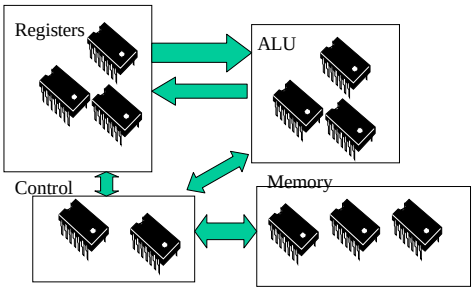
Processor Architecture Components

- Registers : used for storing data values and addresses
- Execution units: perform computations (arithmetic, logic) on data
  - ALU: Arithmetic Logic Unit
- Control: logic for fetching, executing instructions
- Memory: stores instruction and data
- Input/Output: external interface with computer system

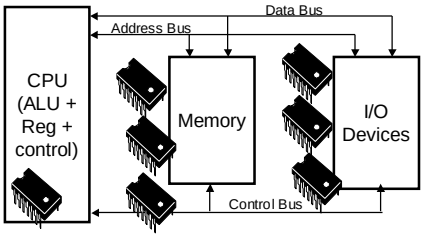
Processor Integration

Early computers had many separate chips for the different portions of a computer system



Microprocessors

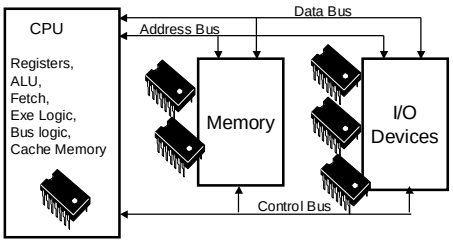
First *micro*processors placed control, registers, Arithmetic logic unit in one integrated circuit (one chip).



CPU – Central Processing Unit

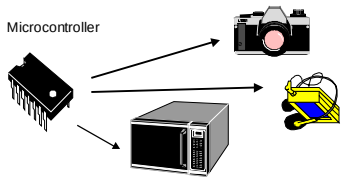
Modern µProcessors

Modern *micro*processors (general purpose µProcessors) also integrate memory onchip for faster access. External memory and I/O components still required. Memory integrated on the microprocessor is called *cache* memory.



Microcontrollers

Microcontrollers integrate all of the components (control, memory, I/O) of a computer system into one integrated circuit. Microcontrollers are intended to be single chip solutions for systems requiring low to moderate processing power.



Intel x86 Processors

- In this course will study the Intel x86 general purpose microprocessor family
  - Instruction set architecture
  - Assembly language programming
  - Hardware features of different x86 implementations
  - Other features of microprocessor systems

ISA vs Implementation

- Instruction Set Architecture (ISA) : the definition of the registers and instructions that define the programmer's view of a processor (is a text document)
- Can have different implementations of the same ISA!
  - Intel and AMD processors both execute the x86 ISA, but internally are very different!!
  - Intel has several different implementations of the x86 ISA!
- New versions of an ISA extend the previous version by adding instructions, registers – but never invalidate the old ISA!!!!!!!

Intel x86 Microprocessors

CPU Name	Year Intro.	Int. CPU Clock	# Trans.	Data Pins	Addr Pins
8080	1974	2-3 MHz	4500	8	16
8086	1978	5-10 MHz	29000	16	20
80286	1982	6-16 MHz	130000	16	24
80386	1985	16-33 MHz	275000	32	32
80486	1989	25-50 MHz	1.2M	32	32
Pentium	1994	60-200 MHz	3.1M	64	32
Pentium Pro	1995	150-200 MHz	5.5M	64	36
Pentium MMX	1997	133-266 MHz		64	32
Pentium II	1998	233-500 MHz	7.5M	64	
Celeron	1998	266-500 MHz	7.5M	64	
Pentium III	1999	450-600 MHz		64	

Intel x86 Microprocessors

8086 - 20 bit Addr. Bus - 1MB of Memory

80286 - 24 Addr. Bus - Added Prot. Mode

80386 - 32 bit regs/busses - Virtual 86 Mode

80486 - RISC Core - L1 Cache - FPU

Pentium - Superscalar - Dual Pipeline - Split L1 Cache

Pentium Pro - L2 Cache - Br. Pred. - Spec. Exec.

