MICROCONTROLLERS

Mohammad Jafar Navabi
Medtronic Microelectronics Center, Tempe, Arizona, USA

Keywords: Microprocessor, peripheral devices, CPU, I/O, analog to digital converter, digital to analog converter, CISC, RISC, Von Neuman architecture, Harvard architecture, interrupt handler.

Contents

1. Introduction
2. Applications of Microcontrollers
3. Architecture
4. Sample Microcontrollers
5. Software
6. Hardware
6.1. Interrupt Handlers
6.2. I/O ports
6.3. Digital to Analog Converters (DAC)
6.4. Analog to Digital Converters (ADC)
6.5. Serial Communication Interface
6.6. Timers and Clocks
6.7. Memory
6.8. External Device Drivers
7. Current/Future Trends
8. Conclusion
Glossary
Bibliography

1. Introduction

Since the inception of integrated circuits and with the advances in technology, more and more devices have been placed in a single integrated circuit. Intel Corporation's 4004, the first microprocessor introduced in 1971, was made possible by high levels of integration of digital circuits. Continued integration of peripherals and memory on the same integrated circuit as the microprocessor core led to the creation of microcontrollers. A microcontroller is an integrated circuit composed of a CPU, various peripheral devices, and typically memory, all in one chip. Using one chip that contains all the necessary functions in place of a microprocessor and multiple peripheral chips has reduced the size and the power consumption of control oriented applications.

A microcontroller is different from a microprocessor both in hardware and software. In hardware it includes peripherals such as I/O, memory, and analog and digital interface. Microcontrollers are more suited for small applications with specific control functions requiring specialized peripherals and interfaces. They are designed for process control and are required to interface to the real world processes. Many of the peripheral devices integrated on a microcontroller are for that specific purpose. Analog to digital

converters perform the task of converting an analog signal to digital for use by the CPU, and digital to analog converters perform the task of converting digital data into analog value and waveforms to control analog functions. In addition to the analog interface, microcontrollers contain peripheral devices that enable them to communicate to other digital components within a system or to monitor and control digital functions. Communication interfaces, digital I/O and interrupt controllers fall into this category of peripheral devices. Other peripheral devices often included on the same chip include clocks and timers.

In terms of the software, microcontrollers have a more compact set of instructions with commands more suited to process control such as input and output from. Single bit operations such as set and reset, bit-wise logical functions or branching instructions that depend on a single bit are commonly available as part of the instruction set to allow for reading input switch status or on/off control of an external event. Since in a given application the microcontroller is programmed for one task, it only has one control program. In a microprocessor based system various programs are stored in a mass storage device and then loaded into the RAM for execution. In contrast the microcontroller program is typically stored in a ROM or PROM and RAM is used for temporary storage of data.

Compared with discrete implementation of a system, the microcontroller based approach provides shorter system development time, reduced implementation cost, lower power consumption, and higher reliability. The only drawback, which is often not important, is the lower speed of execution. For example, for a microcontroller system to perform a logical operation, several clock cycles are needed to read the inputs, perform the function and output the results. The same operation when implemented with discrete components will provide the results as soon as the signals have propagated through the logic gates.

Micro-controllers are used in a variety of process control applications, replacing complex digital circuits and sometimes analog functions while providing more flexibility due to their programmability. Portable electronic devices such as personal audio devices (CD players, MP3 players), mobile telephones, digital cameras and video camcorders rely heavily on the reduced size and low power consumption of microcontroller based electronics. These features are crucial to applications like implantable medical devices such as pacemakers, or personal medical monitoring devices like glucometers (electronic devices used for the measurement of blood glucose). In other applications such as appliances, home audio and video, automotive, power management, and temperature control, using a microcontroller results in reduced board level circuit complexity and consequently reduced cost. With the growing number of applications using microcontrollers, it is not surprising that there are such a wide variety of these components. In addition to those commonly available, many manufacturers custom-design a microcontroller to suit a specific application.

2. Applications of Microcontrollers

Having different components of a system in one chip and ready to use, reduces the complexity of the system level design and therefore time to market. This fact along with

cost reduction has resulted in the use of microcontrollers in a multitude of applications ranging from industrial control to home appliances and electronics.

Charging a battery is a seemingly simple task but it involves a number of control and monitoring functions. In order to properly charge a battery it is necessary to continuously monitor voltage and current and detect the charging state. Another parameter of interest during charging is the ambient temperature. Charging methods vary with battery types. For a versatile charging system it is required that the charger identifies battery type from its terminal behavior. These characteristics have led to the application of microcontrollers in battery chargers. Charger manufacturers typically use 4 bit or 8 bit microcontrollers with analog to digital conversion capability. The ADC requirements are not very stringent and few additional peripheral devices are needed.

Automotive industry has been using microcontrollers for a number of tasks including powertrain engine management, anti-lock brakes, diagnostics, lights control, and entertainment. Many other electronic enhancements to future cars would take advantage of microcontrollers. Electric vehicles require extensive power management, drive-by-wire refers to electrical transmission of driver's input to the automobile's system rather than the conventional mechanical methods, collision avoidance / obstacle detection would use input from a radar to control steering and speed. These are examples of trends in auto industry where microcontrollers will play a significant role. In applying electronics to the automotive one should be aware of extra burdens placed on such devices. Automotive controllers and other electronics have to withstand extended temperature ranges and be able to operate accurately in the presence of noise.

