Processor Basics

Introduction

- Processor designed for a variety of computation tasks
- Carefully designed since higher NRE cost is acceptable
  - NRE: Non-recurring engineering
  - Can yield good performance, size and power
  - manufacturer spreads NRE over large numbers of units
    - Motorola sold half a billion 68HC05 microcontrollers *in 1996 alone*
- Low NRE cost, short time-to-market/prototype, high flexibility
  - User just writes software; no processor design

Processors

Processor	Clock speed	Periph.	Bus Width	MIPS	Power	Trans.	Price
General Purpose Processors							
Intel PIII	1GHz	2x16 K L1, 256K L2, MMX	32	~900	97W	~7M	$900
IBM PowerPC 750X	550 MHz	2x32 K L1, 256K L2	32/64	~1300	5W	~7M	$900
MIPS R5000	250 MHz	2x32 K 2 way set assoc.	32/64	NA	NA	3.6M	NA
StrongARM SA-110	233 MHz	None	32	268	1W	2.1M	NA
Microcontroller							
Intel 8051	12 MHz	4K ROM, 128 RAM, 32 I/O, Timer, UART	8	~1	~0.2W	~10K	$7
Motorola 68HC811	3 MHz	4K ROM, 192 RAM, 32 I/O, Timer, WDT, SPI	8	~.5	~0.1W	~10K	$5
Digital Signal Processors							
TI C5416	160 MHz	128K, SRAM, 3 T1 Ports, DMA, 13 ADC, 9 DAC	16/32	~600	NA	NA	$34
Lucent DSP32C	80 MHz	16K Inst., 2K Data, Serial Ports, DMA	32	40	NA	NA	$75

Sources: Intel, Motorola, MIPS, ARM, TI, and IBM Website/Datasheet; Embedded Systems Programming, Nov. 1998

FIU
Florida International University

von Neumann architecture

- Memory holds data, instructions.
- Central processing unit (CPU) fetches instructions from memory.
- CPU registers
  - program counter (PC), instruction register (IR), general-purpose registers, etc.

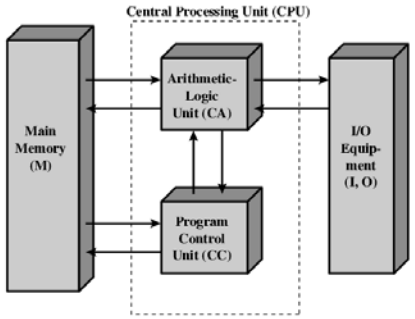
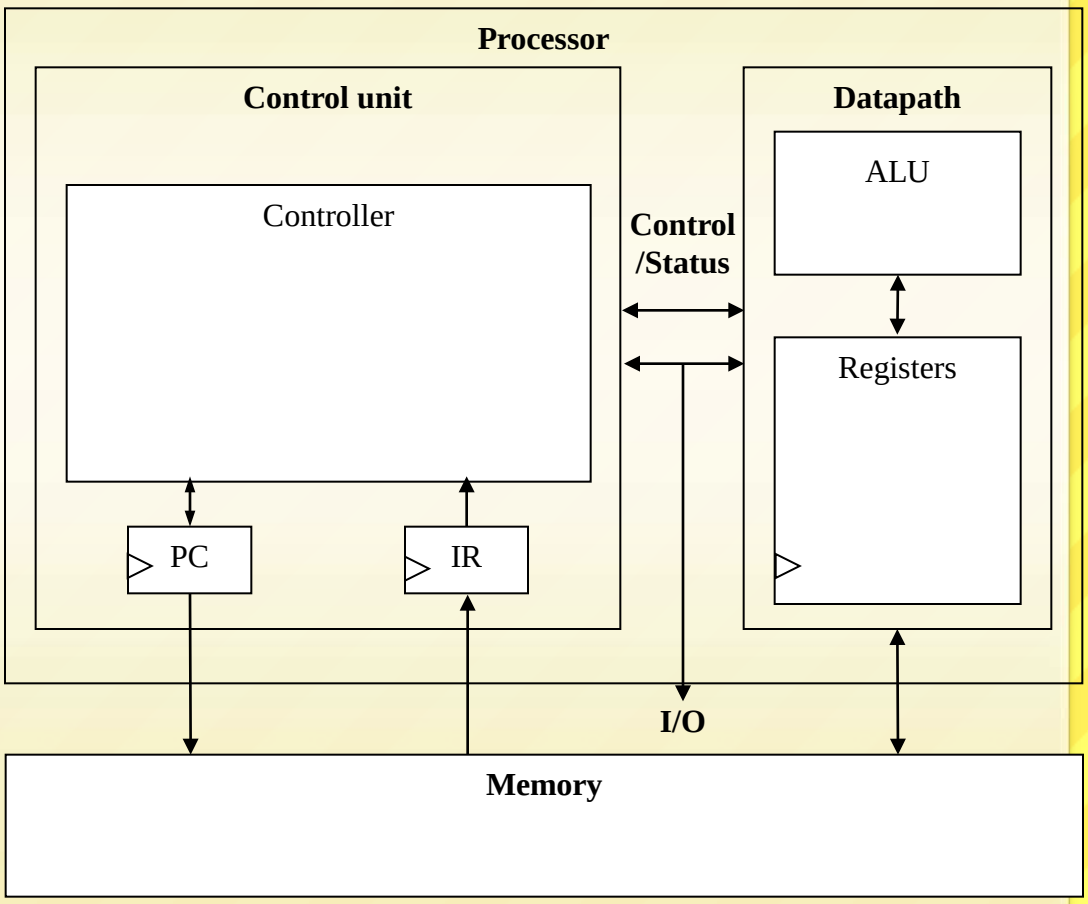


Figure 2.1 Structure of the IAS Computer

Basic Architecture

- Control unit and datapath
  - Datapath
    - The connection of all the functional units (ALU, MUXes, Registers) for each arithmetic operation
  - Control units
    - How datapath should be configured at different time/states
    - Control unit doesn't store the algorithm – the algorithm is "programmed" into the memory

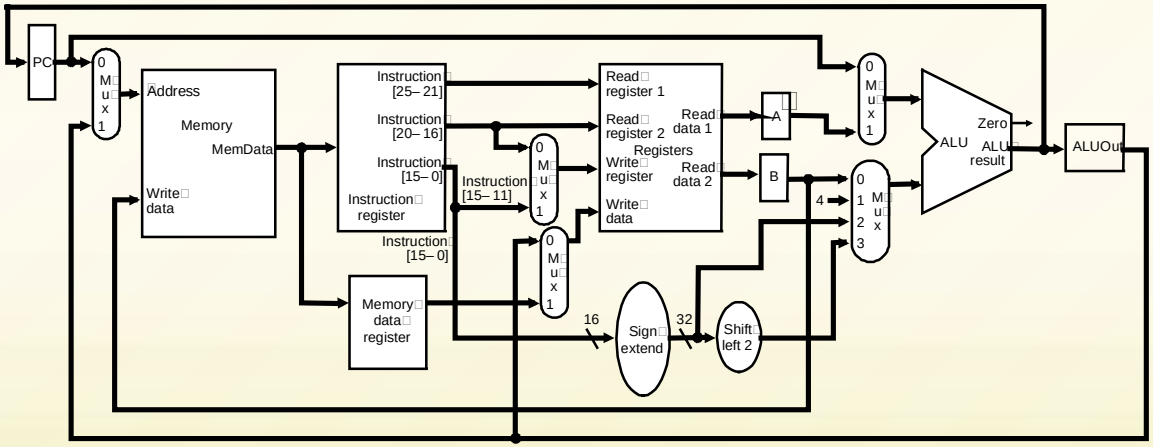


Processor Design

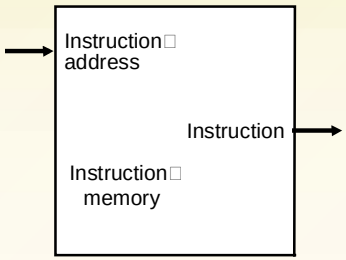
- Data path design
  - Analyze instruction set => datapath requirements
  - Select set of datapath components and establish clocking methodology
  - Assemble datapath meeting the requirements
- Control path design
  - Analyze implementation of each instruction to determine setting of control points that effects the register transfer
  - Assemble the control logic

