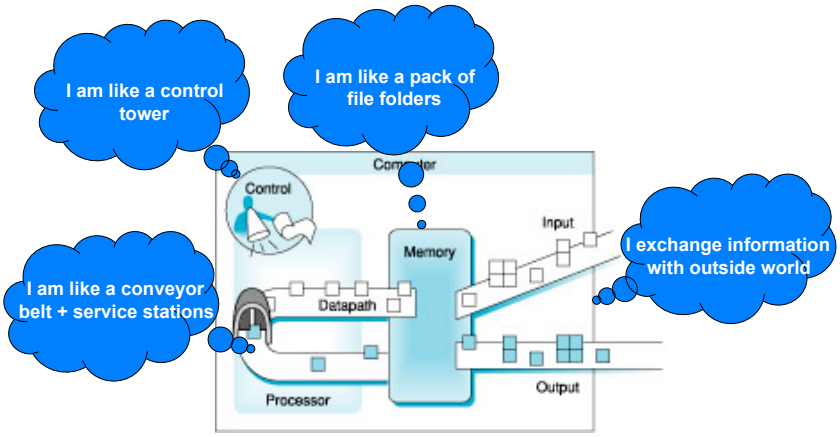
CS/COE0447: Computer Organization and Assembly Language

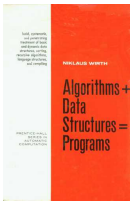
Terminology and Concepts

Five classic components



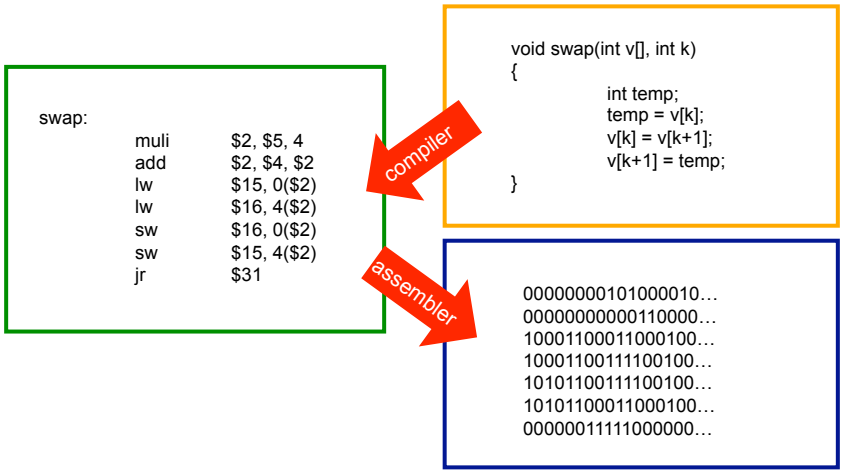
Some definitions for 447

- What is a program?
  - A file of instructions (a static sequence of instructions)
  - Programmer had a goal when writing this program
  - Instructions *execute* on processor ⇒ a **dynamic sequence of instructions**



- What is an algorithm?
  - A procedure to solve a problem
    - E.g., sort an array
- Book titled "Algorithms + Data Structures = Programs"
  - Famous book!! by Niklaus Wirth (Uses Pascal ☺)

Machine instructions



Static vs. Dynamic Instruction

- **Static instructions**: Simply the fixed "file" of instructions making up the program
- **Dynamic instructions**: The actual instructions fetched & executed by the processor

```
int foo(int op, int num, int A[], int B[], int C[]) {
   int i;

   for (i=0; i<num; i++) {
      if (op==1) C[i]=A[i]+B[i];
      else C[i]=A[i]-B[i];
   }
}
```

this program is the "file" of instructions. it is static .