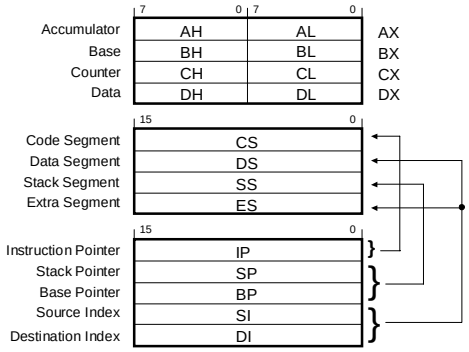
Pentium MMX - 57 Instructions - Integrated DSP (MMX)

Pentium II - 100 MHz Bus - L2 Cache - MMX

Celeron - 66 MHz Bus - True L2 Cache Integration

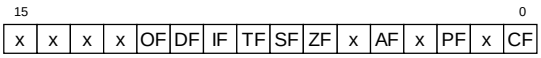
Pentium III - 100 MHz Bus - 70 Instr. Streaming SIMD Ext.

8086/8088 Register File



8086/8088 Register File (cont)

Flags Register



Status and Control Bits Maintained in Flags Register

- Generally Set and Tested Individually
- 9 1-bit flags in 8086; 7 are unused

Status Flags

Indicate Current Processor Status

CF	Carry Flag	Arithmetic Carry
OF	Overflow Flag	Arithmetic Overflow
ZF	Zero Flag	Zero Result; Equal Compare
SF	Sign Flag	Negative Result; Non-Equal Compare
PF	Parity Flag	Even Number of “1” bits
AF	Auxiliary Carry	Used with BCD Arithmetic

Control Flags

Influence the 8086 During Execution Phase

DF Direction Flag Increment/Decrement

– used for “string operations”

IF Interrupt Flag Enables Interrupts

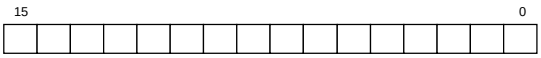
– allows “fetch-execute” to be interrupted

TF Trap Flag Allows Single-Step

– for debugging; causes interrupt after each op

8086/8088 Register File (cont)

Instruction Pointer Register



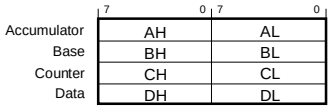
IP Contains Address of *NEXT* Instruction to be Fetched

– Automatically Incremented

– Programmer can Control with `jump` and `branch`

AX, BX, CX, DX

General Purpose Registers



- Can Be Used Separately as 1-byte Registers

  AX ← AH:AL
- Temporary Storage to Avoid Memory Access
  - Faster Execution
  - Avoids Memory Access
- Some Special uses for Certain Instructions

AX, BX, CX, DX

General Purpose Registers - Some Specialized Uses

	7 … 0	7 … 0
Accumulator	AH	AL
Base	BH	BL
Counter	CH	CL
Data	DH	DL

- AX, Accumulator
  - Main Register for Performing Arithmetic
  - mult/div must use AH, AL
  - "accumulator" Means Register with Simple ALU
- BX, Base
  - Point to Translation Table in Memory
  - Holds Memory Offsets; Function Calls
- CX, Counter
  - Index Counter for Loop Control
- DX, Data
  - After Integer Division Execution - Holds Remainder

CS, DS, ES, SS - Segment Registers

Contains "*Base Value*" for Memory Address

- CS, Code Segment
  - Used to "point" to Instructions
  - Determines a Memory Address (along with IP)
  - Segmented Address written as CS:IP
- DS, Data Segment
  - Used to "point" to Data
  - Determines Memory Address (along with other registers)
  - ES, Extra Segment allows 2 Data Address Registers
- SS, Stack Segment
  - Used to "point" to Data in Stack Structure (LIFO)
  - Used with SP or BP
  - SS:SP or SP:BP are valid Segment Addresses

IP, SP, BP, SI, DI - Offset Registers

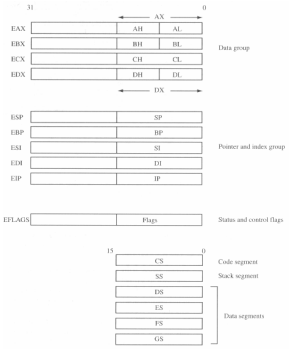
Contains "*Index Value*" for Memory Address

- IP, Instruction Pointer
  - Used to "point" to Instructions
  - Determines a Memory Address (along with CS)
  - Segmented Address written as CS:IP
- SI, Source Index; DI, Destination Index
  - Used to "point" to Data
  - Determines Memory Address (along with other registers)
  - DS, ES commonly used
- SP, Stack Pointer; BP, Base Pointer
  - Used to "point" to Data in Stack Structure (LIFO)
  - Used with SP or BP
  - SS:SP or SP:BP are valid Segment Addresses

These can also be used as General Registers !!!!!!

80386 Register Set

General Purpose



Note how registers were extended from 16 bits to 32 bits in width.

Register EAX refers to the entire 32-bit register.