FIU Florida International University

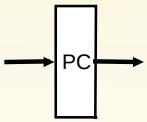
Datapath Example



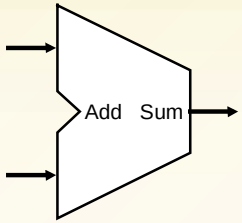
Datapath Components



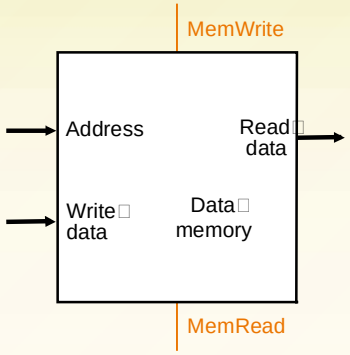
a. Instruction memory



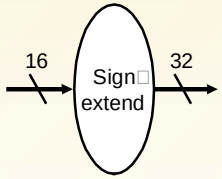
b. Program counter



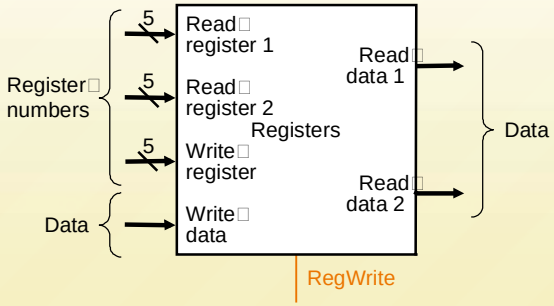
c. Adder



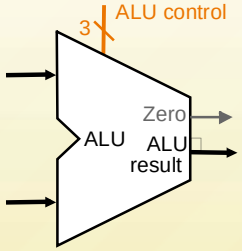
a. Data memory unit



b. Sign-extension unit



a. Registers

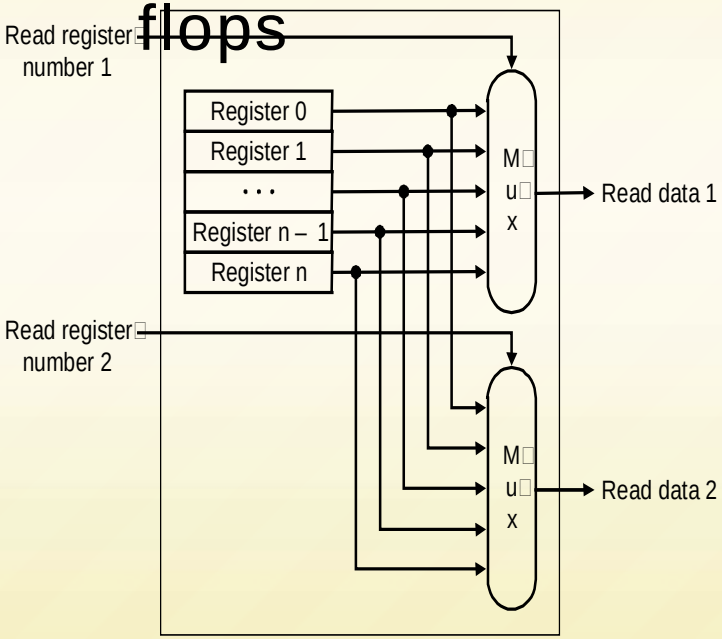