Static vs. Dynamic Instruction

- **Static instructions**: Simply the fixed "file" of instructions making up the program
- **Dynamic instructions**: The actual instructions fetched & executed by the processor

```
int foo(int op, int num, int A[], int B[], int C[]) {
   int i;

   for (i=0; i<num; i++) {
      if (op==1) C[i]=A[i]+B[i];
      else C[i]=A[i]-B[i];
   }
}
```

foo(1, 10, A, B, C);

loop iterates 10 times

if-statement done 10 times

else-statement NEVER EXECUTED

this sequence (red) of what is actually executed is dynamic

Static vs. Dynamic Instruction

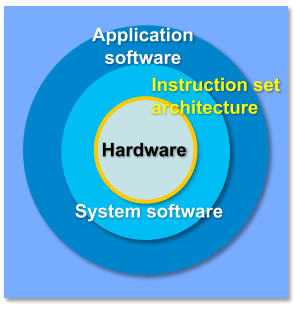
- **Static instructions**: Simply the fixed "file" of instructions making up the program
- **Dynamic instructions**: The actual instructions fetched & executed by the processor

- Observations
  - Static includes much that won't be executed
  - 90-10 (or 80-20) rule: 90% of execution in 10% of the code
  - Typically, many more dynamic instructions (executed) than static code
  - Total number of instructions executed: **instruction count**

- **Instruction count**: For same processor, fewer instructions usually means better performance (faster program)

Instruction set architecture

- A "programmer's reference manual" (PRM) for a processor will disclose the ISA of the processor

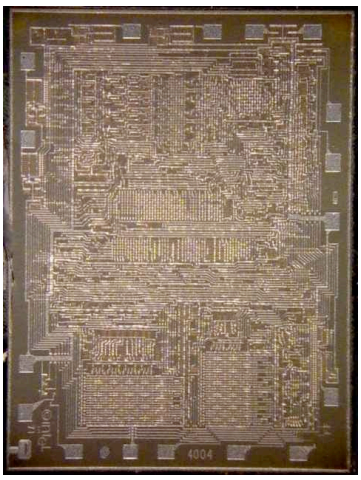


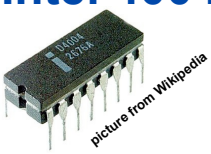
- You are a system software programmer
- Components of ISA in PRM
  - Data types the processor supports
  - **Registers and their usage**
  - **Instructions and their definitions**
  - **Processor modes**
  - Exception mechanism
  - (Compatibility issues)

Register

- It's ***storage in your processor*** that you can directly address and access in an instruction
- If your processor is 32-bit, your registers are (usually) 32 bits wide
- Depending on the processor, there can be many registers or only a few of those
  - Registers were a scare resource – they occupy chip space
  - Today we can put many registers; the concern is the access time and the power consumption

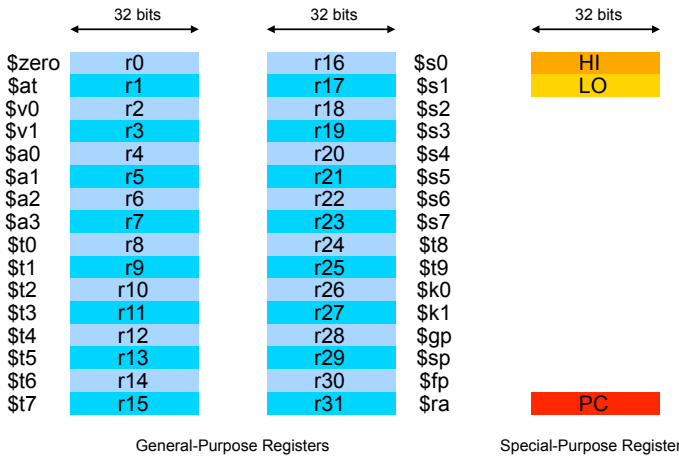
Early processor example: Intel 4004





- b. 1971
- 740 kHz
- It's considered the first microprocessor by many people
- ~2000 transistors
- 4-bit processor
- 1 register – accumulator
- about $15-20 on eBay!

