MICROCONTROLLER VARIENT

There are five major 8 bit microcontrollers

- Free scale semiconductors (formerly Motorola) 88HC08, 68HC11)
- Intel 8051
- Atmel's AVR
- Zilogs Z80
- PIC from microchip technology

❑Above each of the microcontroller family has unique instruction set and register set

❑So Program code of one microcontroller does not run on other

❑ There are 16 bit and 32 bit microcontroller also.

So what are the criteria required to choose the Microcontroller

❖1.Meeting the computational need at task at hand efficialty and cost effectively

❖2. Availability of software and hardware developmewnt tools

❖3 Wide availability of the microcontrollers

I : Meeting o the task at hand efficiently and cost effectively : Out of 8,16,32 bit we should choose properly

In addition to above following criteria

1. **Speed**
2. **Packaging (DIP duel in line/QFP Quad Flat Package (This is important In terms of space, assembling and prototyping the end product**
3. **Power consumption**
4. **The amount of RAM and ROM**
5. **The number of IO pins and the timers on the chip**
6. **Ease of upgrade to high performance and low power consumption versions**
7. **Cost per unit**

II Easiness of development of the product

Whether the tools are available or not such as assemble , compilers, emulators , technical support etc

III Readily availability of the microcontroller

eg, 8051 is readily available in various version

The emerging field of mechatronics need the specific controller for specific application.

AVR Microcontroller

AVR may stands in the name of the designer Alf-Egil Bogen and Vegard Wollan RISC, or Advance Virtual RISC , or the company simply give the name AVR. No as such standard meaning of AVR!

There are may kind of AVR

All AVR are 8 bit microcontroller , ie CPU work on only 8 bit data

All these AVR are having different families. There is no 100% software compatibility among the AVR families . Program one of the family has tobe re-compiled to run on other family.

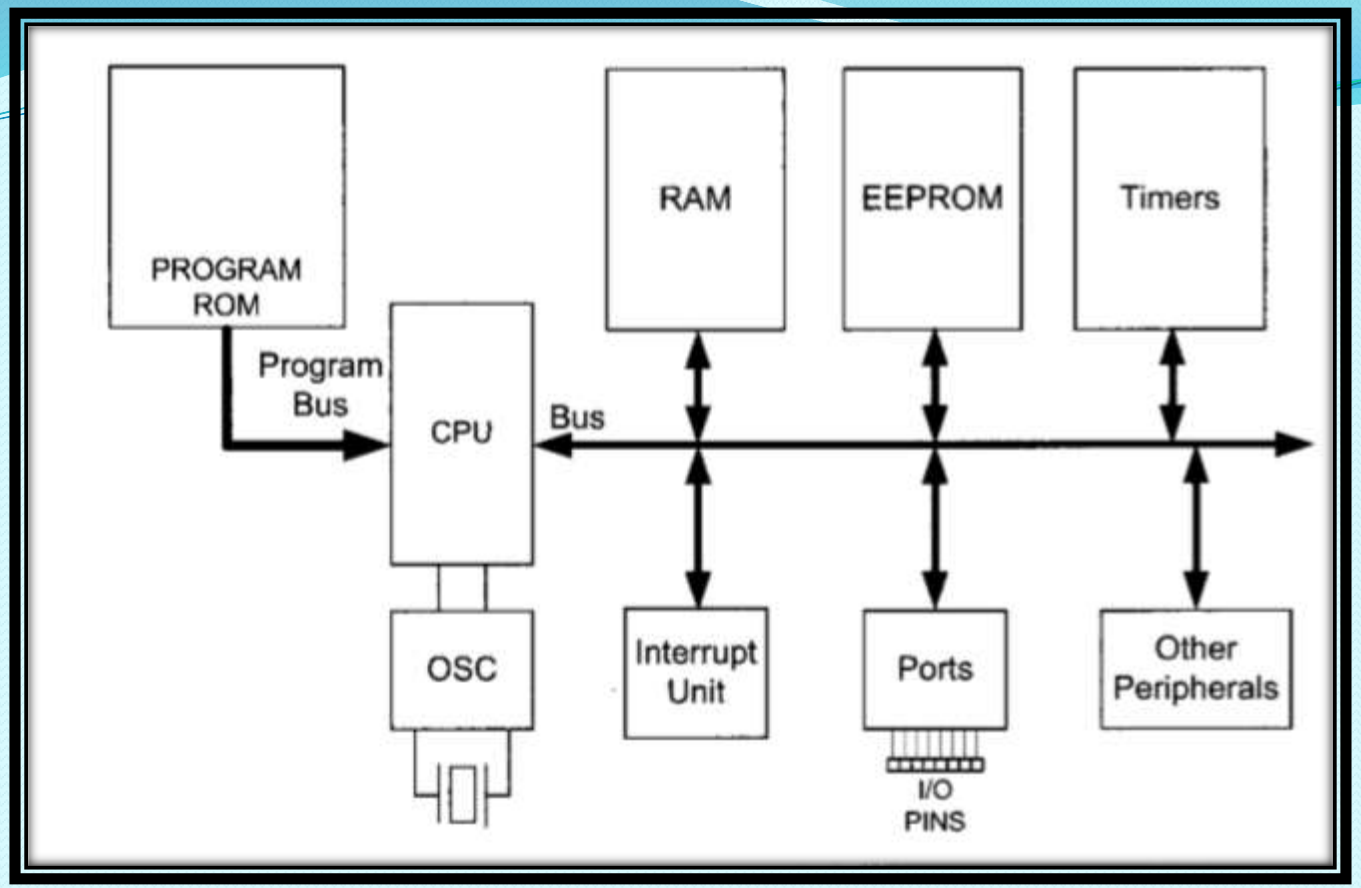
AVRs are generally classified in to four broad groups

