von-Neumann-architectuur
Opbouw van een CPU

Processoren

14 februari 2012

Herhaling: Gates/poorten

- grafische presentatie van Booleaanse functies
- elektronische schakelingen voor
  $\wedge$, $\vee$, $\neg$, NAND, NOR
  - a NAND b = $\neg(a \wedge b)$
  - a NOR b = $\neg(a \vee b)$
- algemene schakeling: PLA

Herhaling: Geheugen

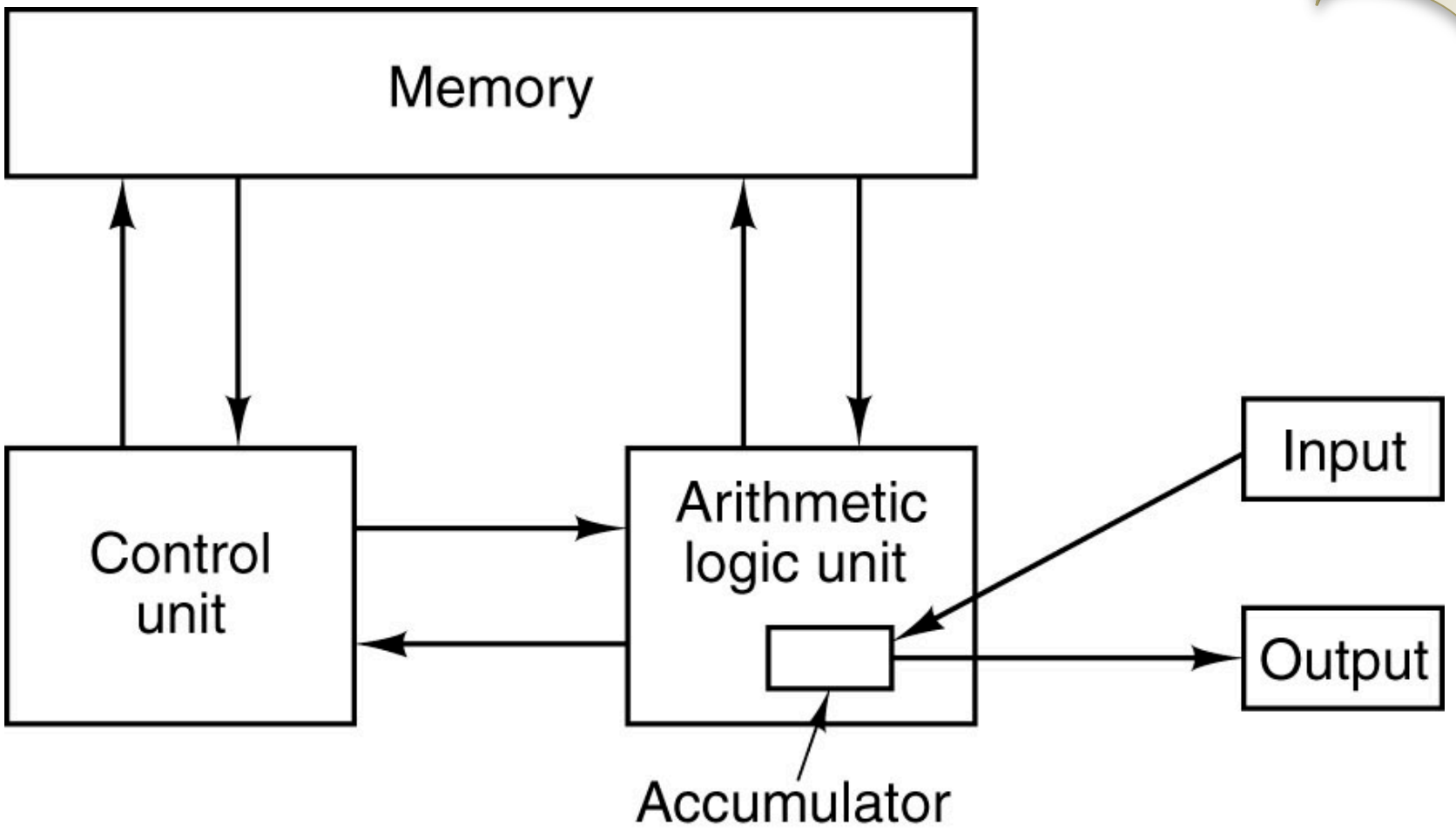
- RAM / random access memory
  = verzameling flipflops met logica eromheen
  - het juiste adres berekenen
  - input daarheen schrijven
  - output daarvandaan lezen
  - alleen verwerken
    als deze RAM-chip gekozen wordt
- HADES demo
  http://tams-www.informatik.uni-hamburg.de/applets/hades/