MIPS registers

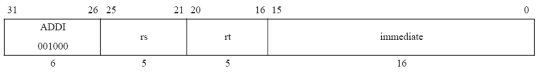


Instruction

- Unit of program execution; program consists of instructions
- **Describes an operation that the processor understands how to perform**
- The amount of work defined for an instruction is usually small
  - Add two numbers in registers (**add $t0,$t1,$t2**)
  - Compare two numbers in registers (**slt $t0,$t1,$t2**)
  - Make a jump in the program if the first number is smaller than the second number
- Complex instructions may ease your programming…
  - For example, “multiply two numbers from memory location A & B and iterate this 100 times or until you meet two zeros”
  - BUT, your processor implementation can become quite complex (slow!)

Instruction description example

Add Immediate Word — ADDI



Format: `ADDI rt, rs, immediate` — MIPS32

Purpose:

To add a constant to a 32-bit integer. If overflow occurs, then trap.

Description: `rt ← rs + immediate`

The 16-bit signed *immediate* is added to the 32-bit value in GPR *rs* to produce a 32-bit result.

- If the addition results in 32-bit 2's complement arithmetic overflow, the destination register is not modified and an Integer Overflow exception occurs.
- If the addition does not overflow, the 32-bit result is placed into GPR *rt*.

Restrictions:

None

Operation:

```
temp ← (GPR[rs]31||GPR[rs]31..0) + sign_extend(immediate)
if temp32 ≠ temp31 then
   SignalException(IntegerOverflow)
else
   GPR[rt] ← temp
endif
```

Exceptions:

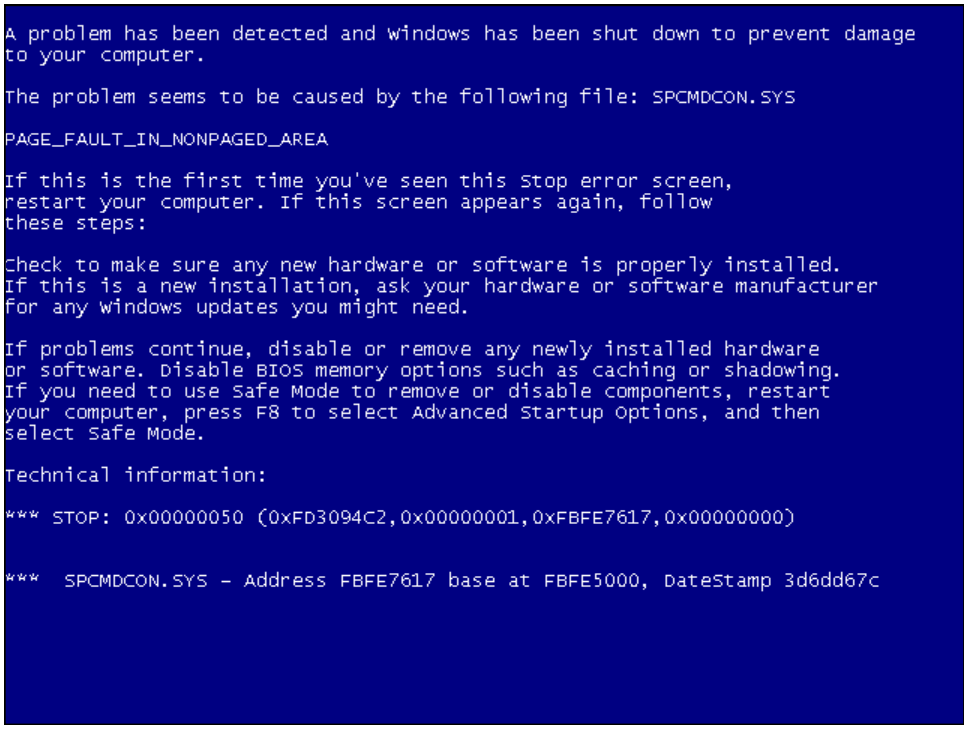
Integer Overflow

Programming Notes:

ADDIU performs the same arithmetic operation but does not trap on overflow.

Processor modes

- "User mode"
  - Ordinary programs run in this mode
  - Most instructions can be executed in this mode (e.g., add, load)
  - Critical system resources are not directly accessed
  - What about other users' programs?
- "Privileged mode"
  - System software runs in this mode
  - Some instructions can be executed only in this mode
  - Critical system resources are managed by the system software (i.e., OS)
- What happens if the user program tries to access prohibited, privileged system resource?