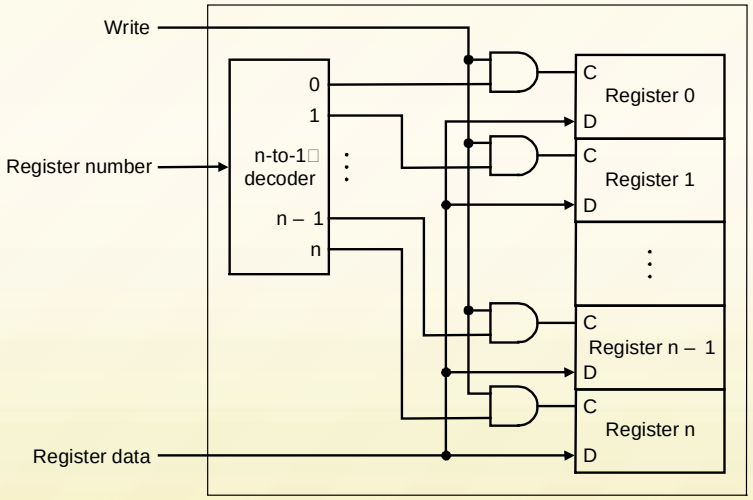
b. ALU

FIU
Florida International University

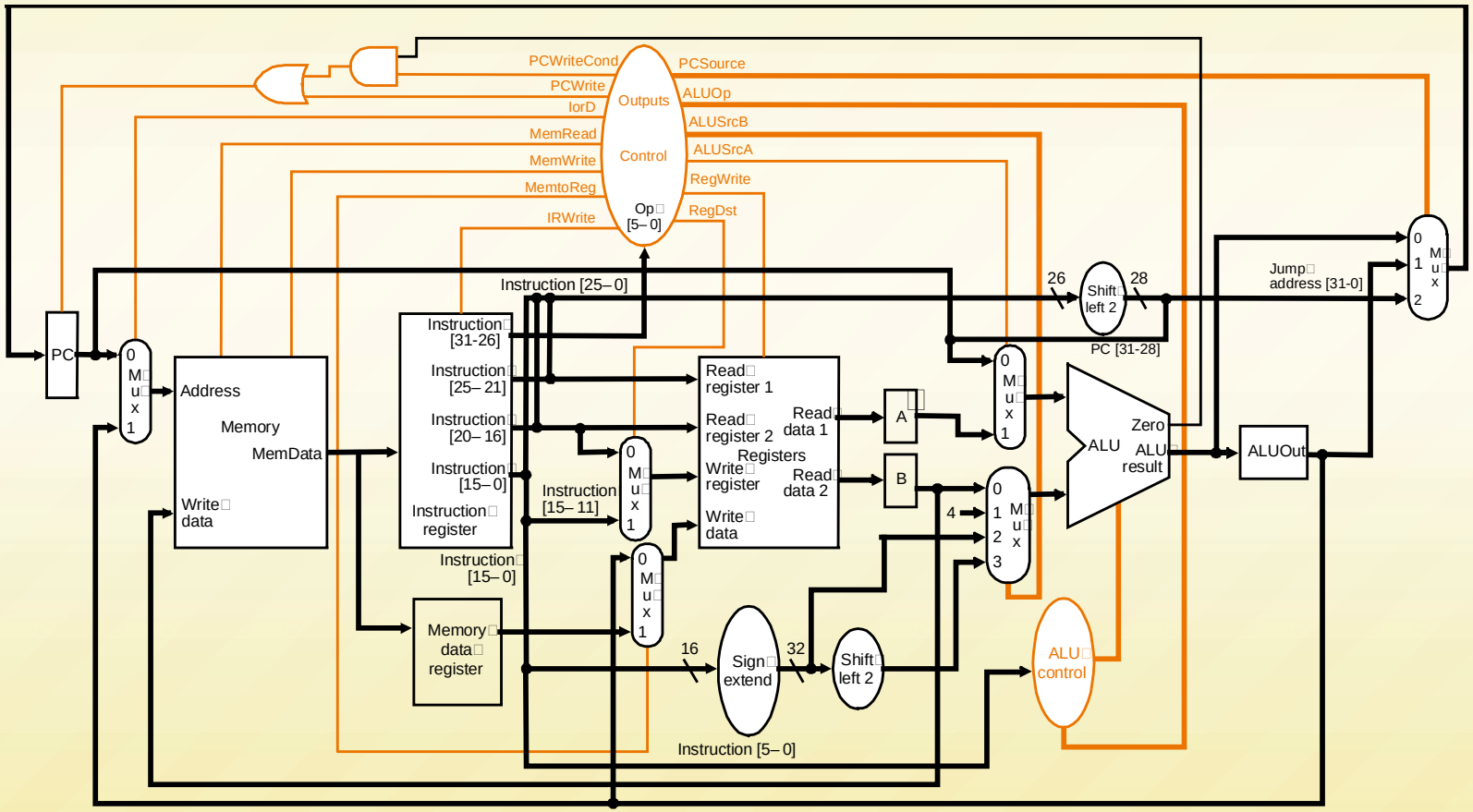
Register File

- Built using D flip-flops



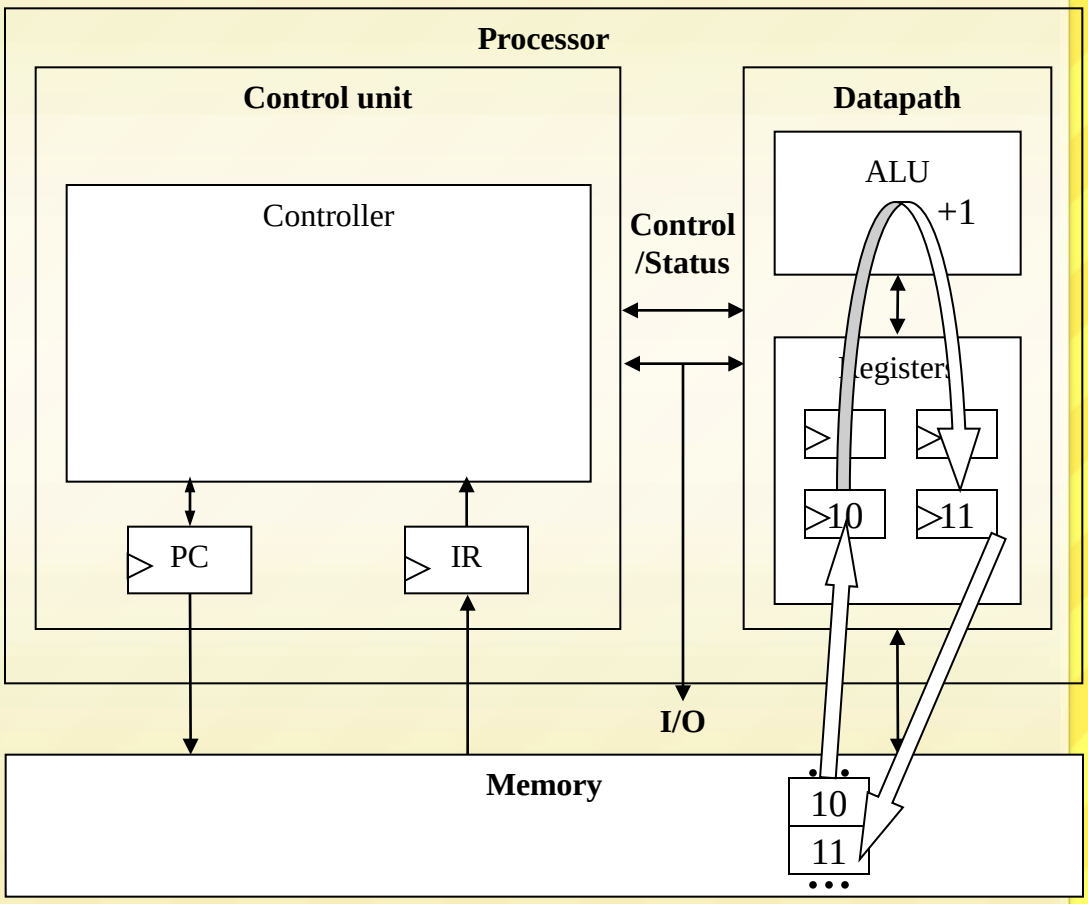


Datapath with Control



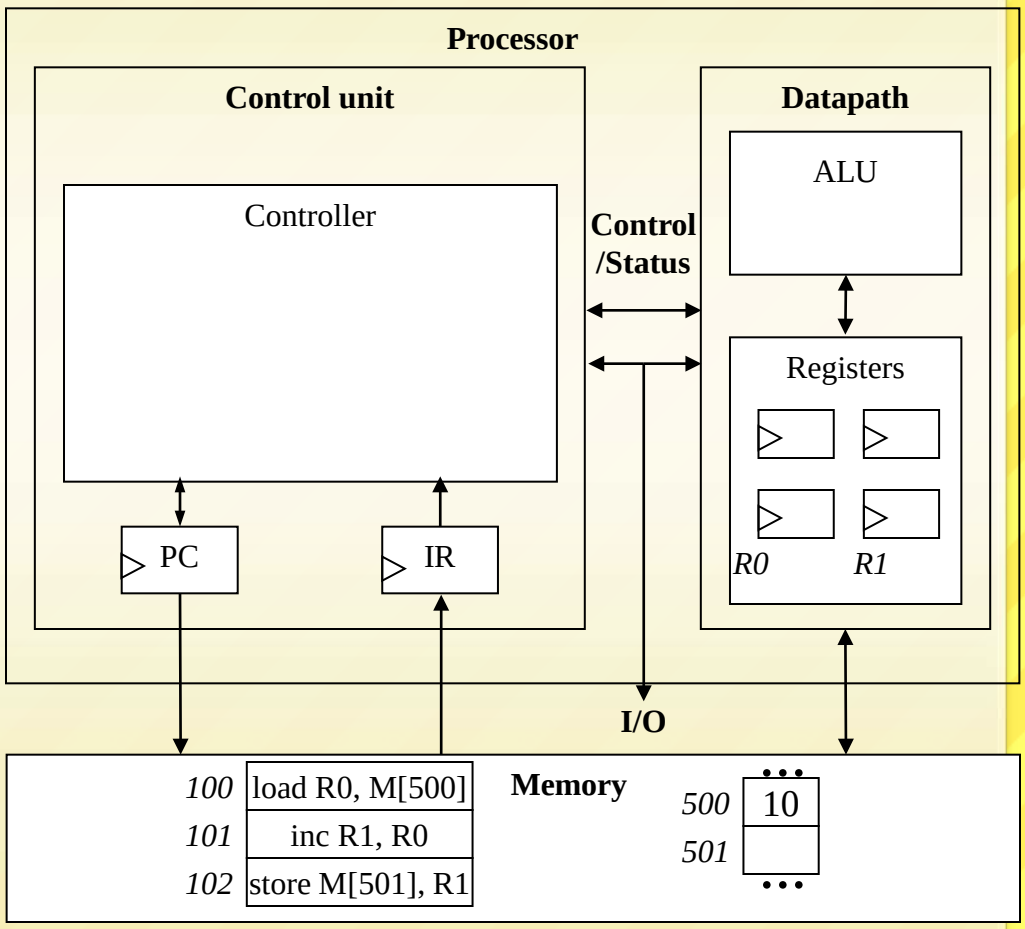
Datapath Operations

- Load
  - Read memory location into register
- ALU operation
  - Input certain registers through ALU, store back in register
- Store
  - Write register to memory location