von Neumann-architectuur

- Welke hoofdonderdelen heeft een computer?
  - besturingseenheid
  - rekeneenheid
  - geheugen (voor programma èn gegevens)
  - in- en uitvoerapparaten

von-Neumann-architectuur

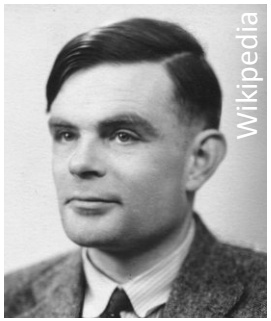
Tanenbaum hoofdstuk 1



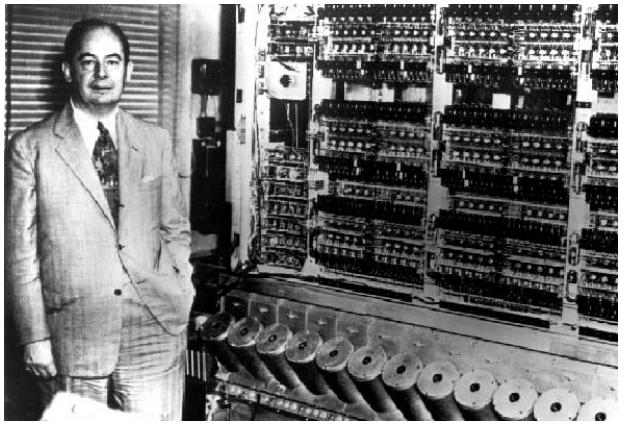
Wiens idee beklijft?

