ENGG3640, Microcomputer Interfacing Lab Outline

School of Engineering,University of Guelph
Fall 2004

Embedded computer systems, which are electronic systems that include a microcomputer to perform a specific dedicated application, are ubiquitous. Our global economy, food production, transportation system, military defense, communication systems, and even quality of life depend on the efficiency and effectiveness of these embedded systems. As system engineers we play a major role in all phases of this effort: planning, design, analysis, manufacturing, and marketing.

1 Laboratory Objectives and Practices

The ENGG3640 Microcomputer Interfacing labs are an integral part of the course. The objectives of the laboratory are:

- to give you practical experience with the process of design and interfacing using the Motorola MC68HC11.
- to help you understand and assimilate the lecture material.
- to give you practical experience with the process of design and implementation of analog/digital circuits.
- to give you hands-on-experience with microcontrollers such as the MC68HC11.

Laboratory Practices

1. Absolutely no food or drink in the laboratories.
2. Clean up after yourselves. Put paper in the recycling bin and garbage in the trash.
3. Do not leave the door or windows open.

The room will be closed after-hours if these rules cannot be followed.!

Laboratory Recommendations

- Labs are to be done in groups of at most 3 people.
- The labs are to be demonstrated during the lab period on the due date.
- All written lab reports are due during demonstration of the the following lab.
- Students are encouraged to work ahead.

2 The MC68HC11 Microcontroller

At the core of the 68HC12 is CPU12, a high-speed 16-bit evolution of the 68HC11 architecture that is designed to maintain complete source-code compatibility with the 68HC11 core. The 68HC12 fully supports all internal registers, instructions, addressing modes, and operating modes of the 68HC11.

Additional features and benefits include:

- Up to 8 MHz bus speed at 5 V
- Up to 5 MHz bus speed at 3 V
- 64 new instructions, 20-bit Arithmetic Logic Unit (ALU), instruction queue, and 7 new indexed addressing modes
- 16-bit data paths
- 64 Kbytes of linear and paged memory addressing capability, enabling access to more than 4 Mbytes of program space and 1 Mbyte of data space
- Call and Return from Call (RTC) instructions for efficient paged addressing
- High-level Language (HLL) optimization
- Fast math capabilities (16 x 16 MUL, 32 w 16 DIV, EMACS)
- Fuzzy logic instructions for simplified programming, reduced code size, and faster code execution
- Low-power wait and stop modes

High Performance Timer

The 68HC11 timer provides flexibility, performance, and ease of use. The system is based on a free-running, 16-bit counter with a programmable prescaler, overflow interrupt, and separate function interrupts.

Additional M68HC11 timer features include:

- Multiple timer channels
  - Each channel configurable for either input capture or output compare functions
- Real-time periodic interrupts
- Computer Operating Properly (COP) watchdog protection against software failures
- Pulse accumulator for external event counting or gated time accumulation

Analog-to-Digital Converter (ADC)

The ADC periodically samples external analog signals and produces corresponding digital values. Typical applications are measuring analog inputs like battery voltage, temperature, pressure, and fluid levels. Linear successive approximation 8-bit or 10-bit resolution Single or continuous conversion modes Multiple result registers Selectable ADC clock Analog multiplexor allows variable number of channels with a single ADC.

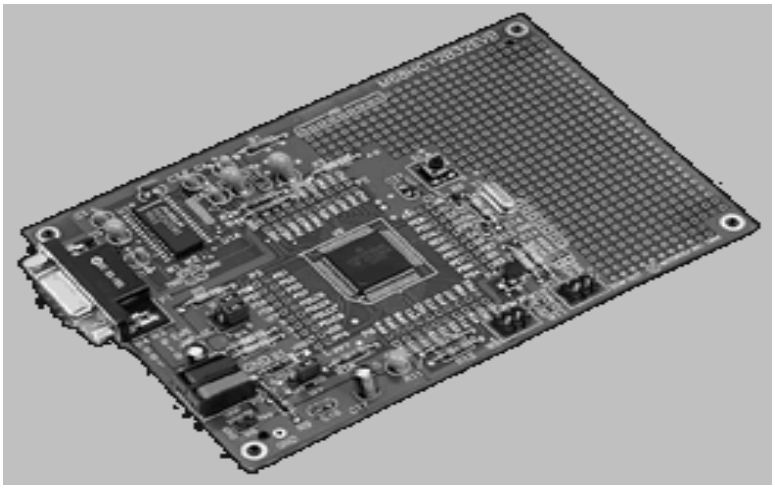

Figure 1: Evaluation Board

Evaluation Board

The M68HC11EVBU Evaluation Board (EVBU) is an economical tool for designing and debugging code for, and evaluating the operation of, the MC68HC11E9 microcontroller. By providing the essential MCU timing and I/O circuitry, the EVBU simplifies user evaluation of prototype hardware and software. The EVBU features a prototype area, which allows custom interfacing with the microcontroller's I/O and bus lines.