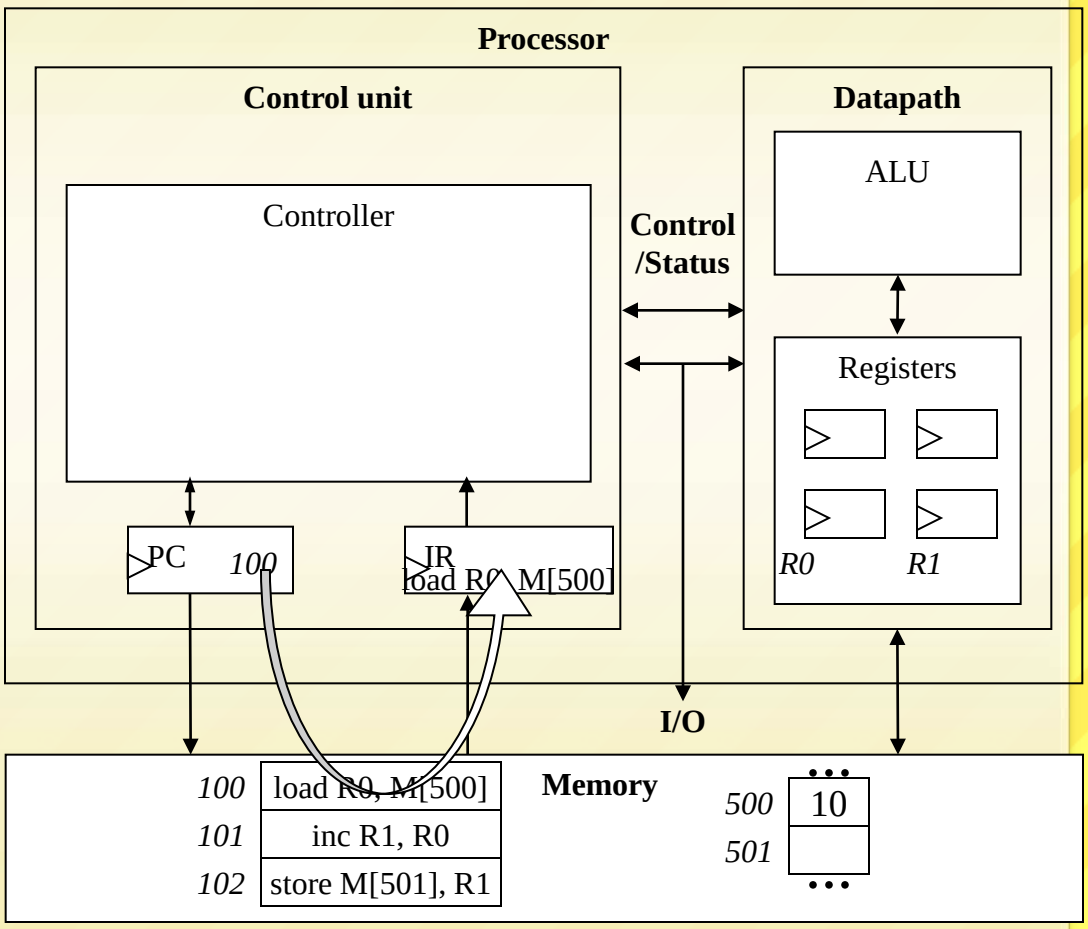
Control Unit

- Control unit: configures the datapath operations
  - Sequence of desired operations (“instructions”) stored in memory – “program”
  - Determining the operations (ALU, read/write, etc.)
  - Controlling the flow of data (multiplexer inputs)
- Instruction cycle – broken into several sub-operations, each one clock cycle, e.g.:
  - Fetch: Get next instruction into IR
  - Decode: Determine what the instruction means
  - Fetch operands: Move data from memory to datapath register
  - Execute: Move data through the ALU
  - Store results: Write data from register to memory



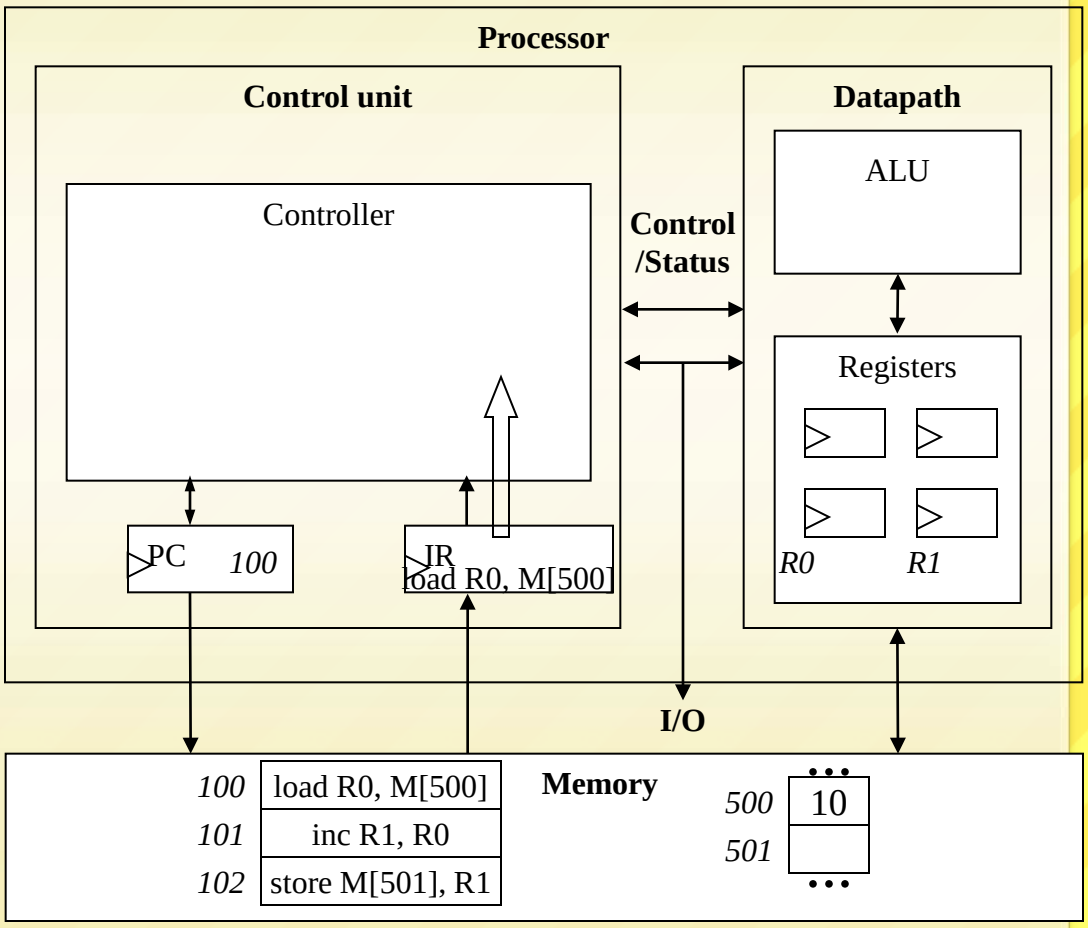
Control Unit Sub-Operations

- Fetch
  - Get next instruction into IR
  - PC: program counter, always points to next instruction
  - IR: holds the fetched instruction



