Table of Contents

ACKNOWLEDGMENTS

"Microprocessor Design Made Easy Using the MC68HC11" was designed and developed through the talents and energies of many hard working and dedicated individuals.

Dedicated staff at AMS, Todd Sutter and Sunita have spent several hundred hours towards this book. To each of them we extend our thanks and appreciation.

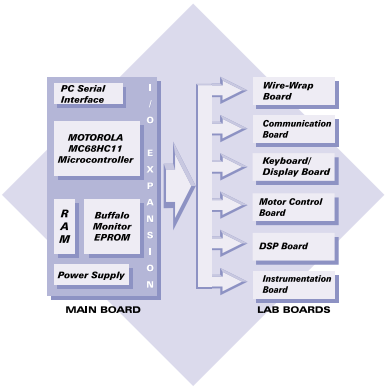
We would like to thank this opportunity to thank Dave Miller from the new Hampshire Technical Institute who helped us very extensively in preparing this book.

Library of Congress Catalog-in-Publication Data

Microprocessor Design Made Easy Using the MC68HC11
Includes Index.
ISBN 0-9642962-4-1
First Edition

Microprocessor Design Made Easy Using the MC68HC11
Imprint 1995

Advanced Microcomputer Systems Inc
1460 SW 3rd street Pompano Beach, FL 33069
Phone: (954) 784-0900 • Fax: (954) 784-0904
E-Mail: info@advancedmsinc.com



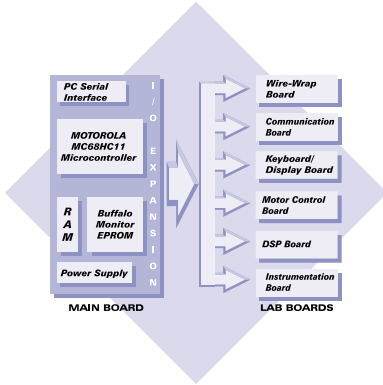
Lesson One

Introduction to Microprocessors

1. The history and evolution of the microprocessor.

2. The different types of microprocessor available and their features.

3. The difference between 8-bit, 16 bit and 32- bit microprocessor.

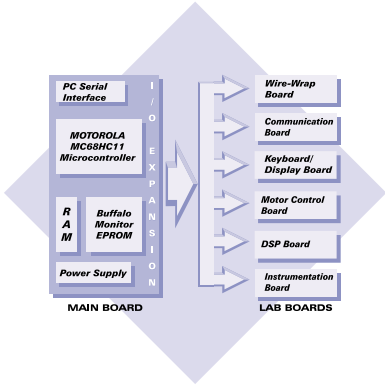


Lesson Two

The MC68HC11 Architecture and its addressing modes.

After completing this chapter you will learn:

1. The internal structure of the MC68HC11.
2. The different type of the addressing modes.
3. How binary information moves inside an MC68HC11 based system.

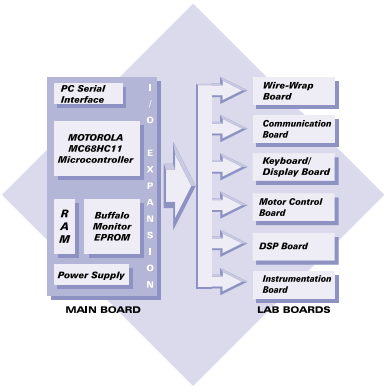


Lesson Three

Programming the MC68HC11

After completing this chapter you will learn :

1. The different instructions groups of the MC68HC11
2. How to create common data structure in MC68HC11 assembly.
3. How does EZ-MICRO TUTOR manager software works.
4. The commands of the EZ-MICRO TUTOR Manager Software.
5. How to use an assemble and debugger.

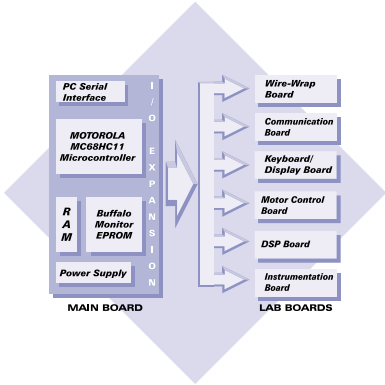


Lesson Four

How the AMS Tutor Board works:

After completing this chapter you will learn :

1. The architecture of the AMS Tutor board.
2. The feature and capabilities of the AMS Tutor board.
3. How to download programs on the AMS Tutor board.

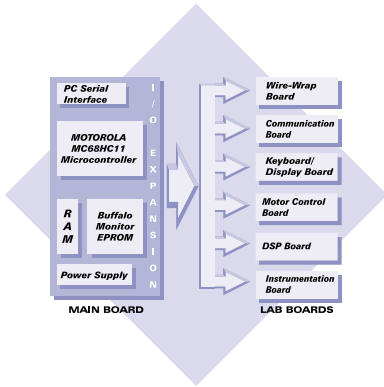


Lesson Five

Software Interface to AMS Tutor Board

The objectives of this laboratory are:

1. To become familiar with the AMS 68HC11 Tutor package
2. To set up the PC/AMS TUTOR DEVELOPMENT system.
3. To be introduced to the assembler source code syntax and assembly process.
4. Exercise the software development tools

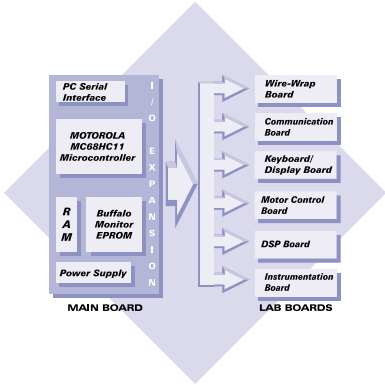


Lesson Six

Assembling Language Programming

Overview of the Lab :

This lab will involve developing code from a specification. The specification will be translated into pseudo-code, then into assembly language, then downloaded to Tutor board and debugged.

