CS280-Lecture 5 3/13/2006



# GNU as Assembler for HC11

- Modern, "single pass" assembler
  - Converts mnemonics to machine code
  - Labels sections (e.g., code)
  - Labels entry points (e.g., function)
  - Handles forward and backward references
  - Can choose smallest jump instruction (jmp/bra, etc)
  - Can generate a listing and map
  - Reports any errors
  - Generates ELF (Executable and Linking Format)
  - In particular, a "relocatable file"
    Code and data sections located later by linker
- Used automatically by gcc (C++ compiler)



# Assembler labels

- Labels are not limited in length
- Regular labels must be unique in source
  - Must be unique among all files if global
- Begin with A-Z a-z . \$ \_
- Legal subsequent characters (0-9 A-Z a-
- The colon (:) terminates a label

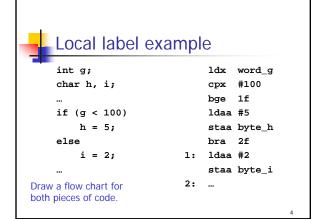


#### Local assembler lables

- Local labels are 1 to maxPositiveInteger
- Scope
  - Back to same label or beginning of file
  - Forward to same label or end of file
- Must specify whether you want the
  - next (f forward) matching label
  - or the previous (b backward) one
  - Otherwise, the number is interpreted as a relative addressing mode offset!

© Eric A. Durant, PhD

CS280-Lecture 5 3/13/2006





### **Numbers**

- Radix prefixes
  - 0b, 0B binary (0b11000001)
  - 0 octal (0377)
  - 0x, 0X hexadecimal (0xFF, 0x5a)
- All values decimal unless specified

5



### **Sections**

- section name
  - Standard ones are:
  - .bss uninitialized, mutable data ("block started by symbol")
  - .data initialized, mutable data
  - .rodata initialized, read-only data
  - .text code

6

© Eric A. Durant, PhD

CS280-Lecture 5 3/13/2006



# Data storage (1/2)

	Byte	Word
Reserve but don't initialize	.space N .rmb N	.ds N
Reserve and initialize	.byte 0xAB, .fcb 0x3F,	.word 0x1234,

7



# Data storage (2/2)

- Strings
  - .ascii "string" stores ASCII characters
  - .asciz "string" zero byte inserted at end (standard C string termination)
- Data in .bss will not be initialized (*i.e.*, not included in download image), even if you use .byte, .word, etc.

8



# Other Assembler Directives

- .global sym1 define global symbolsFor linking multiple modules
- .include "name" include another source file

CS280-Lecture 5 3/13/2006



# Invoking the assembler

- as
  - o *output.o* specify output file
  - *input.s* specify input file(s)
  - -a output a listing (only relative addresses)
  - -v version information
- Documentation:
  - GNU 68HC11 | Assembler

10



# Invoking the linker

- Id
  - scriptname.x usually .x (e.g., fox11w.x), linker configuration, especially memory layout – where to put the sections in memory (RAM, ROM, data, code, etc.)
  - o output.elf specify output file
  - input.o specify input files (extension doesn't matter, internal data does)
  - r relocatable link (incremental)
  - -v -V version information (and more)
  - -relax replace extended addressing mode with relative or direct mode when possible

11



#### Simulator

- Wookie, latest version 1.71
- Simulator
- Can view memory, registers
- Emulates a matrix keypad (but not Fox11's)
- Can see port status (but not Fox11 special norts)
- Can single step even through interrupts
- Missing: A/D, handshaking

12

© Eric A. Durant, PhD