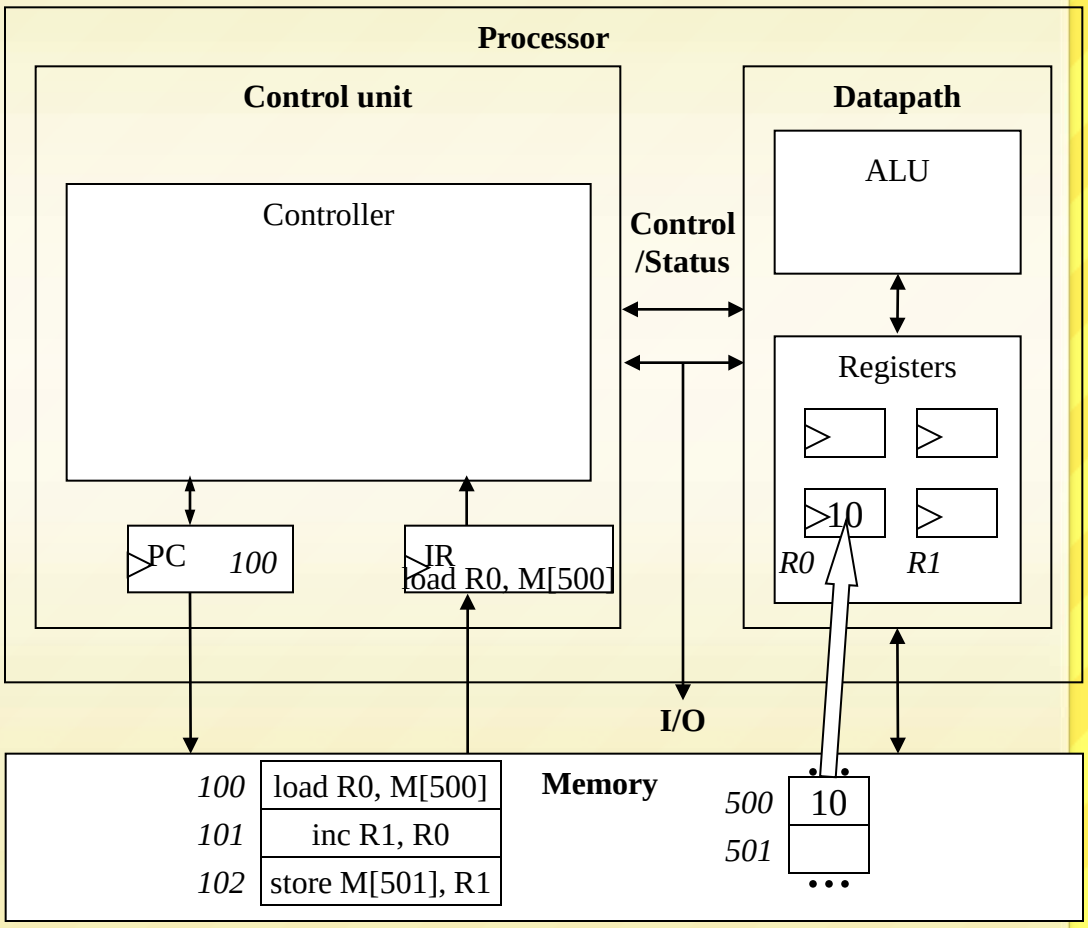
Control Unit Sub-Operations

- Decode
  - Determine what the instruction means



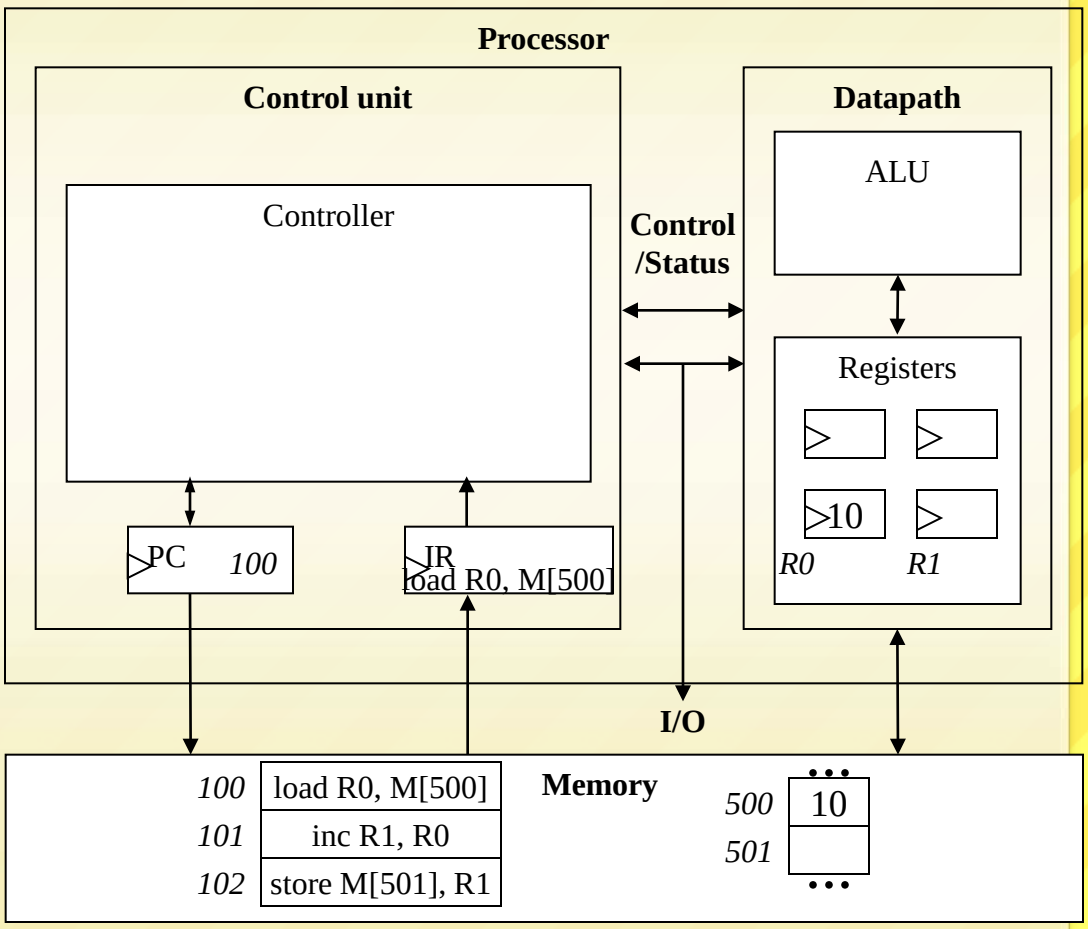
Control Unit Sub-Operations

- Fetch operands
  - Move data from memory to datapath register



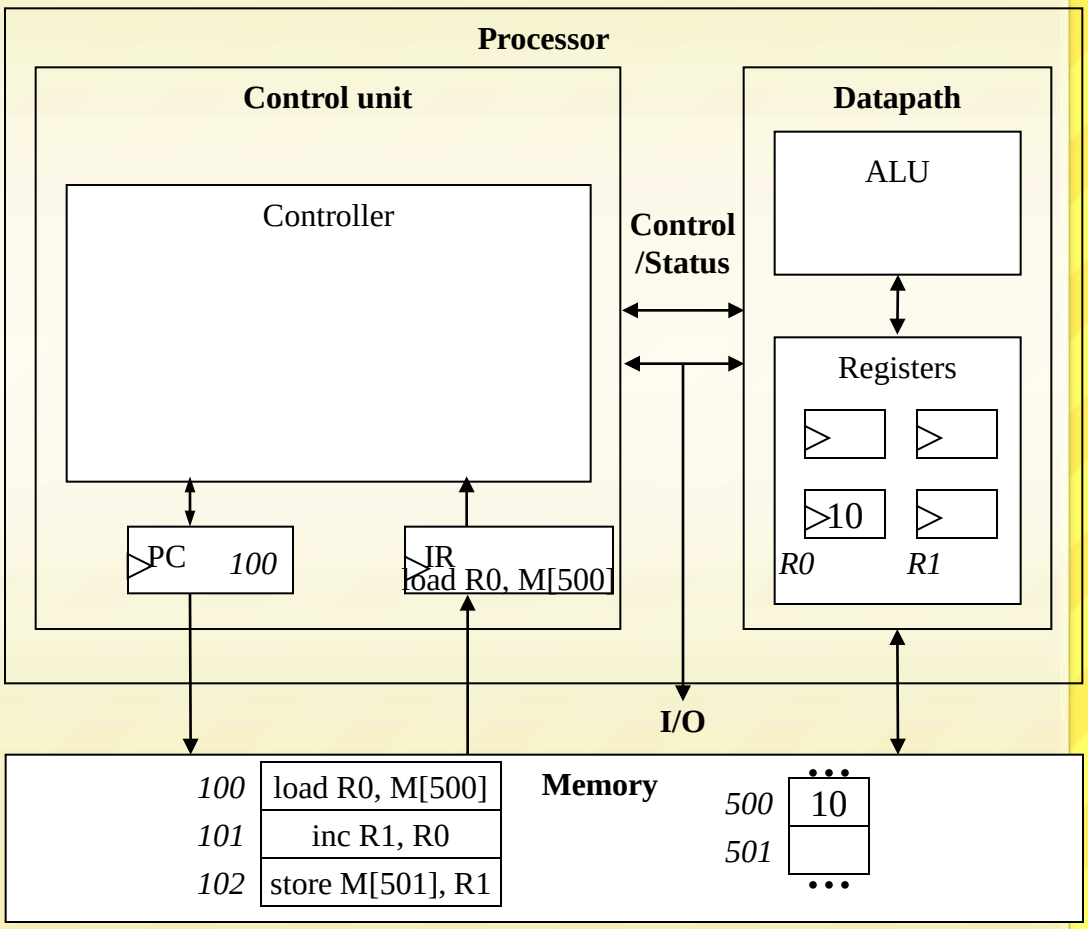
Control Unit Sub-Operations

- Execute
  - Move data through the ALU
  - This particular instruction does nothing during this sub-operation