Mega ,Tiny , Special Perpose, Classic.

Mega Family are widely used. Once the Mega family is mastered the other families could be understood with ease.

Features of AVR (only AVR32 IS 32 BIT)

- Its 8 Bit
- RISC Harward Acrcitecture
- Consists of On chip PROM, Data RAM, Data EEPROM Program and data memory
- Timers , I/O Ports
- Most AVR have on-chip ADC, PWM, Serial interface such as CAN,USB,USART, I2C



Simplified Block diagram of AVR

Other peripherals such as ADC,PWN, USART, CAN, I2C, EEPROMs Are additional devices on chip

AVR *CLASSIC* FAMILY

Table 1-2: Some Members of the Classic Family

Part Num.	Code ROM	Data RAM	Data EEPROM	I/O pins	ADC	Timers	Pin numbers & Package
AT90S2313	2K	128	128	15	0	2	SOIC20, PDIP20
AT90S2323	2K	128	128	3	0	1	SOIC8, PDIP8
AT90S4433	4K	128	256	20	6	2	TQFP32, PDIP28

Notes:
1. All ROM, RAM, and EEPROM memories are in bytes.
2. Data RAM (general-purpose RAM) is the amount of RAM available for data manipulation (scratch pad) in addition to the register space.

Table 1-3: Some Members of the ATmega Family

Part Num.	Code ROM	Data RAM	Data EEPROM	I/O pins	ADC	Timers	Pin numbers & Package
ATmega8	8K	1K	0.5K	23	8	3	TQFP32, PDIP28
ATmega16	16K	1K	0.5K	32	8	3	TQFP44, PDIP40
ATmega32	32K	2K	1K	32	8	3	TQFP44, PDIP40
ATmega64	64K	4K	2K	54	8	4	TQFP64, MLF64
ATmega1280	128K	8K	4K	86	16	6	TQFP100, CBGA

Notes:
1. All ROM, RAM, and EEPROM memories are in bytes.
2. Data RAM (general-purpose RAM) is the amount of RAM available for data manipulation (scratch pad) in addition to the register space.
3. All the above chips have USART for serial data transfer.

AVR MEGA FAMILY

Tiny AVR family

Program memory 1 to 8 k
Limited Peripheral set
Limited instruction set

Table 1-4: Some Members of the Tiny Family

Part Num.	Code ROM	Data RAM	Data EEPROM	I/O pins	ADC	Timers	Pin numbers & Package
ATtiny13	1K	64	64	6	4	1	SOIC8, PDIP8
ATtiny25	2K	128	128	6	4	2	SOIC8, PDIP8
ATtiny44	4K	256	256	12	8	2	SOIC14, PDIP14
ATtiny84	8K	512	512	12	8	2	SOIC14, PDIP14