- ideeën van een “stored program computer”:
  - Konrad Zuse 1936

  - Alan Turing 1936

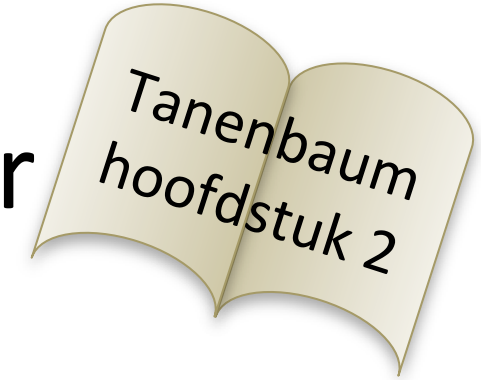


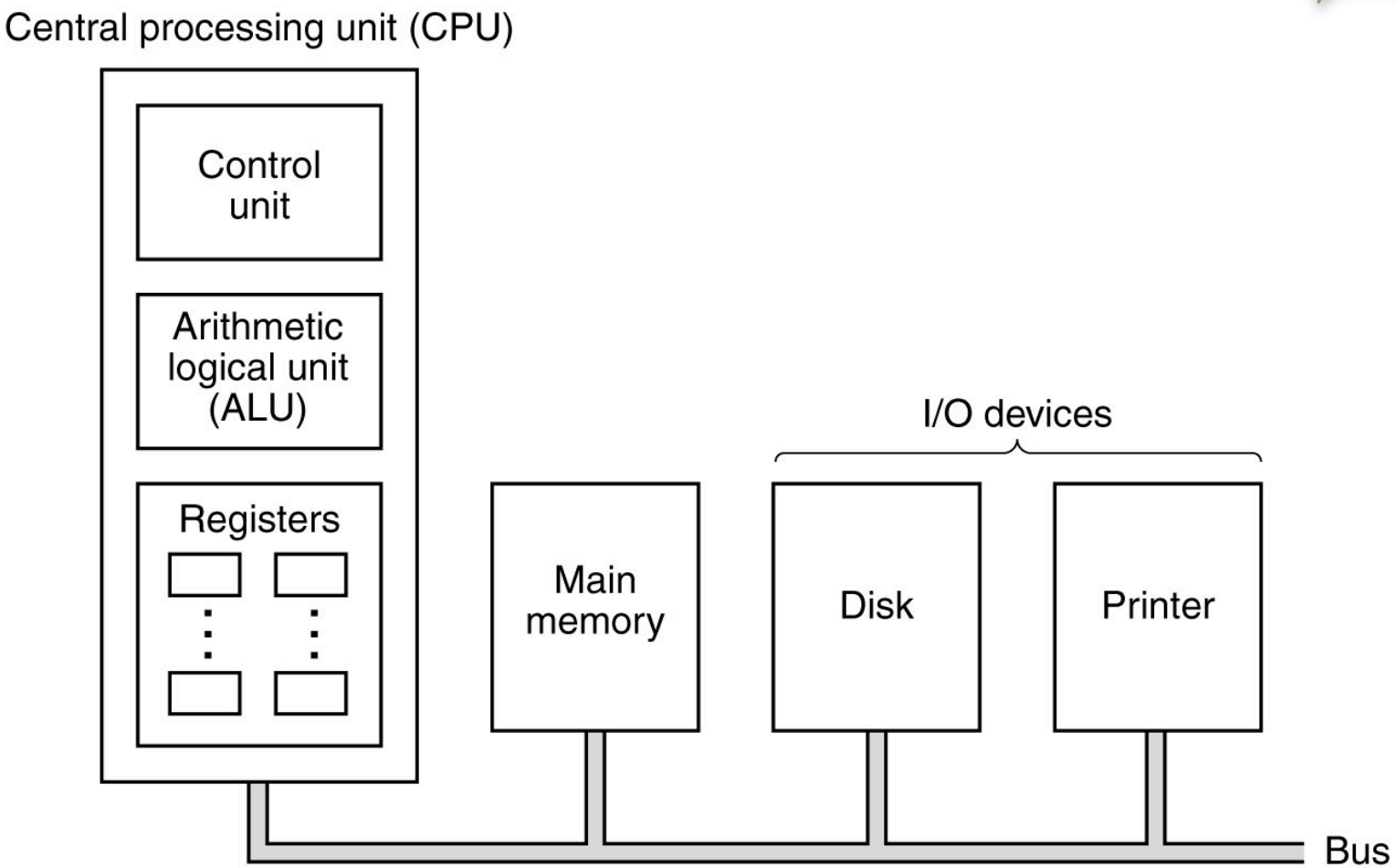
  - John von Neumann 1945, n.a.v. ideeën van Eckert en Mauchly 1943



http://www2.lv.psu.edu/OJJ/courses/ist-240/reports/spring2001/fa-cb-bc-kf/1941-1950.html