Control Unit Sub-Operations

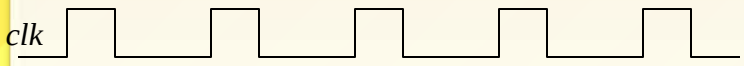
- Store results
  - Write data from register to memory
  - This particular instruction does nothing during this sub-operation

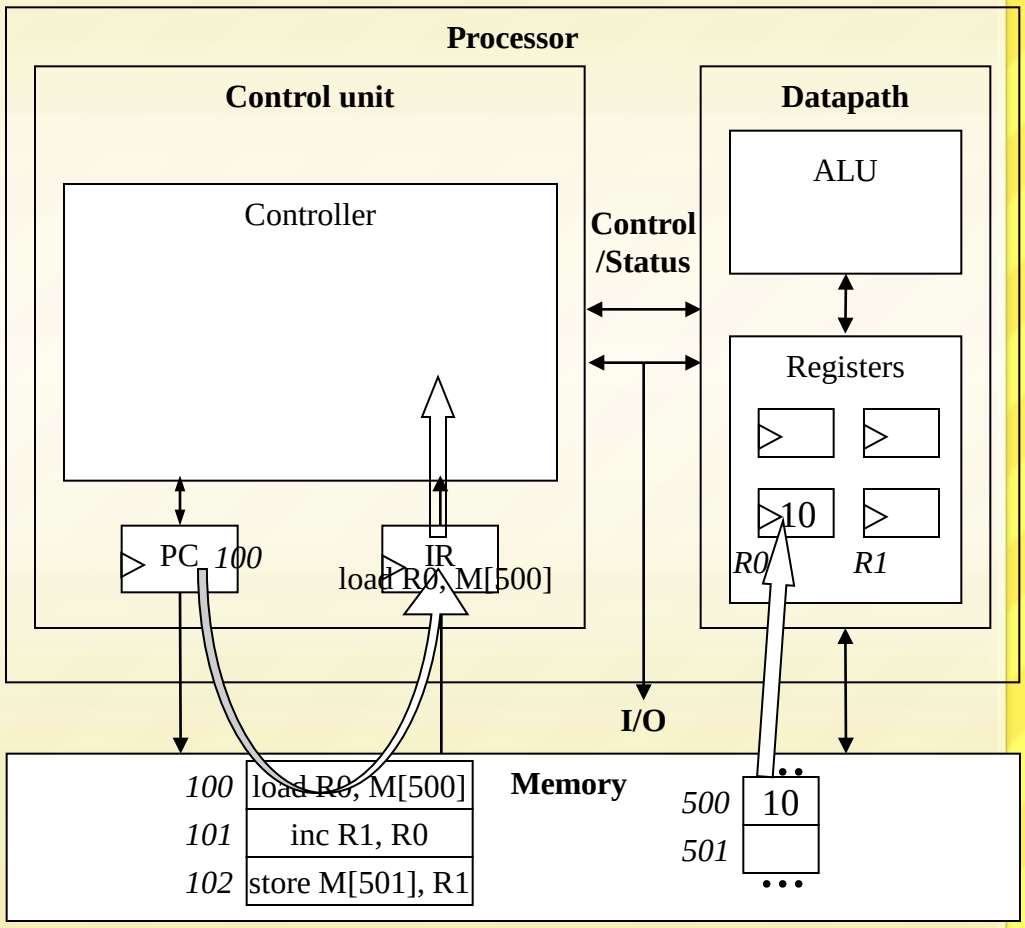


Instruction Cycles

PC=100

Fetch Decode Fetch ops Exec. Store results





Instruction Cycles

PC=100

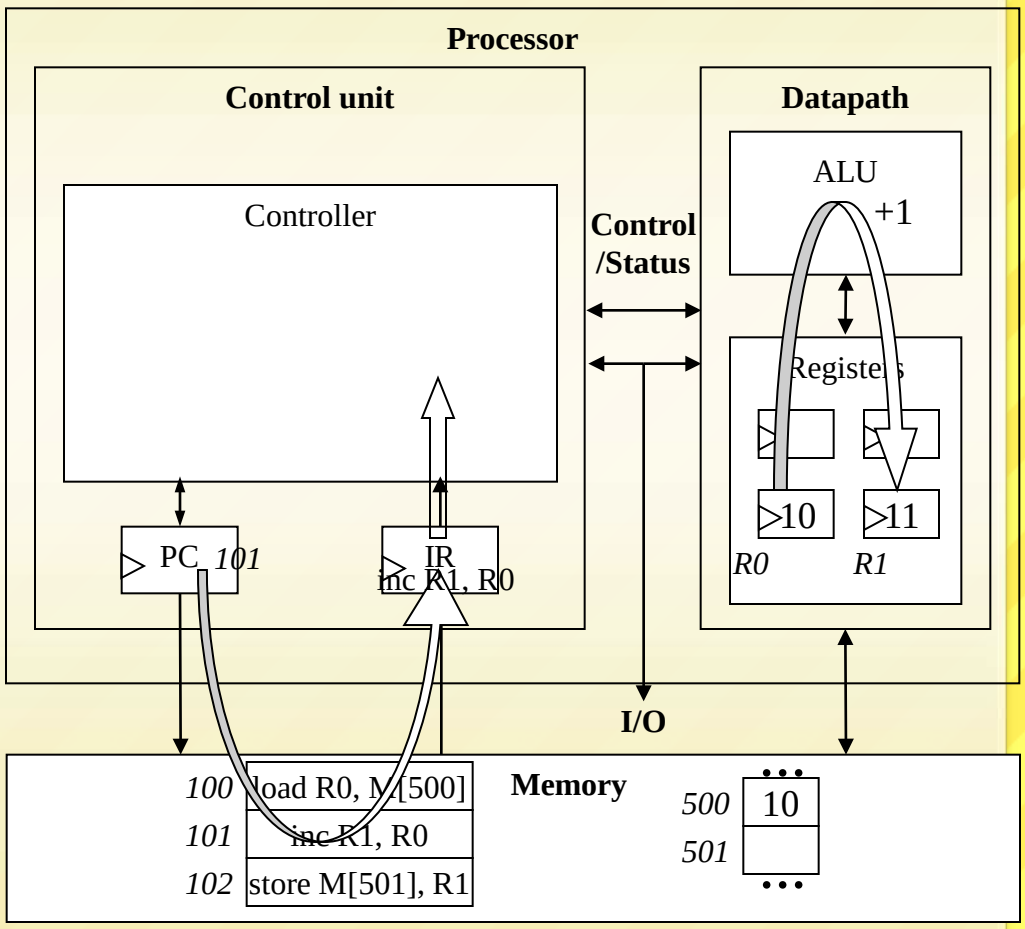
Fetch Decode Fetch ops Exec. Store results