Special purpose AVR

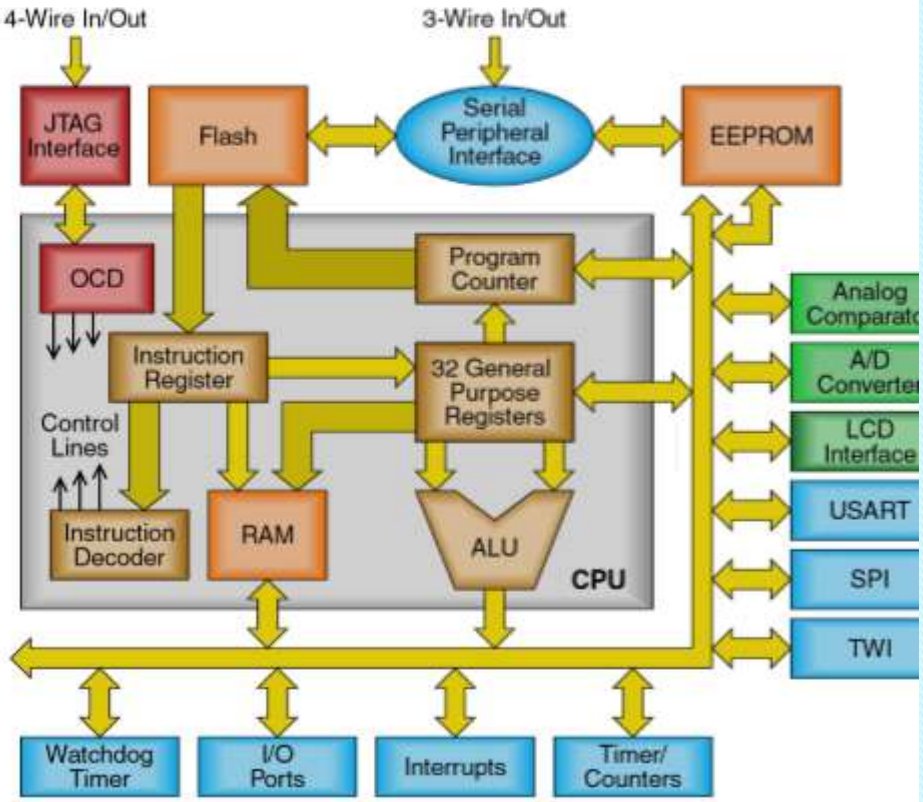
USB controller
CAN controller
PWM
LCD controller

Table 1-5: Some Members of the Special Purpose Family

Part Num.	Code ROM	Data RAM	Data EEPROM	Max I/O pins	Special Capabilities	Timers	Pin numbers & Package
AT90CAN128	128K	4K	4K	53	CAN	4	LQFP64
AT90USB1287	128K	8K	4K	48	USB Host	4	TQFP64
AT90PWM216	16K	1K	0.5K	19	Advanced PWM	2	SOIC24
ATmega169	16K	1K	0.5K	54	LCD	3	TQFP64, MLF64

AVR ARCHITECTURE (RISC)

- RISC architecture with CISC instruction set
  - Powerful instruction set for C and Assembly
- Scalable
  - Same powerful AVR core in all devices
- Single cycle execution
  - One instruction per external clock
  - Low power consumption
- 32 Working Registers
  - All Directly connected to ALU!
- Very efficient core
  - 20 MIPS @ 20MHz
- High System Level Integration
  - Lowest total system cost



PIC (usually pronounced as "pick") is a family of microcontrollers made by Microchip Technology

- Early models of PIC had read-only memory (ROM) or field programmable EPROM for program storage, some with provision for erasing memory.
- All current models use flash memory for program storage, Newer models allow the PIC to reprogram itself.
- Program memory and data memory are separated.
- Data memory is 8-bit, 16-bit, and, in latest models, 32-bit wide.
- Program instructions vary in bit-count by family of PIC, and may be 12, 14, 16, or 24 bits long.
- The instruction set also varies by model, with more powerful chips adding instructions for digital signal processing functions.

•The hardware capabilities of PIC devices range from 6-pin SMD 8pin DIP chips up to 144-pin SMD chips, with discrete I/O pins
• ADC and DAC modules
•Communications ports such as UART, I2C, CAN, and even USB.
•Low-power and high-speed variations exist for many types.