3 Laboratory Assignments

The course labs are structured as six assignments. They cover the following material:

- lab0: Introduction to lab equipment and MC68HC11.
- lab1: Introduction to MC68HC11 Controllers.
- lab2: Parallel Port Programming.
- lab3: Keypad and LCD Interfacing.
- lab4: Servo motor control and Hex keypad interfacing.
- lab5: High-Tech Etch-a-sketch module and LCD interfacing.

These labs will expose you to a range of design activities, tool functionalities and implementation technologies. Each lab has a set of deliverables. These typically include the circuit schematics, demonstration of the hardware implementation and a complete report.

Due Dates

There will be 5 labs throughout the term. The following are the due dates.

Item	Takes Place	Due Date	Topic
Lab #0	Fri Sep 12 (Week #1)	-	Equipment in the Lab
Lab #1	Fri Sep 19 (Week #2)	Fri Sep 26 (Week #3)	Introduction to MC68HC11/12 Controllers
Lab #2	Fri Sep 26 (Week #3)	Fri Oct 03 (Week #4)	Parallel Port Programming
Lab #3	Fri Oct 03 (Week #4)	Fri Oct 17 (Week #6)	Keypad and LCD Interfacing
Lab #4	Fri Oct 17 (Week #6)	Fri Oct 24 (Week #7)	Servo Motor Control/Hex Keypad
Lab #5	Fri Oct 24 (Week #7)	Fri Nov 07 (Week #9)	High-Tech Etch-a-sketch

Demonstration

Be prepared to:

- demonstrate the operation of the system
- explain how your design works
- explain how the components of the system works

Writeup

- **Problem Statement**
- **Assumptions and Constraints**
- **System Analysis, Design and Justifications of Decisions**
- **Hardware**
  1. used components and their specifications
  2. schematic
  3. brief explanation of operation
  4. any required calculations
  5. timing diagram if appropriate
- **Software Modules with Specifications**
- **System**
  1. brief user's manual for the system
  2. if the system did not work, summary of current state of the system and how you would resolve the problems
- **Error Analysis**
- **Performance Analysis**

4 Microcomputer Interfacing - Project (Optional)

The project is intended to allow you to express your creativity by applying what you have learned in this course to a project of your own choosing. You are to design and implement a 2-week project of your own choosing that uses MC68HC11 Microcontrollers in some creative way. You may use any of the parts available in the lab. An important part of this lab is the creativity required to think up an interesting project, and then negotiate with a TA or instructor as to the final form of the project.

Originality Approval

The first step in your project is to come up with an idea (an original idea is preferred of course!). You must submit your idea in a 1-3 line description, via email to your TA/instructor for "originality" approval. The TA/instructor will quickly respond to tell you if the idea has already been proposed more than once or twice. If it has, you'll have to come up with something different. Note that the approval is the first step and only deals with the basic idea, and not the scope/effort required for the project.

Complete Proposal

You will submit a two-page project proposal of what you project is about. This should be a short description that gives:

- The basic idea of the project, and the basic function to your circuit.
- Describe the inputs and outputs, and give a simple block diagram describing how the various parts of your circuit interact.
- Your plan of action for each of the two lab periods – "milestones".

Present this to the TA or instructor to get their opinion on whether the project is viable. Once approved, you should get their signature. This is just a check to make sure that you do not try something overly ambitious.

Written Report

You will be required to hand in a written report describing your project, which is due **Friday Nov 30th, at 5PM**. The report should have a **maximum of 1500 words** and contain the following sections:

- Introduction/Motivation – what you're doing and why.
- A description of the design, in both illustration (block diagram) and words. This should be a well-written easy-to-read document!
- A description of the working/not working status of the project If it did not work, indicate the reasons why you think it didn't work.
- Indicate what you would do differently if you were to start the project over again. What did you learn.