clk

PC=101

Fetch Decode Fetch ops Exec. Store results

clk



Instruction Cycles

PC=100

Fetch Decode Fetch ops Exec. Store results

clk

PC=101

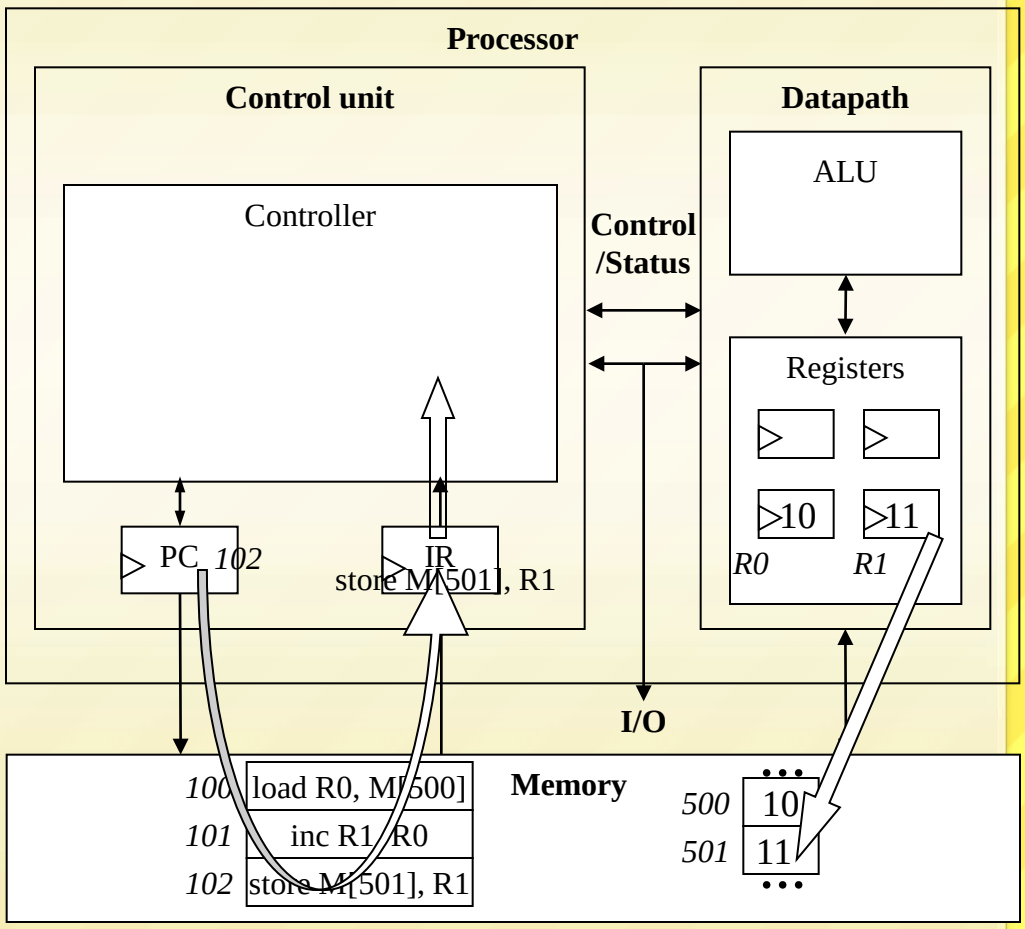
Fetch Decode Fetch ops Exec. Store results

clk

PC=102

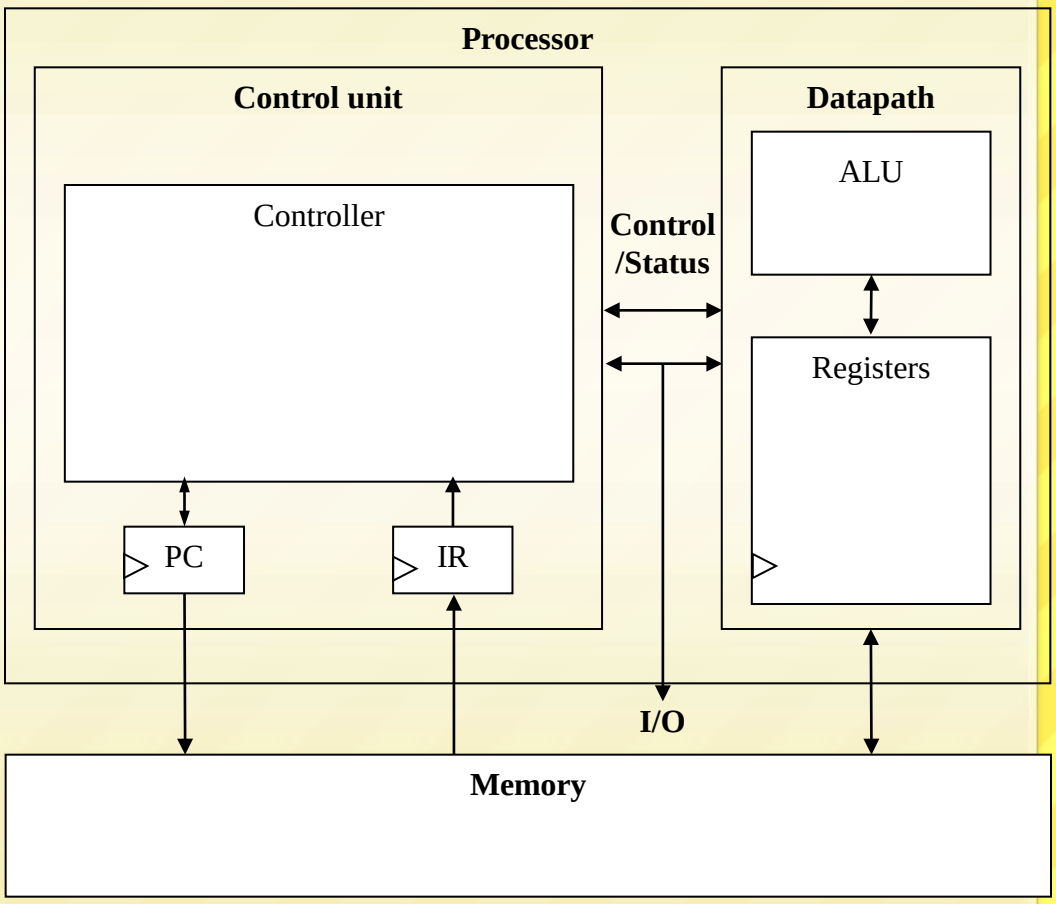
Fetch Decode Fetch ops Exec. Store results