„von Neumann”-architectuur

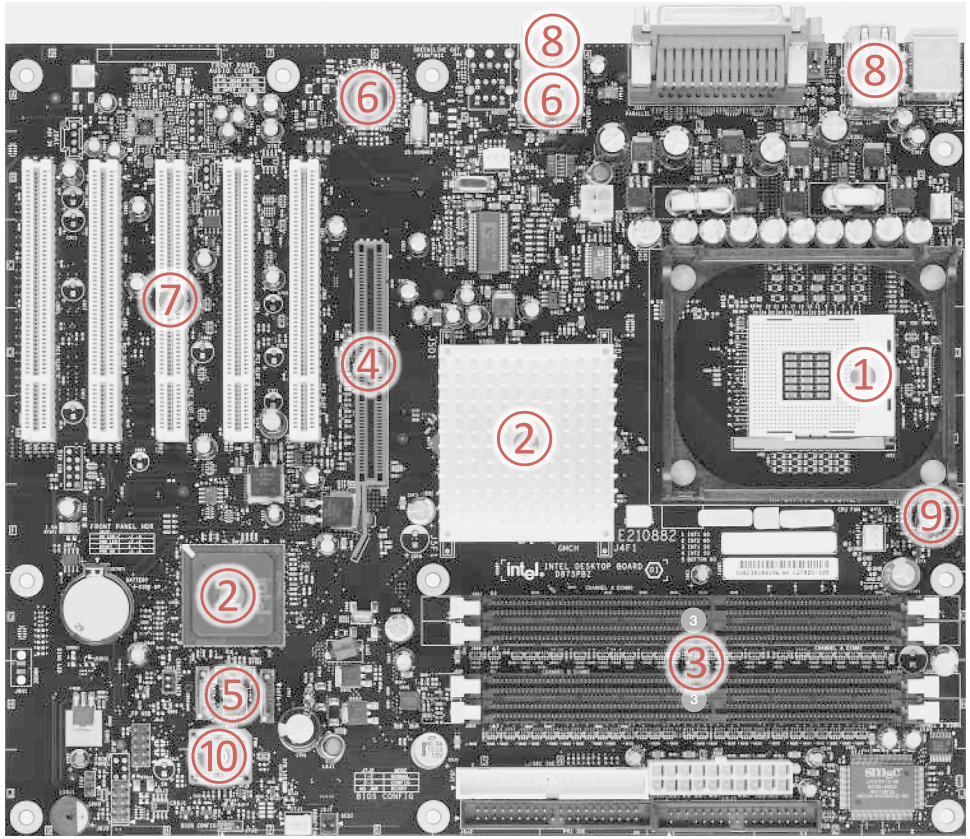




Doelen van de onderdelen

- CPU: besturen èn berekenen
- hoofdgeheugen:
  programma en gegevens opslaan
- in-/uitvoer: externe communicatie
- bus: interne communicatie

Personal Computer



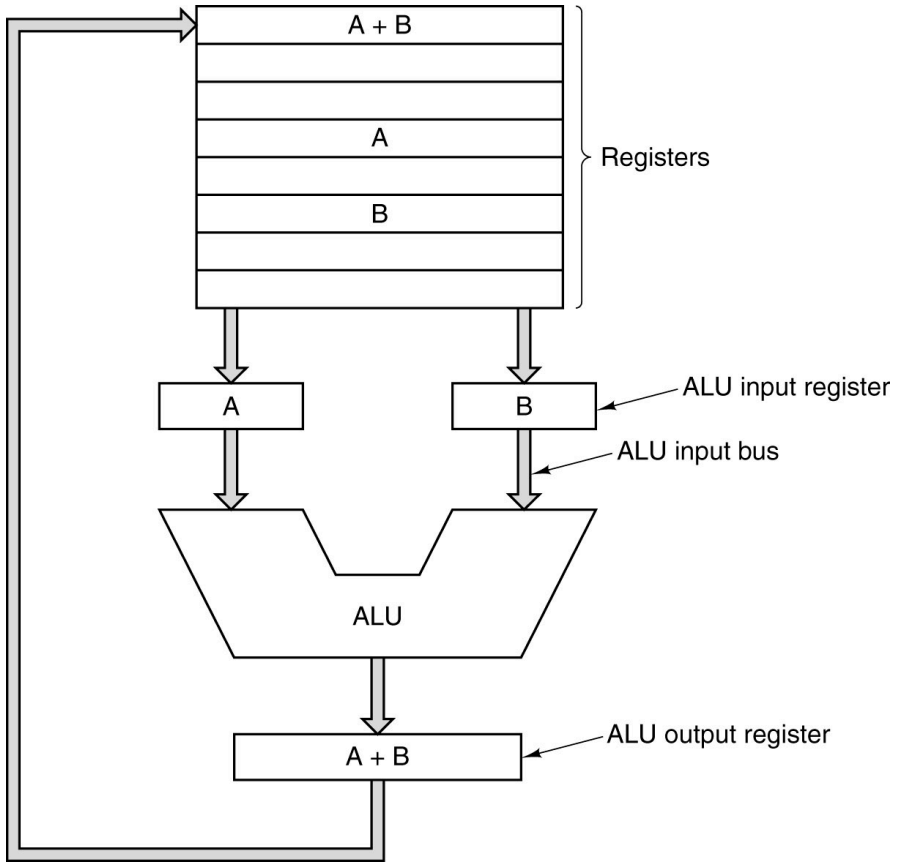
1. Pentium 4 socket
2. 875P Support chip
3. Memory sockets
4. AGP connector
5. Disk interface
6. Gigabit Ethernet
7. Five PCI slots
8. USB 2.0 ports
9. Cooling technology
10. BIOS