An advantage of integrating various components into one chip is the increased power efficiency. One source of this increase in efficiency is that there is no need to drive signals off chip. A signal traveling from one point on a chip to another on the same chip will only have to overcome small capacitances associated with the thin traces on the chip. To drive a signal from one chip to another it becomes necessary to have large and high-power line drivers and special signal receivers on the chips to ensure signal integrity as it travels across the chip boundary. Another reason for increased efficiency is that integration often results in elimination of otherwise redundant circuitry. For example when an analog to digital converter is integrated on the same chip as the microprocessor, it can use the CPU or general purpose counters to do some of its processing functions. Or, one clock generator can be used for the CPU, analog to digital converter and serial I/O timing. The low power characteristics of a microcontroller based design makes it especially suitable to battery operated applications such as portable electronics and implantable medical devices.

An implantable pace-maker is a device that is surgically placed under the skin at the shoulder and delivers electrical pulses to the heart via a pare of electrodes. Since these devices are implanted through surgery, low power consumption, and compact physical size is extremely important. A pace-maker uses analog to digital converters for monitoring various vital signs, digital to analog converters for generating the pulses, microprocessor for calculations and data control, and memory for storage of temporary data and semi-permanent calibration information. There is usually at least a serial communication interface for programming, testing, and calibration. 8-bit

microcontrollers are typically sufficient for the computational power needed, but the ADC and DAC need to have high accuracy.

Another example of a low power application of microcontrollers is in personal portable audio such as Internet Audio Players. Manufacturing profit margins in this application are relatively low and the manufacturers must drive costs down and offer advantages such as lower power consumption in order to compete in the market.

Some of the other applications where microcontrollers have been used in order to improve performance or reduce cost include: industrial control, power management, home entertainment, appliances, temperature control, medical instruments and monitoring, wireless communication, and digital video recorders and cameras.

3. Architecture

Architecturally all microcontrollers share certain features. They all contain a CPU, memory and I/O on the same chip. Another common feature is the interrupt handling capability. What sets them apart from one another is the choice of CPU, the structure of memory, and choice of peripheral devices, I/O and interrupt handling hardware.

The major distinguishing architectural characteristic of microcontrollers is the word size. Micro-controllers are available in 4, 8, 16, or 32 bit wide words. The width of the data path impacts several features of the microcontroller. The complexity of the instruction set (number of available instructions and addressing modes), program efficiency (code generation and storage space), execution speed, as well as chip implementation and interfacing complexity (as it relates to bus width and data and address decoding) are all influenced by the width of the data path.

For simple control tasks 4-bit, and for a vast number of control and measurement applications 8-bit microcontrollers would suffice. For higher precision and speed applications like speech and video processing, or complex instrumentation, 16-bit and 32-bit microcontrollers are more appropriate.

Another distinction between microcontrollers is the instruction set. Micro-controllers with complex instruction set (CISC) provide capability to perform complex computations rapidly. The extensive set of instructions, allow complex operations to be performed with few instructions. On the other hand reduced instruction set computers (RISC) decrease program execution time by having fewer less complex instructions. Fewer available instructions results in faster execution due to smaller size of the op-code and less decoding time needed for each instruction. The trade-off depends on the complexity of operations needed for a specific application. In simple control applications a RISC based microcontroller is more suitable because of its lower overhead for each instruction. In more complex applications, the availability of a more diverse instruction set results in a more efficient and faster executing code because fewer instructions are needed to accomplish a complicated task. For microcontroller applications the instruction set should include common computational instructions plus instructions optimized for the specific application at hand.

Just as in microprocessors, microcontrollers are also differentiated according to their memory structure. Von Neuman architecture maps the data and program to same memory address space. In the Harvard architecture the instructions are stored in a separate memory space than that used for data storage. Another memory related architectural characteristic of a processor is the addressing scheme. In linear addressing there is a one to one correspondence between an address and a memory location. So with an 8-bit address register, 2^8 distinct address locations can be accessed. In segmented addressing a separate register is used to point to a segment in memory, and the address register is used to point to an offset from that segment's start point. This way if all of the program or data are in the same segment, in order to access them, only the address register need to be used and the segment register can remain pointing to the start point of that segment.

-
-
-

TO ACCESS ALL THE **15 PAGES** OF THIS CHAPTER,
Visit: http://www.eolss.net/Eolss-sampleAllChapter.aspx

Bibliogrphy

Garrett, Patrick H. 1981. *Analog I/O Design, acquisition: conversion: recovery*, Reston Publishing Company, Inc. [This presents methods of digital to analog and analog to digital conversion and design of these circuits.]

Peatman, John B. 1988. *Design With Microcontrollers*, McGraw-Hill Book Company. [Covers different aspects of microcontrollers including software, hardware and peripherals.]

Cady, Fredrick M. 1997. *Software and Hardware Engineering, Motorola M68HC11*, Oxford University Press [Provides a guide to hardware and software engineering for the Motorola 68HC11.]

Hintz, Kenneth; Tabak, Daniel 1992. *Microcontrollers, Architecture, Implementation, & Programming*, McGraw-Hill, Inc. [Provides detailed description of microcontroller and its components with many examples for common microcontrollers.]

Cady, Fredrick M. 1997. *Microcontrollers and Microcomputers, Principles of Software and Hardware Engineering*, Oxford University Press [Presents an in depth look at micorcontroller and its architecture, programming and peripheral devices.]

Chang, Henry; Cooke, Larry; Hunt, Merrill; Martin, Grant; McNelly, Andrew 1999. *Surviving the SOC Revolution*", Kluwer Academic Publishers. [This book presents the challenges of highly integrated design and proposes new approaches for designers and design technology developers.]

Stewart, James W. 1993. *The 8051 Microcontroller. Hardware, Software & Interfacing* Regents/Prentice Hall [This is a guide on the Intel 8051 and provides information on interfacing.]

Mackenzie, I. Scott 1997 *The 8051 Microcontroller* Prentice Hall [This is a guide to the 8051 hardware architecture, instruction set, features and interfacing. It also covers program design and assembly language programming.]