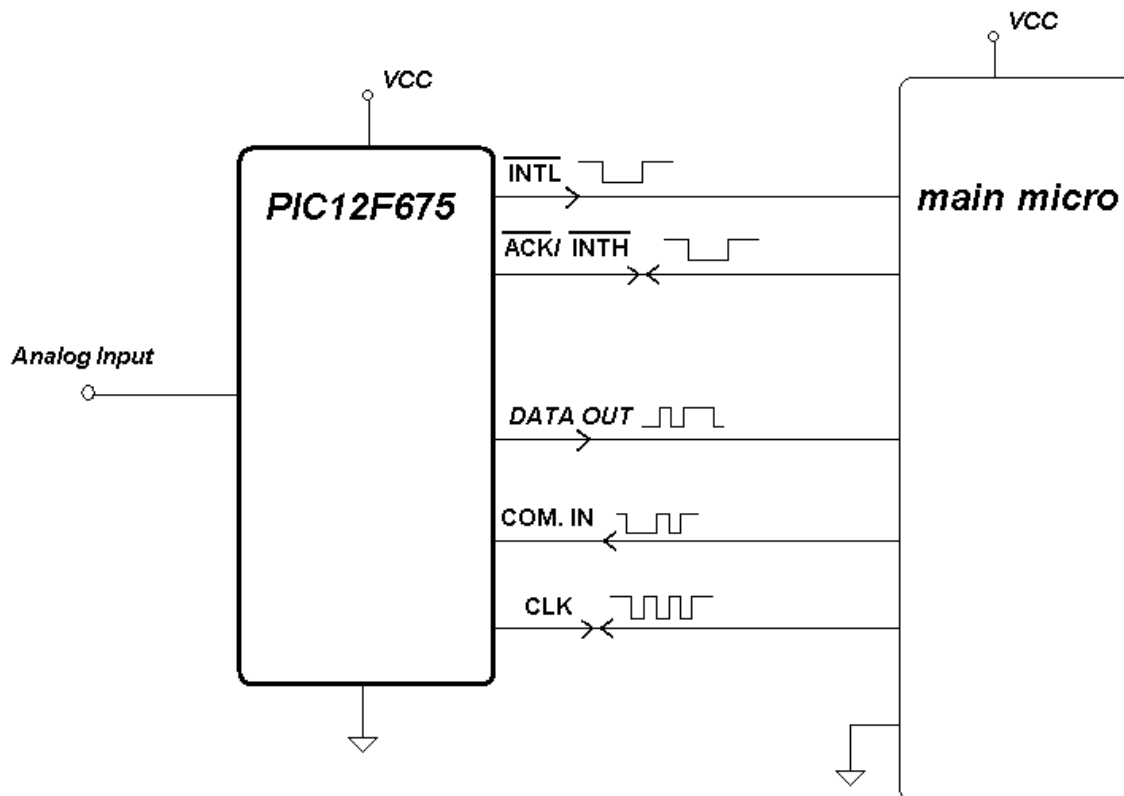


Figure 1: Using an 8-PIN microcontroller from Microchip to make a smart A/D

All five digital lines are in high state at idle mode. CLK signal will be polled in PIC side to respond when a command is send out by main micro. As you see the ACK/INTH pin has two duties: first it receives the acknowledge from the main micro that the main micro is ready to answer the INTL and second it sends a higher priority interrupt to the main micro indicating the buffer is going to overflow in the next sampling time.

How it works:

When a sampling is done a low interrupt (INTL) will be send to the main micro if the main micro is in sleep or for any reason we don't receive the ACK signal, then the sampled data will be saved in a FIFO buffer and for the next sampling it sends another low interrupt again until the buffer is close to be full, then it will send a higher interrupt

(INTH) this may cause the main micro to recover from sleep or interrupt from other tasks to fetch the stored sampled data.

Some pins are defined as bi-directional like CLK and ACK/INTH so be careful about the pin in/out assignment in every stage of the program. It may need a large pull up resistor depending to the type of the main micro.

What are the advantages of using this idea on the side of a main microcontroller just as a smart buffered A/D?

1. If the micro-controller doesn't have the appropriate timing for sampling or can't provide it because of other tasks or necessary timings, then this circuit will help.
2. If the main microcontroller for some reason misses one or some of the interrupts then the smart PIC saves the sampled data in the FIFO buffer.
3. Main microcontroller in the non-busy time can fetch all data in FIFO buffer with three serial communication lines.
4. We can do some filtering on the data stream before sending them to the main microcontroller. This reduces the manipulation load on the main microcontroller.
5. This is possible to send a command to the PIC micro, by using the serial communication line too; these commands may adjust the sampling rate, data length, filtering parameters and so on.
6. If you are dealing with a nonlinear system, like a non-linear pressure sensor you can make the data linear before sending.
7. You can also do any function on the data before transmitting to the main processor.

The idea of using this smart A/D reduces the complexity of the main micro firmware and also it helps to reduce the power consumption because it help the main micro to spend more time in sleep.

It is obvious that, depending on your system, you can modify the signals for your specific application and you may use another micro with higher number of pins to do parallel data exchange for higher speeds, but you need more assigned pins of the main micro and your cost increases as well.

Mazi Hosseini M.A.Sci. P.Eng
Astinco