•The manufacturer supplies computer software for development known as MPLAB X, assemblers and C/C++ compilers, and programmer/debugger hardware under the MPLAB and PICKit series.

PIC devices are popular with both industrial developers and hobbyists due to their *low cost, wide availability, large user base*, *extensive collection of application notes, availability of low cost or free development tools, serial programming, and re-programmable flash-memory capability*

PIC and AVR will be compared on variety of parameters. Following are some parameters

- **Frequency: Speed at which the microcontroller will operate**
- **Number of I/O pins : Required ports and pins**
- **RAM: All the variables and arrays declared(DATA) in most MCUs**
- **Flash Memory: Whatever code you write goes here after compiling**
- **Advanced Interfaces: Advanced interfaces such as USB, CAN and Ethernet.**
- **Working Voltage: Working voltage of MCU such as 5V, 3.3V or Low voltage.**

Target Connectors: The connectors for ease of circuit design and size

The most of the parameters are similar in both PIC and AVR but there are some parameters that surely differ when compared given in next slide ....

Working Voltage

With more battery operated products, the PIC and AVR have managed to improve for the low-voltage operations. **AVR are better known for low-voltage operation than the older PIC series** such as PIC16F.

AVR has improved and launched the latest P (pico-power) variants such as ATmega328P which are extremely low-power. Also the current ATtiny1634 has improved and comes with sleep modes to reduce power consumption when brownout is used which is very useful in battery powered devices.

AVR were focused on low-voltage previously but PIC now has been transformed for the low-voltage operation and has launched some products based on picPower.

TARGET CONNECTORS

Target connectors are very important when it comes to design and development.

AVR has defined 6 and 10-way ISP interfaces, which makes it easy to use

Whereas PIC doesn’t have it, so PIC programmers comes with flying leads or RJ11 sockets which are difficult to fit in the circuit.

Conclusion is that the AVR has made it simple in terms of circuit design and development with the target connectors whereas PIC still needs to rectify this.

Advanced Interfaces

In terms of advanced interfaces, PIC is surely the option as it has got their act with advanced features such as USB, CAN and Ethernet which is not the case in AVR. However one can use external chips, such as FTDI USB to serial chips, Microchip Ethernet controllers or Philips CAN chips.

Conclusion is that the **PIC has surely got the advanced interfaces** than AVR.

WHY AVR?

	8051	AVR	PIC
Speed	Slow	Fast	Moderate
Memory	Small	Large	Large
Architecture	CISC	RISC	RISC
ADC	Not present	Inbuilt	Inbuilt
Timers	Inbuilt	Inbuilt	Inbuilt
PWMs	Not present	Inbuilt	Inbuilt

Comparing Microcontrollers

	8051	AT 89S52	68HC11	Atmega 328	PIC 16F84A	PIC 18F452	ARM7-TDMI-S LPC2148	ARM Cortex-M3 LPC1768
CPU	8-bit	8-bit	8-bit	8-bit	8-bit	8-bit	16-bit/ 32-bit	16-bit/ 32-bit
Data Bus / Instr bus	8-bit	8-bit	8-bit	8-bit	8-bit / 14-bit	8-bit / 16-bit	32-bit / 32-bit	32-bit / 32-bit
Address bus	16-bit	16-bit	16-bit	16-bit	14-bit	16-bit	32-bit	32-bit
Clock Speed	12 MHz	0-33 MHz	0-4 MHz	20 MHz	20 MHz	0-40 MHz	10-25 MHz	0-100 MHz
Memory								
Program Memory	Int - 4 KB Extl - 64 KB	8 KB (flash)	8 KB	32 KB	14-bits 1 KB	32 KB	512 kB	512 kB
Data Memory	128 B (64 KB ext)	256 B	256 B	2 KB	RAM - 68B EEPROM-64B	RAM-1536 B EEPROM-256B	32 KB + 8 KB	Up to 64 KB
Interrupts	5	8	18	2	4	18	22	
DMA Channels	No	No	No	No	No	No	One (for USB)	8 Channel Gen purpose
I/O ports	4 x 8-bit 32 I/O lines	4 x 8-bit 32 I/O	40 I/O lines	23/28/32	13	4 x 8-bit 1 x 3-bit	45 pins	70 / 52