Application Based Microcontroller Selection
Benjamin P. Guild
April 2, 2010

Executive Summary:
Microcontrollers are a nearly ubiquitous technology. Their versatility allows them to be used in nearly any device that needs an electronic control system. The selection of a microcontroller for a specific application can be a challenge as there are numerous vendors, specifications, and design requirements. This abundance of variation can make selecting a microcontroller a time consuming task. This application note is intended to help facilitate microcontroller selection by discussing the factors to be considered for an application.

Introduction:
When selecting a microcontroller for an application, the main selection criteria that need to be considered can be divided into two main parts. These are system requirements and application specific requirements. As with any electronics application, the best reference to the technical specifications is the device datasheet.

Keywords:
microcontroller (MCU), design specifications, datasheet, development environment

System Requirements:
When selecting a microcontroller for an application, there are several system requirements that should be considered. The most common requirements are power consumption, programming interfaces and support, and mounting requirements.

- Power Consumption
Power consumption is a determining factor in many applications of microcontrollers. Many embedded systems are battery operated or only demand intermittent use and require the microcontroller to be energy efficient. Most microcontroller vendors offer specific microcontroller lines designed for low-power applications and energy consumption if that is a requirement of the design. Energy savings are commonly achieved through low-power modes or selective activation. The common tradeoff to decreased power consumption is often clock and processing speed. Careful consideration should be given when determining if a low power microcontroller is appropriate for a specific design. An example of an application where a low power microcontroller may not be a feasible design choice would be a real-time or continuous monitoring application or where the design requires high computational complexity to be performed by the microcontroller.

- Programming Interface and Support
While all microcontrollers run at the lowest level in assembly/machine code, most modern microcontrollers are commonly programmed using a development environment. A development environment usually consists of a user interface that allows for high level programming and a compiler or translator that does the work of writing the
assembly and/or machine language necessary for the processor to run an application. There are also development environments that allow direct memory manipulation and low level code writing. Common development environments utilize various languages for programming, the most common of which are C and BASIC. Some development environments implement unique or proprietary interface languages or variations on common languages. The programming requirements of a microcontroller need to be taken into account. Microchip, for example, provides a development environment for programming in C language. The figure below is a screenshot of the MPLAB IDE from the microchip website.

![MPLAB IDE](image)

This is an example of a development environment that allows high level programming as well as low level process manipulations.

- **Mounting Requirements**
  Microcontrollers come in various shapes and architectures. Nearly all microcontrollers come in a package consisting of a plastic, ceramic, or epoxy resin. This both protects the device and allows for the electrical connections to come in many forms. As microcontrollers have become increasingly miniaturized, the mounting technologies and techniques have developed. Some mounting packages such as dual inline package (DIP), single inline package (SIP), and small-outline integrated circuit (SOIC) can be hand soldered to a circuit board relatively easily. Other packages such as surface mounted packages require more advances soldering techniques such as reflow soldering.
Reflow soldering is where an adhesive soldering paste is used to temporarily hold the microcontroller and other devices to a circuit board. The board is then passed through a machine that heats the paste, through the use of specialized reflow oven technology or through the use of infrared lamps, to the point where the solder permanently connects the device to the board. Adapters are also a common and useful tool used in the development of a microcontroller designs. They allow for non-permanent mounting using mechanical contact as opposed to soldering. They are very useful in cases where it is possible to have volatile power source fluctuation that could damage the microcontroller, making replacement a necessity. Once development is completed, finalized designs usually incorporate a more permanent mounting design.

**Design Requirements:**
Selecting a microcontroller for an application requires knowledge of how the controller will be used and what it needs to be accomplished. Based upon this, there are two main factors to consider, I/O and onboard devices.

- **I/O**
  In selecting a microcontroller for a specific application, one of the factors in selection is what devices need to be connected to the microcontroller and how those devices will communicate with it. In many microcontrollers, there are pins that are configurable to either accept input signals or produce outputs. An example of this is the Microchip PIC18F4520. There are multiple ports that can either be used for internal devices or as general input/output ports. This device has four eight-bit ports and one four bit port that are configurable between onboard devices and general purpose I/O. These are software configurable and can be utilized as needed. Many microcontrollers also have ports that are specifically utilized by an onboard device or are configured for a specific use.

- **Onboard Devices**
  Microcontrollers usually offer a collection of devices on the chip. The most common are devices such as analog to digital (ADC) or digital to analog (DAC) converters, communication interfaces such as SPI, I²C, RS232, USB, wireless, as well as other
specialized interfaces. Another property of a microcontroller that should be taken into account is flexibility in peripheral support. Several microcontrollers need peripheral devices such as external clocks to manage a system or to synchronize communications.

**Conclusion:**
The selection of a microcontroller for an application can be a challenging process. The key to microprocessor selection is identification of the necessities of the design. The aforementioned categories are a good basis to identifying what to look for in terms of a microcontrollers form and function. As mentioned previously, the datasheet for a microcontroller is the best reference to the device capabilities and functions and should be used heavily in the selection process.
References:
- Provides a broad list of microcontroller vendors and products

http://www.microchip.com/
- Microchip microcontrollers are common in many applications and Microchip offers many application and program development tools

http://www.ti.com/
- TI offers many microcontrollers and they have a very useful MCU Selection tool http://focus.ti.com/mcu/docs/mcuhome.tsp?sectionId=101&DCMP=TIHeaderView

http://www.atmel.com/
- Atmel microcontrollers are used in various applications including the increasingly popular Arduino prototyping platforms

- A good list of microcontroller package types with many links to common packages for more detail
