* 1 cycle = 0.1 μs
* 100 KHz and 200 cycles takes 2000 μs
* Why C?
  + Designed to expose machine details for efficiency
  + Borrows features from high level languages
  + Easier to manage large embedded systems
* Why Assembly?
  + High speed, low code size, low energy
  + However, low programmer productivity
* Data representation
  + Bit (1 for true, 0 for false)
  + Nibble – 4 bits
  + Byte – 8 bits
  + Word – 16 bits
  + Double word – 32 bits
* C syntax
  + All these lines are equivalent; Computers understand binary
    - Char x = 2 + 1;
    - Char x = 0b10 + 1;
    - Char x = 0x2 + 1;
    - Char x = 0x02 + 0x01;
  + Use these for bases
    - 0x – Hexadecimal
    - 0b – Binary
    - 0 – Octal (e.g. 020 is not 20)
* Processor Architecture
  + Von Neumann Architecture
    - Single data area that stores both program and data memory
  + Harvard Architecture
    - Separate memories, one for data and one for program memory
  + RISC Architecture (Reduced Instruction Set Computing)
    - Reasoning: reduced number of instructions will increase simplicity and lead to fast processors, fewer transistors and less power