A printed circuit board is at the heart of every personal computer. This figure is a photograph of the Intel D875PBZ board. The photograph is copyrighted by the Intel Corporation, 2003 and is used by permission.

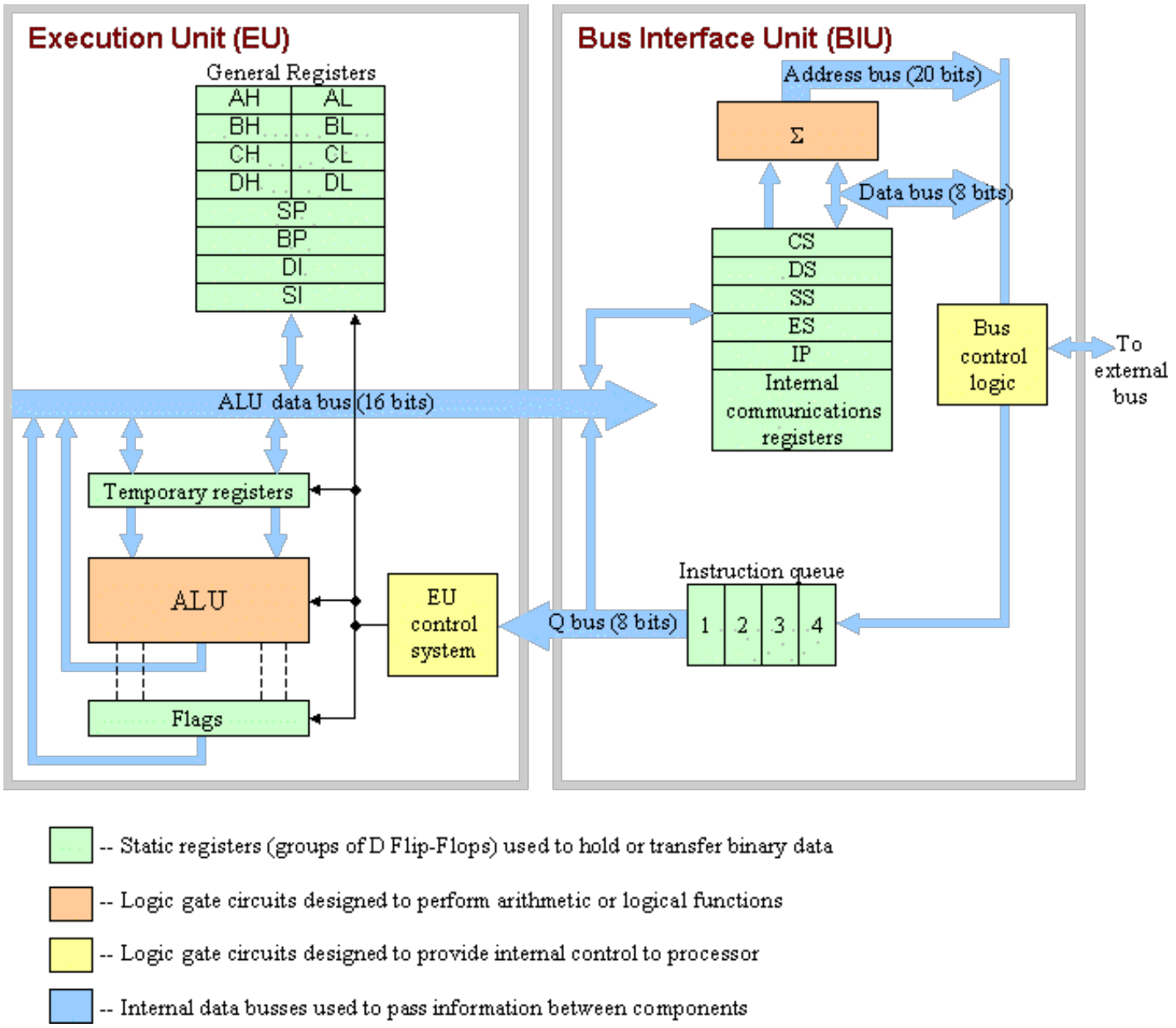
Structuur van de CPU

- registers
  - gegevens- en adresregisters
  - vlaggen
  - instructie/programma-teller
  - interne registers
- arithmetisch-logische eenheid = ALU
- besturingseenheid = control logic