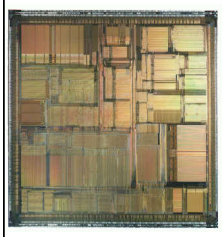
Switching between modes

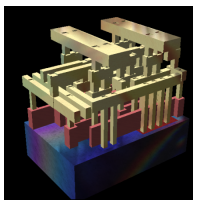
- When powered on, a processor will be in its privileged mode
- When the system boots up and becomes initialized, the system starts to execute user programs or interact with the user
- The processor switches back and forth between the modes when
  - There is an *instruction-induced exception*
    - Divide-by-zero
    - Access to unallocated memory space
    - System calls
  - There is an *interrupt from the I/O system*
    - Clock interrupt (possibly causing another program to run)
    - Keyboard & mouse

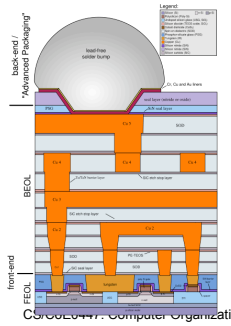
Inside your PC

- Integrated circuits (ICs)
  - CPU (central processing unit), chipsets, memory, peripheral I/O chips (e.g., for USB, IEEE1394, …)
- Printed circuit boards (PCBs)
  - Substrate for ICs and interconnection
  - Distribution of clock, power supply
  - Heat dissipation
- Hard drive
- Optical storage (CD-ROM, DVD-ROM, CD-RW, …)
- Power supply
  - Converts line AC voltage (100V) to regulated DC low voltage levels
  - GND (0V), +/-12V, +/-5V, …
- Chassis
  - Holds boards, power supply, and provides physical interface for user and other system components
- Connectors and cables

Integrated circuits







- Primarily crystalline silicon
- 1mm~25mm on a side
- “Feature size”: 32~250nm
- 0.1B~2B+ transistors
- Intel Nehalem-EX 8 core: 2.3B!
- 3~10 metal “conductive” layers
- CMOS (complementary metal-oxide semiconductor) technology

- Package spreads chip level signal paths to board level
- Provides heat dissipation

Printed circuit board

Fiberglass or ceramic

1~20 conductive layers

1~20 inches on a side

IC packages are mounted and soldered on a board

Input devices

Atari 400, see Atari Museum

- Accepts input from human
- Desktop computers
  - Keyboard (wireless / bluetooth)
  - Mouse (from mechanical to optical, touchpad)
  - Joystick (Atari 400 to Guitar Hero!)
  - Web camera, voice recognition, home control (e.g., video streaming)
  - What's next????
- Servers
  - Terminals on network, home control
- Cell phone – embedded computer
  - Keypad
  - Voice recognition
  - Gesture based (e.g., bumping to exchange contact)

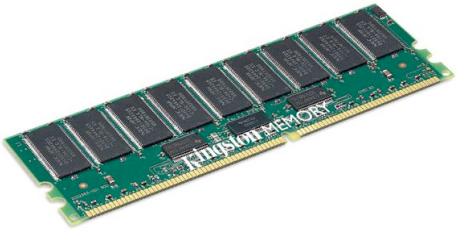
Output devices

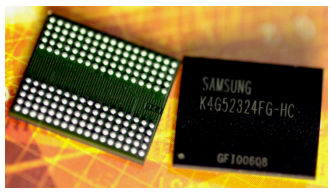
ADM3A, from Wikipedia

- Passes information to human, machine, actuator
- Desktop computers
  - Display (transition from CRT to LCD, 10 to 30", possibly more than one)
  - Sound (from little speaker to complete stereo sound system)
  - Printer (from line dot matrix to high DPI color photo printing at home)
- Servers
  - Terminals (thin client) or other computer on network
- Cell phone – embedded computer
  - Screen
  - Sound
  - Vibration

Main memory

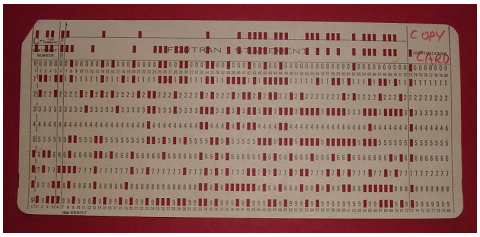
- DRAM (dynamic random access memory) is the choice technology (invented mid 1960s, capacity 1024 bits)
  - Large capacity
  - Low price
  - Standard products from multiple vendors
- What is SRAM?