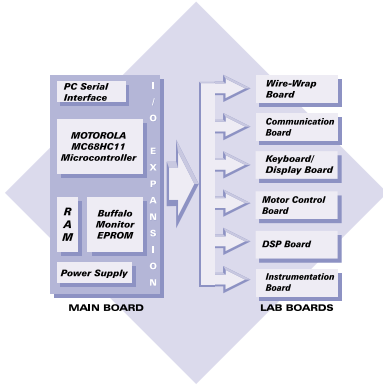


Lesson Seven

Application Programming: Look up Tables and Subroutines

1. This laboratory will involve the exploration of data arrays and look-up tables (LUTs). These data handling techniques have many applications in embedded systems. They can provide the embedded systems designer the capability to generate or translate data that cannot be calculated due to timing constraints.

2. Subroutines : Use of subroutines to make your program efficient structured program

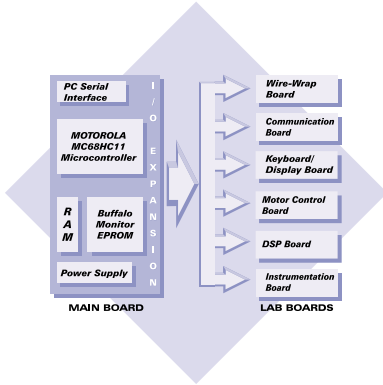


Lesson Eight

How the AMS Tutor Board interfaces with outside world

This laboratory will involve interfacing input and output devices to the AMS Tutor Board. The input device will be a keypad, and the output device will be three 7 segment LED display.

- Asynchronous Vs Synchronous send handshake
- Send/Receive handshake
- Interrupt Vs Polled
- Single direction Vs Bi-directional Vs Bus Interconnection

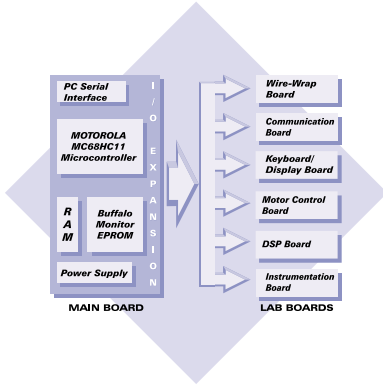


Lesson Nine

Concept of Interrupt for measuring period of input signal using Timer

Overview

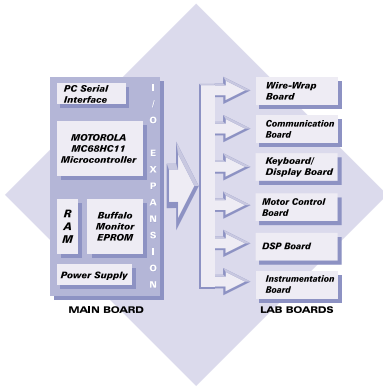
This laboratory experience will involve two major concepts: measuring the period of an input signal using the input capture timer, and interrupts. These two methods work together to perform a complex task with an efficient combination of hardware and software.



Lesson Ten

Generate output pulse using internal timer.

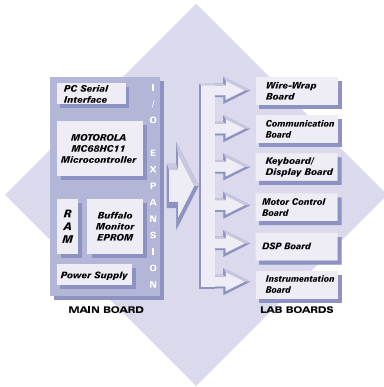
The 68HC11 Microcontroller timer system is a flexible and powerful tool for solving a wide variety of problems that involve measuring and generating time series waveforms. This lab will involve using the timer features that output pulses at precise time intervals.



Lesson Eleven

Introduction to Analog to Digital conversion.

Analog to digital conversion is the process of transforming a continuous signal into a discrete representation. This process has been described as the process of converting real world signals to a form that is used in the artificial world of digital systems. Extensive theoretical and mathematical studies have been performed to analyze the A/D process. This introduction will try to identify the key concepts involved with A/D conversion.



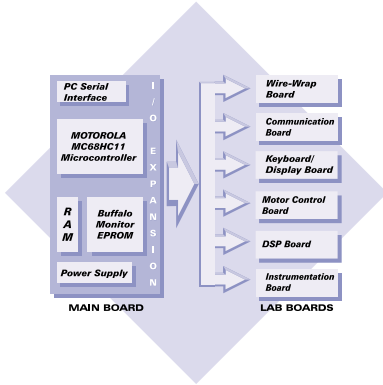
Lesson Twelve

Analog Input and Output

In this laboratory, the program will be developed to develop the driver to test the analog input and output features.

The specification is as follows:

Write a program which will generate a 2 kHz sample rate clock to convert an analog sample and output this sample value to the DAC port. Set up a function generator to provide a sinusoidal signal with an amplitude range of 0 to 3 volts and a frequency range of 5 to 3 kHz. Use an oscilloscope to monitor the waveforms at various points in the circuitry to observe the action of the DSP board at different input frequencies.



Lesson Thirteen

Introduction to Digital Signal Processing

Digital Signal Processing (DSP) describes the process of manipulating signals while they are represented in a discrete format. Computer systems are capable of performing a wide variety of processes on signals under software control. One common class of DSP algorithms is the digital filter.