CPU Organization

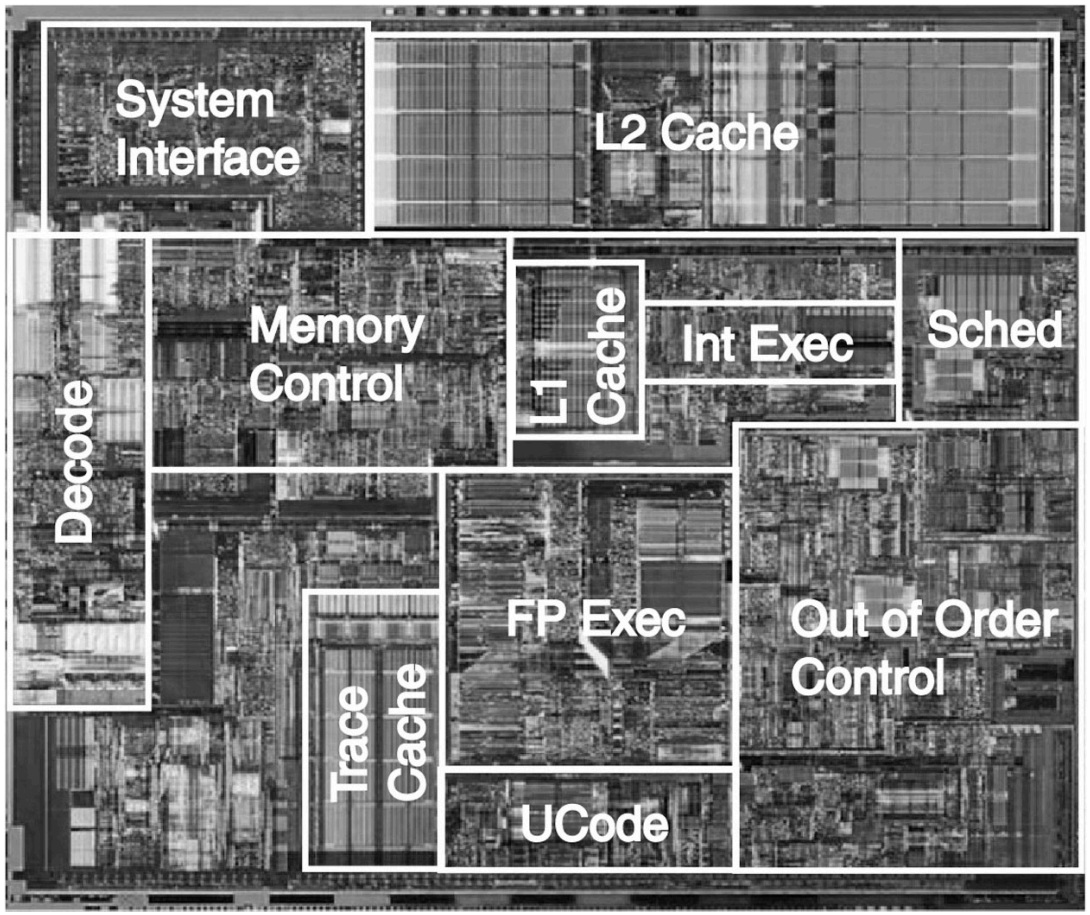


The data path of a typical Von Neumann machine.



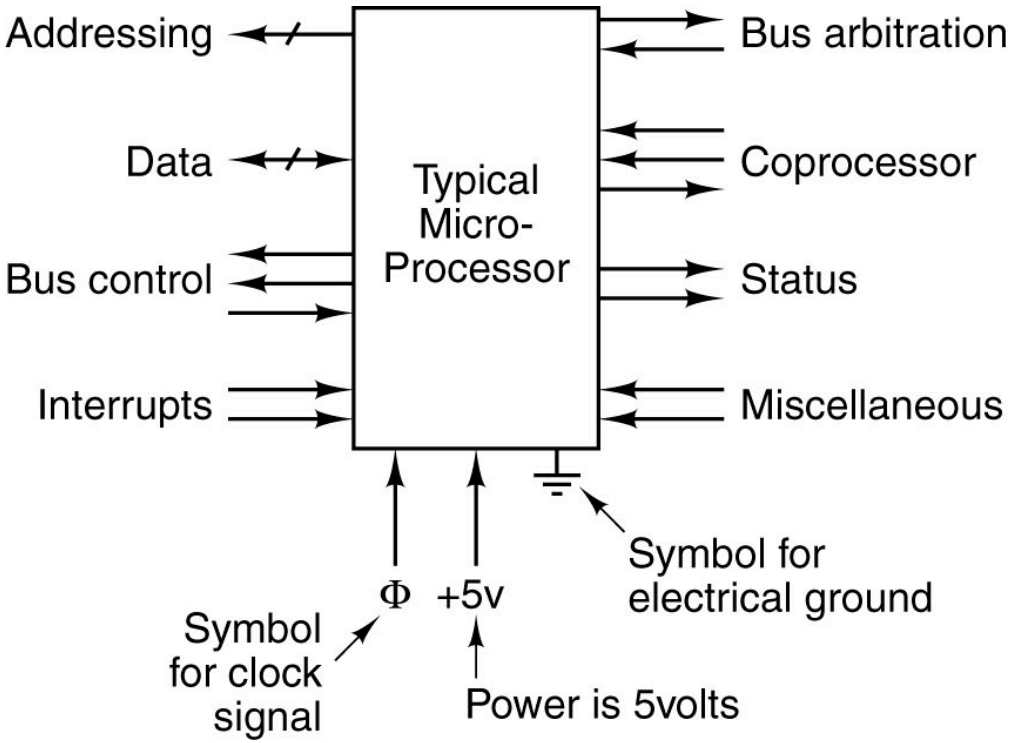
http://faculty.etsu.edu/tarnoff/ntes2150/uproc/blck8088.gif

Intel Computer Family (2)



The Pentium 4 chip. The photograph is copyrighted by the Intel Corporation, 2003 and is used by permission.

