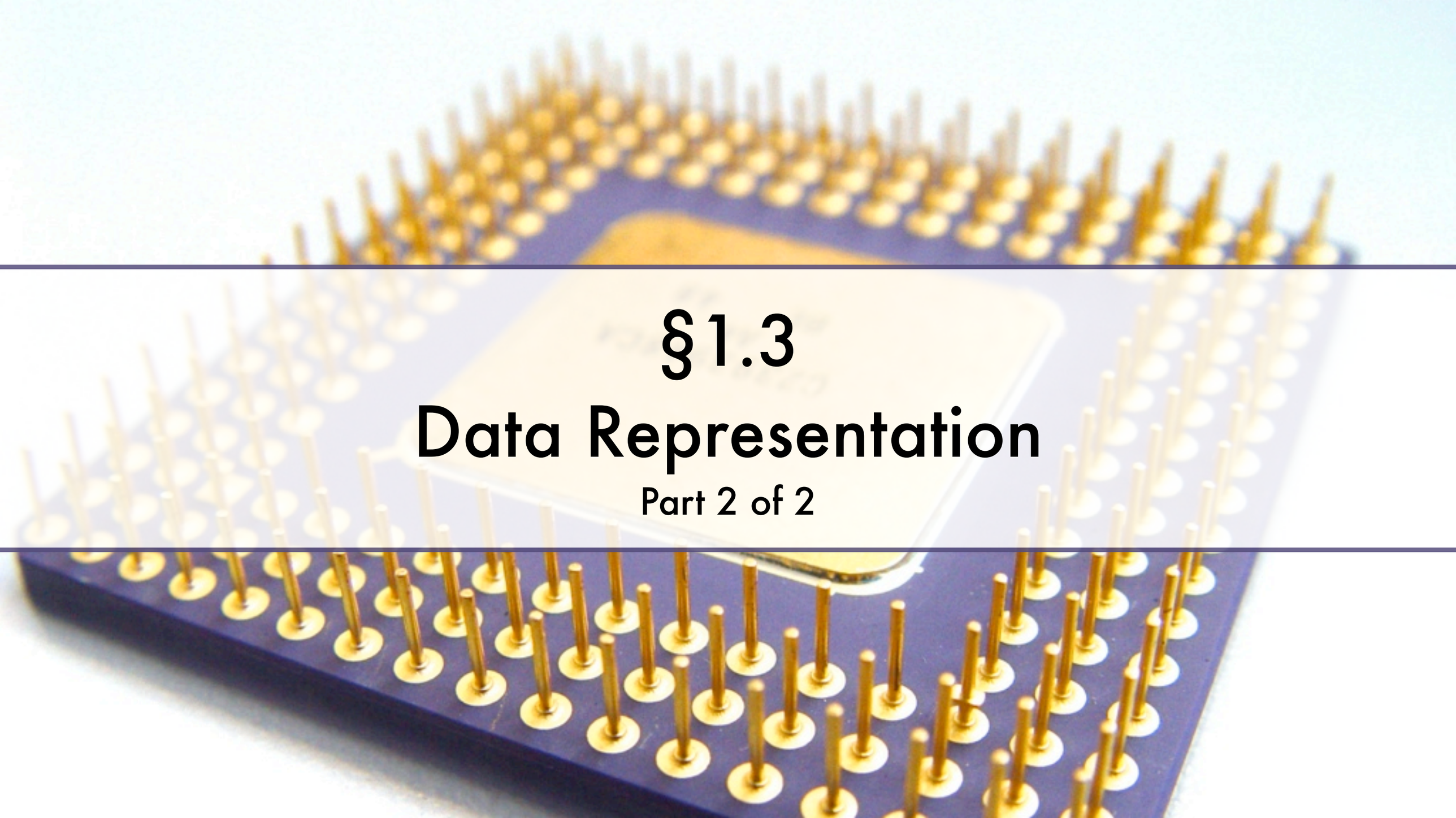
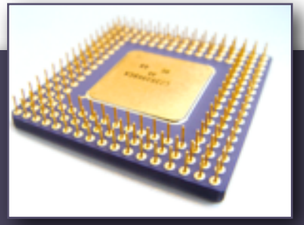


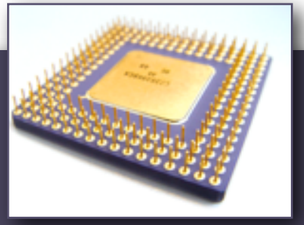
§1.3
Data Representation
Part 2 of 2

Topics Covered in Whiteboard Notes:



- Maximum and minimum representable values (finish from last time)
- Conversion of integers between representations

Questions Answered on Whiteboard:



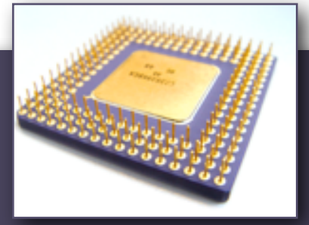
1. What is the range of values that can be represented by a 7-bit unsigned integer?