clk



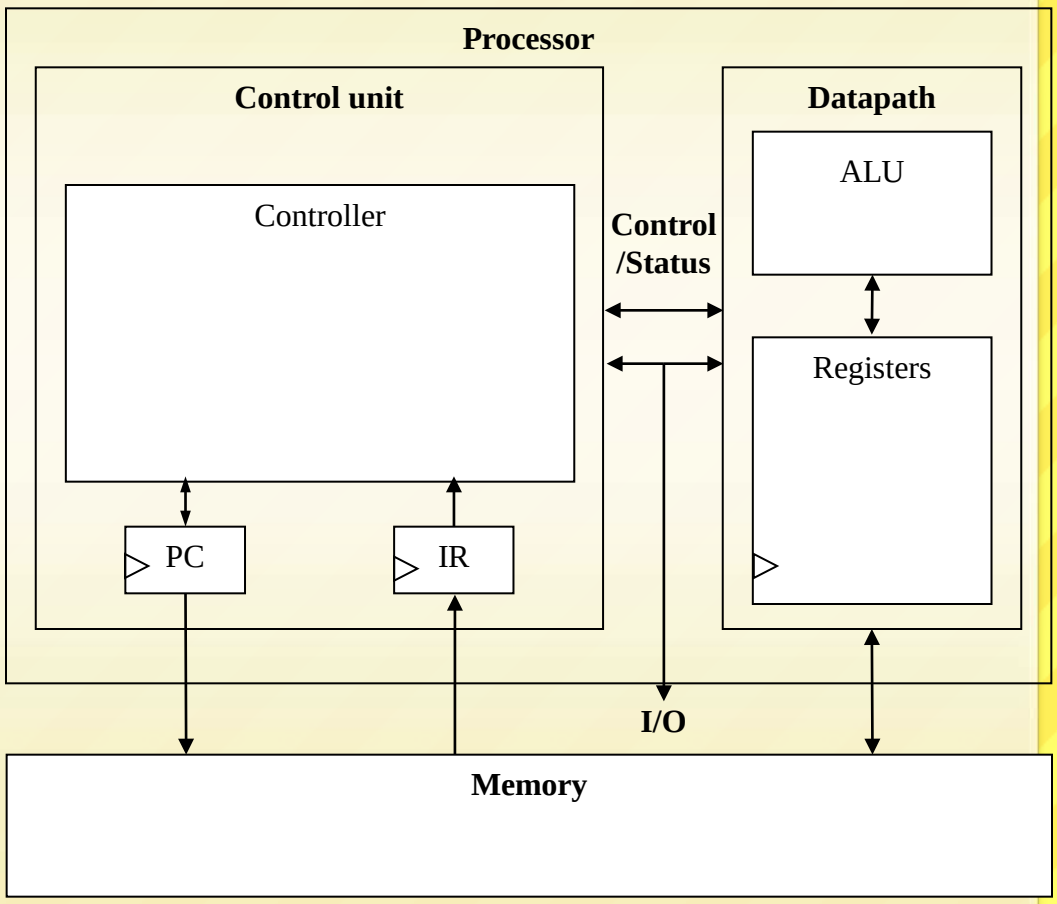
Architectural Considerations

- *N-bit* processor
  - N-bit ALU, registers, buses, memory data interface
  - Embedded: 8-bit, 16-bit, 32-bit common
  - Desktop/servers: 32-bit, even 64
- PC size determines address space

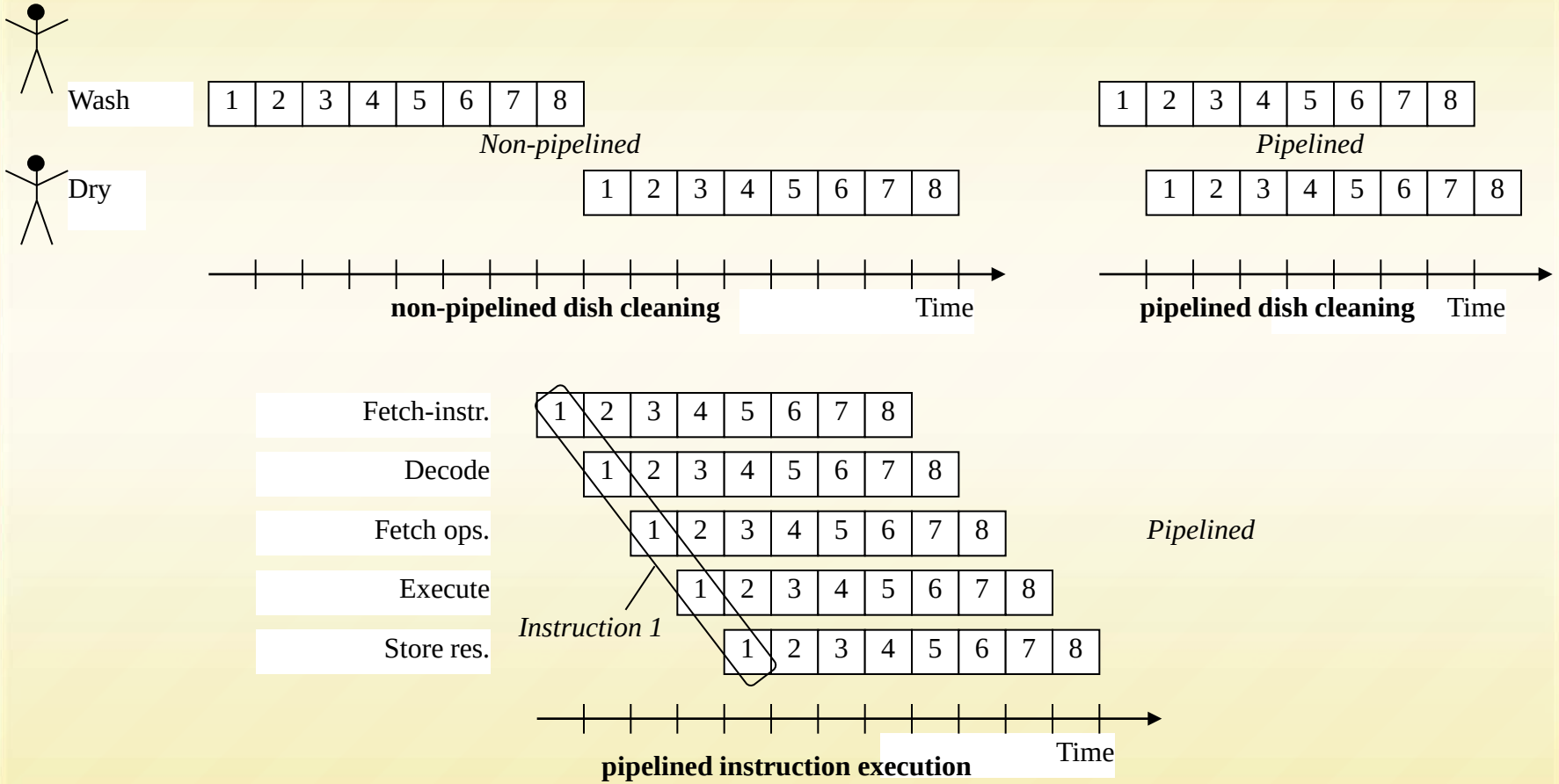


Architectural Considerations

- Clock frequency
  - Inverse of clock period
  - Must be longer than the longest single stage
  - Memory access is often the longest



Pipelining: Increasing Instruction *Throughput*



Programmer's View

- Programmer doesn't need detailed understanding of architecture
  - Instead, needs to know what instructions can be executed
- Two levels of instructions:
  - Assembly level
  - Structured languages (C, C++, Java, etc.)
- Most development today done using structured languages
  - But, some assembly level programming may still be necessary
  - Drivers: portion of program that communicates with and/or controls (drives) another device
    - Often have detailed timing considerations, extensive bit manipulation
    - Assembly level may be best for these

FIU Florida International University

Summary

- Processor architecture
  - Datapath/Control path
  - Instruction execution cycles
  - Architectural considerations and pipelining

FIU
Florida International University