AVR – PIC μC

Competitive Analysis

AVR Flash Microcontroller Team

ATMEL Corporation

Introduction:

In the 20th Century, Microchip introduced a single accumulator 8-bit microcontroller Architecture called the PIC. The principal selling features were small outline packages and OTP capability. There were a few large die Flash devices in their portfolio. Microchip tried, but never mastered integrating cost effective Flash technology with the PIC core so they focused on selling OTP.

Because the PIC Architecture was designed over 20 years ago when logic gates were very expensive, certain limitations were built in the PIC to reduce die size. The limitation of the instruction set resulted in an increase of word length from 12 to 14 and ultimately 16 bits to satisfy the need for increased capability and performance. This means the PIC family is not compatible across the entire line, and code must be rewritten when attempting to migrate up or downwards.

For many designs, PIC had adequate performance and features. We have AVR devices that have a similar feature set, can be factory or Disti programmed to look like an OTP or ROM and have rock bottom prices. Additionally the AVR has additional features like on chip EEPROM, BOD, Flash and In-System- Programming. The following competitive analysis is designed to help you identify the lowest cost AVR to sell against a comparable PIC device. It can also help identify AVR's which have additional features integrated on chip to reduce system cost, board space and improve reliability.

Advantages of the AVR over the PIC-Architectures

Feature	AVR SOLUTION	PIC SOLUTION
Clock Frequency	AVR executes instructions in a single clock cycle. It will run up to **16Mips** at a clock speed of 16MHz.	PIC executes instructions in 4 clock cycles. This means 40MHz yields about **10Mips** performance. The **disadvantage** of this architecture is that 4x higher frequency creates more EMI issues. The increase in power consumption is linear with frequency. The AVR will consume less power to perform the same task.
Memory Type	AVR flash parts can be pre-programmed to behave like an OTP or ROM with the added benefit of **Re-programmability** in case of code change.	PIC offers a 2-step solution with windowed or large die size flash part for prototyping and a re-qualification of an OTP later in the production cycle.
Memory Organization	**Linear Memory Space** is easier to program.	Non-linear address space. Needs up to 3% code overhead for Bank-Handling on 1-2KB memory parts. It is more complicated to write software.
Instruction Set	Up to 118 instructions – 16 bit wide. The rich instruction set improves code density and leads to and faster execution. PIC's small instruction set needs multiple instructions to complete the many task executed by the AVR in one instruction.	**Different instruction length** requires code rewrite to use other PIC family members. – 12bit word PIC12C5X / PIC16C5X – 14bit word PIC12 / PIC16 parts – 16bit word PIC18 parts Word widths of less than 16 bits store and handle constants and text strings inefficiently because each word can only contain one byte.
Code Compatibility	**Code is fully compatible** across the entire AVR Family. Reassembling or recompiling is all that is needed.	Architectural differences and word length require code to be rewritten when migrating from 12, 14 and 16 bit architectures. Different bank sizes require change in program memory. Different interrupt handling and conversion overhead may be several man month efforts.

Feature	AVR SOLUTION	PIC SOLUTION
Interrupt Handling	**Individual Interrupt Vectors** for all interrupt sources. This allows the AVR to have a constant interrupt latency as short as 4 external clock cycles. This is **minimum 4 times faster** than any of the PIC devices.	Several interrupt sources use same interrupt vector. Response time and code size increase to determine which interrupt occurred. The interrupt latency grows symmetrically with the number of interrupts enabled.
Code Size	Optimized architecture yields a **2.5 times smaller code than PIC**. 32 general-purpose registers can be directly accessed by ALU eliminating 66% of move instructions used by single accumulator architectures.	PIC has larger code size due to a less efficient architecture and limited instruction set. Two operand instructions loading to and saving from Accumulator have an overhead of up to 8 extra cycles per operation. There is no conditional branch instruction available on PIC12/16, which require 8 extra cycles per branch.
Analog Comparator	Available on almost all AVR parts. Can be used to **implement low cost AD-channel** with only one external resistor and capacitor.	No Analog Comparator available on 8-pin parts. Some of the 18-pin parts of the PIC16CXXX series have up to two Comparators.
Power Consumption	AVR consumes 4x less power when executing a specific task because of 10x higher throughput at same frequency. AVR in the Sleep Mode consumes less than 1µA while the PIC is still active.	Both devices use a similar amount of power in the active mode.

Advantages of the PIC over the AVR

FEATURE	PIC	AVR
Power Consumption	PIC devices use 25µA power when running at 32KHz at 3V.	AVR consumes 200µA at 3V while executing 10x more operations. AVR is more efficient than PIC when running at a higher frequency and utilizing power down modes.
Reset Sensitivity	PIC has synchronous reset making them less vulnerable to noise on reset.	AVR can detect a reset even when the clock is stopped. This is not possible with PIC.