2. What is the decimal representation of the following 8-bit signed (two's complement) integers?

   (a) 3Ch

   (b) 9Ah

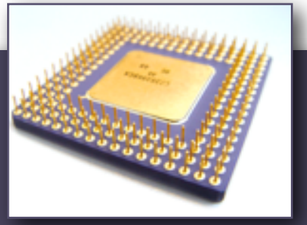
Character Strings



- Since computers only store binary data, how can they store character strings?
- A *character set* maps characters to integers
  - ASCII Character Set *(See Activity 2 & inside back cover of textbook)*
    - "American Standard Code for Information Interchange"
    - Assigns a 7-bit integer (0–127) to each character
    - Characters 0–31 are *control characters* (backspace, newline, etc.)
  - ANSI Character Set†
    - 8-bit character set used by Windows 95/98/ME
    - Characters 0–127 correspond to ASCII

† From http://msdn.microsoft.com/en-us/goglobal/bb964658.aspx#a: "The term "ANSI" ... is a historical reference, but is nowadays a misnomer that continues to persist in the Windows community. The source of this comes from the fact that the Windows code page 1252 was originally based on an ANSI draft..." See also http://ascii-table.com/ansi-codes.php#note

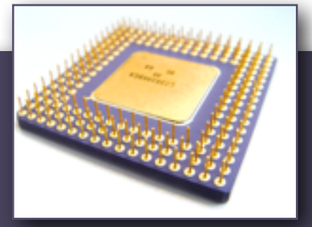
Unicode



- The Unicode standard defines characters for *all* major languages (unlike ASCII)
  - Unicode 7.0 (June 2014 version) has character codes for 113,021 characters
- Used by modern versions of Windows, Java, etc.
- Unicode characters are *encoded* in various formats:
  - **UTF-8** (Unicode Transmission Format, 8-bit)
    - Backward-compatible with ASCII, i.e.,
      every ASCII string is also a valid UTF-8 representation of the same string
    - Non-ASCII characters like ∞ and 何 are encoded using 2-, 3-, or 4-byte sequences
    - Dominant character encoding for Web pages
  - Other encodings: **UTF-16**, **UTF-32**

For more information, see http://www.unicode.org and http://www.unicode.org/faq/utf_bom.html

Activity 2



Activity 2, Question 1

- What is the ASCII representation of the 5-character string "`Hi!  3`"

Space

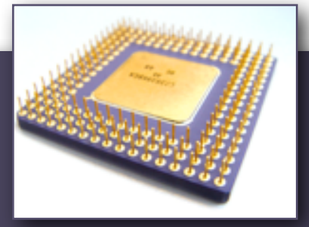
Activity 2, Question 2

- Can "It costs €300" be represented in ASCII?

Activity 2, Question 3

- Can the 3-character string "今日は" be represented in ASCII? Windows ANSI? UTF-8? UTF-16?

Line Endings

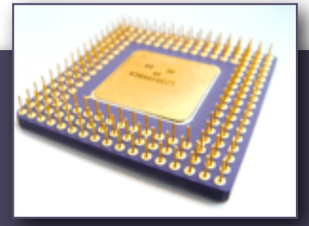


- How are line endings represented?
  - On Windows:
    - Represented by a two-character sequence:
      carriage return (ASCII 13), then linefeed (ASCII 10)
    - Programmers often call this sequence “CRLF”
    - You will see this in Lab 1:

```
message BYTE "Hello", 0dh, 0ah, 0
```

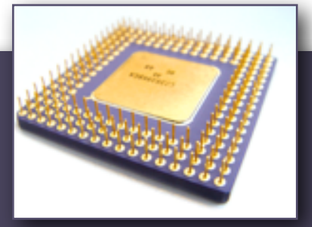
  - On Linux/Unix/Mac OS X:
    - Represented by linefeed only (ASCII 10)

Null-terminated Strings



- You will see this in Lab 1:
  `message BYTE "Hello", 0dh, 0ah, 0`
  - In Java, the closest equivalent is `String message = "Hello\r\n";`
- This defines 8 bytes of memory (what are they?)
- It is common to use a NUL character (ASCII 0) to indicate the end of a string
  - The reason why will become clear later in the course
  - This is also common in the C programming language

Some Terminology

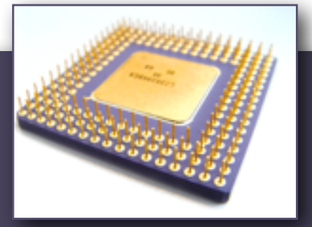


- The integer value 123 is different from the string “123”
  - Like in Java: `int n = 123;  String n = "123";`
- The ASCII character “0” is represented by the byte value 48, “1” is 49, and “9” is 57
- $123 = 01111011_2$, so the **integer value** 123 is represented as the single byte 01111011
- The **ASCII digit string** “123” is the 3-byte sequence 49 50 51

Activity 2, Question 4

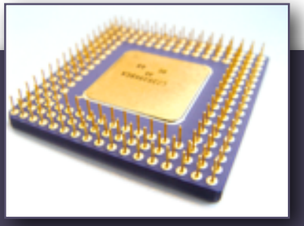
- What is the binary representation of the value 5? If that binary number is represented as a 3-character ASCII digit string, what are the bytes in that string?

Installing Visual Studio



- Monday: Lab 1 (Getting Started with Assembly Language in Visual Studio)
- Machines in the labs (Shelby 2119, 2122) have Visual Studio 2010 installed
- To install Visual Studio on your own machine...
  - Auburn students can obtain Visual Studio 2010, 2012, 2013 for free through Microsoft DreamSpark
    - If you do not have a DreamSpark login, e-mail Ms. Kelly Price <pricekg@auburn.edu> to request one
  - Follow the instructions for "Getting Started with MASM and Visual Studio" at www.asmirvine.com
  - Labs are written with instructions for VS2010 but should work OK with VS2012/13
- Note: When installing, the Microsoft Macro Assembler (MASM) is installed as part of Visual C++

Homework



- **On Monday, class will meet in the computer labs (2219 and 2122 Shelby)**
  - Go to either room—wherever you can find a seat (each seats 26–28 people)
- Skim Section 1.3 to review material from this and the previous lecture
- **Read Section 2.1** from the **6th edition** of the textbook (pp. 29–35)
  - PDF in Canvas under “Readings”
  - 6th edition contains some material omitted from the newer 7th edition
- Be prepared to verbally answer review questions
  1, 2, 4, 5, 9, and 11 from Section 2.1.5