Storage and computer networks

- Storage and network devices are I/O devices like keyboard and graphics card
- Today, storage and network devices need high bandwidth
  - Fast data retrieval and storing
  - Fast communication

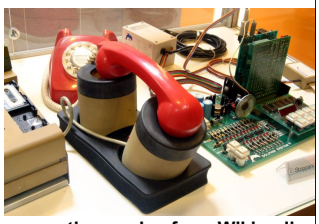


punch card - from Wikipedia

Storage

- Secondary storage (c.f., main memory)
- Non-volatile
- Stores programs, user-saved data, etc.
- In PC/server domain, magnetic disk (hard disk) is dominant
- In embedded computers, “flash” memory and “ROM” are quite popular
- Due to performance, power, and reliability issues, solid-state disk drives (based on flash memory technology) become increasingly common
  - Here today, but limited capacity (128 GB & HUGE demand for parts!)
  - Can’t quite compete (yet) with hard disks for $/gigabyte
  - Requires techniques to work around “wear-out”

Computer networks



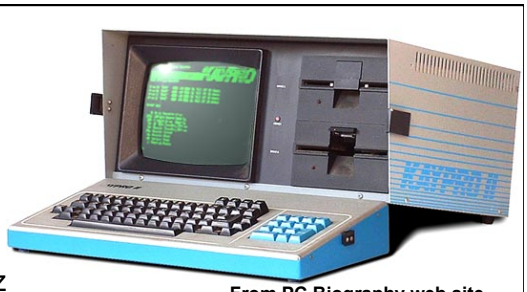
acoustic coupler, from Wikipedia
"Shall we play a game?"

- Local area network (LAN)
  - Within a limited distance (e.g., inside a building)
  - 100/100/1000Mbps, …
- Wide area network (WAN)
  - Connecting smaller networks far apart
- At home
  - Modem: acoutstic coupler - 110 baud, 300 baud to direct connect - 14.4kbps, 28.8kbps, 33.6kbps, 56kbps
  - Cable modem/DSL: several hundred kbps ~ several Mbps
  - Optical connection delivering multiple media streams (Verizon FiOS)
  - Home network
- Proliferation of wireless LAN (IEEE 802.11)
  - 1~100Mbps (newest proposed standard 802.11n up to 600 Mbps)

Exponential technology advances

- Important trends drive computer system design
- Memory
  - DRAM capacity: 2x / 2 years (since '96)
  - 64x capacity improvement in the last decade
- Processor
  - Speed (in terms of clock frequency): 2x / 1.5 years (since '85)
    - It slows down these days
  - 100x performance improvement in the last decade
- Disk
  - Capacity: 2x / 1 year (since '97)
  - 250x capacity improvement in the last decade

My computer



From PC Biography web site

- Around 1983
- CPU: Zilog 80 8-bit, 4 MHz
- 64KB RAM
- 2 5.25” built-in singled-sided 190kb floppy
- 9-inch text-only green monitor (no graphics!!)
- Attached keyboard (no mouse!!)
- Ran CP/M operating system
  - Famous story of losing out to MS
- Weighed about 22-23 lbs
- Came with lots of software (an entire application suite!)

If I buy a new PC in the future…

- Processor: (something x86) @at least 3GHz
  - # of cores in this processor must be 8 or more
- Memory: at least 16GB
- Disk capacity: several TB
- Optical drive: Blu-ray enabled DVD/CD drive
- Rich I/O support (USB 3.0, WiFi, etc.)
- Small form factor!!!!
- New units: Mega to Giga, Giga to Tera, (Tera to Peta, Peta to Exa, Exa to Zetta, Zetta to Yotta)