

## 15.5 Other Practical Considerations

**For basic prototype circuit assembly and troubleshooting advice, see Section 7.4 in Lab 7. For advice for debugging circuits with PIC microcontrollers see Section 10.4 of Lab 10.**

Here are some other practical suggestions for microcontroller-based designs:

### General Electrical Design Suggestions:

- When ordering ICs, make sure you specify **DIP** (dual in-line packages) and not surface mount packages (e.g., SOP). DIP chips are well suited to use in breadboards and protoboards. Surface mount ICs require printed circuit boards (PCBs) and special soldering equipment.
- Make sure your power supply can provide **adequate current** for the entire design. If necessary, use separate power supplies for your signal and power circuits.
- Use **breadboards** with caution and care because connections can be unreliable, and the base plate adds capacitance to your circuits. Hard-wired and soldered protoboards or printed circuit boards (PCBs) can be much more reliable. See Section 15.4 for advice on how to solder properly. Be sure to use sockets for all ICs to prevent damage during soldering and to allow easy replacement of the ICs. Also, if you have a working breadboard circuit, it is advisable to use duplicate components (where possible) for the soldered board (i.e., don't cannibalize components from a working prototype circuit in case something goes wrong or gets damaged when soldering your board).
- Use a **storage capacitor** (e.g., 100  $\mu\text{f}$ ) across the main power and ground lines of a power supply that does not have built in output capacitance (e.g., batteries, wall transformers, and regulated voltages) to minimize voltage swings during output current spikes. Also, use **bypass capacitors** (e.g., 0.1  $\mu\text{f}$ ) across the power and grounds lines of all individual ICs to suppress any current and voltage spikes.
- Make sure all components and sources have a **common ground** unless using relays, wireless interfaces, or opto-isolators, in which case you should keep the independent power supply grounds separate.
- Avoid **grounding problems and electromagnetic interference (EMI)**. Section 2.10 in the textbook presents various methods to reduce EMI, specifically using opto-isolators, single point grounding, ground planes, coaxial or twisted pair cables, and bypass capacitors.
- Don't leave **IC pins floating** (especially with CMOS devices). In other words, connect all used and functional pins to signals or power or ground. As an example, do not assume that leaving a microcontroller's reset pin disconnected will keep a microcontroller from resetting itself. You should connect the reset pin to 5V for an active-low reset or ground for an active-high reset, and not leave the pin floating where its state can be uncertain.
- Be aware of possible **switch bounce** in your digital circuits and add debounce circuits

or software to eliminate the bounce.

- Use **flyback diodes** on motors, solenoids, and other high inductance devices that are being switched.
- Use **buffers**, line drivers, and inverters where current demand is large for a digital output.
- Use **Schmitt triggers** on all noisy digital sensor outputs (e.g., a Hall-effect proximity sensor or photo-interrupter).
- Use a **common-emitter** configuration with transistors (i.e., put the load on the high side) to avoid voltage biasing difficulties.
- Be careful to identify and properly interface any **open-collector** or open-drain outputs on digital ICs (e.g., pin RA4 on the PIC).
- For reversible dc motors, use "off-the-shelf" commercially available **H-bridge** drivers (e.g., National Semiconductor's LMD 18200) instead of building your own.

PIC-related Suggestions (see more in Section 10.4 of Lab 10):

- Follow the microcontroller design procedure in Section 7.9 of the textbook.
- Modularize your software and independently develop and test each module (i.e., don't write the entire program at once expecting it to work).
- Use LEDs to indicate status and location within your program when it is running, and to indicate input and output states.
- Be aware of the different characteristics of the I/O pins on the PIC. Refer to Figures 7.15 and 7.16 in the textbook to see how to properly interface to the different pins for different purposes.
- Be aware that PicBasic Pro commands totally occupy the processor while they are running (e.g., the line after a SOUND command is not reached or processed until the SOUND command has terminated).
- Refer to Design Example 7.1 in the textbook for ideas on how to interface to 7-segment digital displays with a minimum number of pins.
- When prototyping with a soldered protoboard or printed circuit board, use IC sockets to allow easy installation and removal of the PICs without damaging pins. Also, always use a "chip puller" tool to remove ICs (e.g., PICs) from breadboards or soldered IC sockets.