ECE 3120 Computer Systems

Manjeera Jeedigunta http://blogs.cae.tntech.edu/msjeedigun21 Email: msjeedigun21@tntech.edu Tel: 931-372-6181, Prescott Hall 120

□ Prev:

- Basic Computer Organization and Concepts
- Hardware issues
- □ Today:
 - Software
 - CPU Registers
 - Data Types
 - Memory Addressing

Software

- Programs. A program is a set of instructions that can be executed by the computer hardware.
- □ Machine instructions:
 - A sequence of binary digits that can be executed by the processor

e.g: 000110000000110 (Accumulator A<--Accumulator A+B)

Hard to understand, program, and debug for human being

Assembly level language

- Defined by assembly instructions
- An assembly instruction is a mnemonic
 representation of a machine instruction (e.g: ABA)
- Assembly programs must be translated before it can be executed (assembler)
- Programmers need to work on the program logic at a very low level and can achieve high productivity.

Assembly Language examples

line	addr.	machine code	source	source code	
1:		= 00001000	org	\$1000	
2:	1000	B6 0800	ldaa	\$800	
3:	1003	BB 0801	adda	\$801	
4:	1006	BB 0802	adda	\$802	
6:	1009	7A 0900	staa	\$900	
			end		

- □ The programmer must be very familiar with the hardware organization of the microcontroller.
- □ It's difficult to understand for anyone other than the author;
- □ Hard for large projects: Work on a very low level.

High-level languages

- Syntax of a high-level language is similar to English, (C/C++,Java,Fortran,PASCAL...)
- A translator is required to translate the program written in a
 - high-level language -- done by a compiler
- High-level languages allow the user to work on the program logic at higher level.
 - Source Code: A program written in assembly or high-level languageObject Code: The output of an assembler or compiler

Registers

- A register is a storage location in the CPU. It is used to hold data or a memory address during the execution of an instruction.
- □ The number of registers varies from computer to computer.
- □ 68HCS12 registers:
 - CPU registers: solely perform general-purpose operations (arithmetic,logic,flow control).
 - □ Do not occupy memory space.
 - I/O registers: mainly used to configure the peripheral functions, to hold data transferred in/out of the peripheral subsystem, and to record the status of I/O operations.
 - □ Data, data direction, control, status registers.
 - □ Treated as memory locations.



Figure 1.2 HCS12 CPU registers.

Data Types

- □ 68HCS12 supports:
 - Bits
 - 5-bit signed integers
 - 8-bit signed and unsigned integers
 - 9-bit signed integers
 - 8-bit, 2-digit BCD numbers
 - 16-bit signed and unsigned integers
 - 16-bit effective address
 - 32-bit signed and unsigned integers
- □ A multi-byte integer is stored in memory from most significant to least significant bytes starting from low to higher address.
- □ A number can be represented in binary,octal,decimal, or hex format.

b a s e	prefix	exam ple
binary octal decimal hexadecimal (shorthand hex)	% @ \$	% 1 0 0 0 1 0 1 0 @ 1 2 3 4 5 6 7 1 2 3 4 5 6 7 8 \$ 5 6 7 8

Table 1.1 Prefixes for number bases

Memory size

- □ Size of memory is measured in bytes.
- \square 1 byte=8 bits
- □ Nibble: 4-bit
- □ Word: 16 bits or 2 bytes.
- \square 1K=2¹⁰=1024
- $\square 1M = K^{2=} 2^{20} = 1048576$
- \Box 1G= K³= 2³⁰
- MC68HCS12 has a 16-bit address bus and a 16-bit data bus.
- Some 68HC12 members (68HC912DG128) use paging techniques and incorporate hardware to support addressing a larger memory space than 64KB.

Memory Addressing

- Memory consists of a sequence of directly addressable locations.
- A location is referred to as an information unit.
- A memory location can be used to store data, instruction, and the status of peripheral devices.
- A memory location has two components: an address and its contents.

Figure 1.5 The components of a memory location



Figure 1.6 Transferring data between CPU and memory

- Data transfers between the CPU and the memory are done over the common buses: address bus and data bus.
- Notations: m[addr] represents the contents of a memory location, [reg] refers to the contents of a register.
 - For example, **[\$20]** refers to the contents of memory location at \$20.
 - **[A]** refers to the contents of accumulator A.