



Pros and Cons of Popular Microcontrollers for Robotics

Portland Area Robotics Society
September 6, 2003 Meeting



Popular Microcontrollers and Robotics

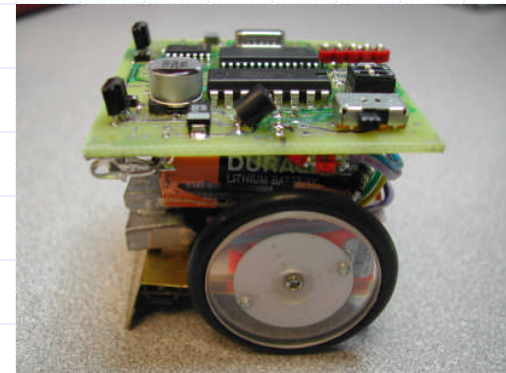
Agenda:

- ◆ Atmel AVR: Monty Goodson
- ◆ Motorola 68hc1x: Warren Leach
- ◆ Microchip PIC: Pete Skeggs
- ◆ Other Micros to Avoid or Consider
- ◆ Discussion

PIC Micros in Robots:

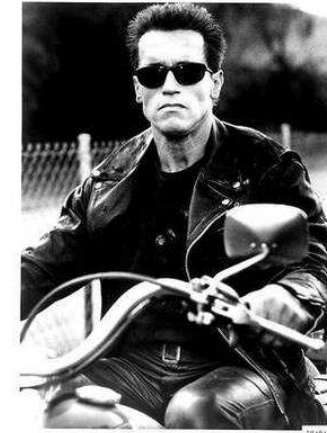
Pros and Cons

Portland Area Robotics Society Meeting
9/6/2003



PIC Micro Pros:

- ◆ Very Popular (like Ahnold)
- ◆ Lots of resources available to learn from:
 - Easy PIC'n books
 - PARTS members
 - PICList.com!
 - Google...
- ◆ Simple But Powerful Architecture:
 - Self-contained designs possible with very low parts count (e.g., an 8 pin 12F675 needs just power and ground, leaving 6 usable pins!)
 - Lots of robot-friendly hardware units included:
 - ◆ A/D converters, RS-232, I²C, SPI, PWM, timers, digital I/O, interrupts



PIC Micro Pros: (cont)

◆ Simple But Powerful Architecture:

- Flash memory for easy program updates (don't need to erase and burn an EPROM each compile -> test -> debug cycle)
- EEPROM available for calibration constants or last-state retention (the wheels are straight)
- Pretty fast execution speed; fast enough for all but the most complex hobbyist robots
- Many different versions available with lots of different combinations of features but the same core instruction set, so it is easy to find one that fits your robot's design perfectly

◆ Free Integrated Development Environment

◆ Lots of Good Commercial Compilers:

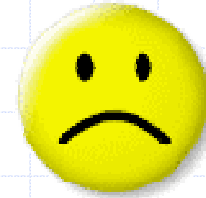
- PIC Basic Pro, CCS C, FlashBasic

PIC Micro Pros: (cont)

- ◆ Some Free / Low Cost Compilers:
 - JAL, PICAnt(!), Hi-Tech's CC5X
GNU PIC (SDCC) in development
- ◆ Support Libraries in Some Compilers Make Use of Hardware Units Painless
- ◆ Lots of Kits and Development Boards available:
 - MarkIII
 - Oricom Tech
 - Microchip's Eval Boards

PIC Micro Cons:

- ◆ Monty and Warren are not behind it
- ◆ GCC not yet ported; maybe too hard
- ◆ Assembler instructions are bizarre!
- ◆ Not enough RAM for some applications
- ◆ No built-in boot loader (but they are possible to add)
- ◆ Commercial compilers can be expensive
- ◆ Design is not quite keeping pace with competition
- ◆ Math support is poor (18x series does have hardware multiply, though)
- ◆ Hardware units can be hard to use without prewritten libraries
- ◆ Reset vector not easily relocatable, so bootloader usage can be hard

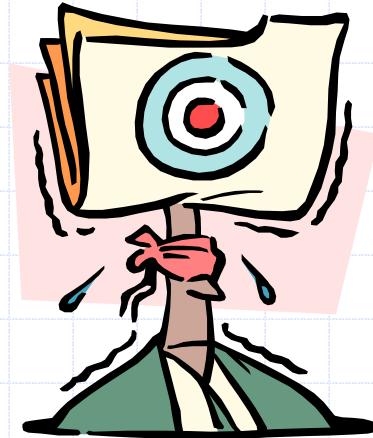


Special Bonus Section!

◆ Micros to AVOID!

- Intel 8031, 8052, 8085
- Intel 80x86 or AMD V2x / V3x (JK Micro Flashlight at Fry's)
- Motorola 68000, PowerPC
- Zx80 (Rabbit 2000 at Fry's)

- ◆ These (and many others in this class; generally general purpose microprocessors) are missing the helpful hardware units needed for robotics, require too many external components to do anything, and have little or no internal RAM or ROM
- ◆ PCs (notebooks) can be used, but require either custom built or expensive off-the-shelf expansion boards to do useful things (motor control, sensor data acquisition)



Special Bonus Section Part 2!

◆ Integrated Micros to CONSIDER!

- OOPIC-1 and OOPIC-2
- Acroname Brainstem
- Parallax Basic Stamp 2
- Kronos Robotics' DIOS



(all of these use PICs with custom language interpreters that read from external serial EEPROM)

◆ Complete Solutions at Low Cost

◆ But they can limit you as your skills and goals grow