Aansluitingen van een CPU-chip

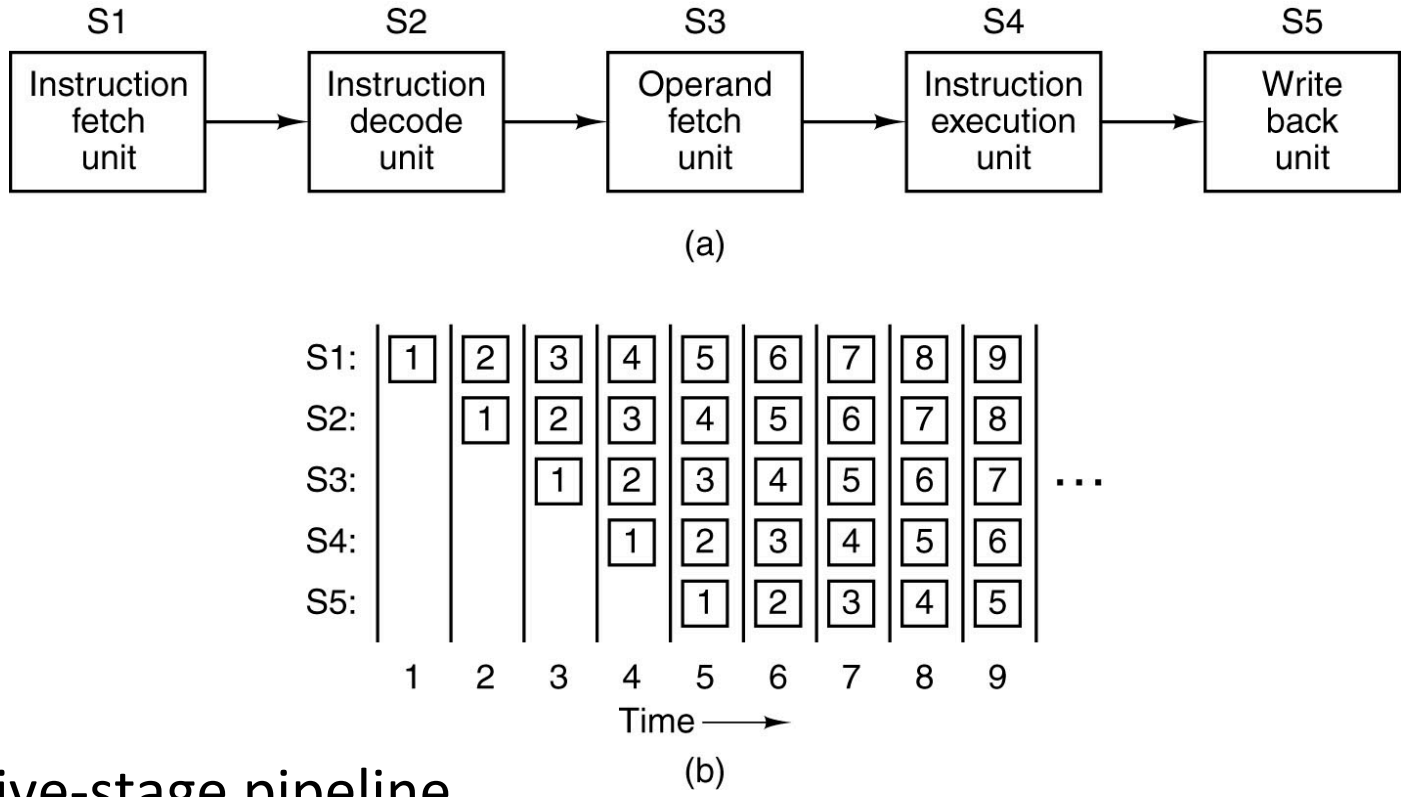


The logical pinout of a generic CPU.
The arrows indicate input signals and output signals.
The short diagonal lines indicate that multiple pins are used.

Gedrag van de CPU

- programma executeren
  - ① instructie lezen (adres: programmateller)
  - ② programmateller ophogen
  - ③ instructie decoderen
  - ④ instructie uitvoeren
- in huidige processoren: parallel = pipelined

Instruction-Level Parallelism



- A five-stage pipeline
- The state of each stage as a function of time.
- Nine clock cycles are illustrated

Soorten instructies

- gegevenstransport
- berekenen
- programmaflow

Gegevenstransport

- voorbeeldinstructies in x86-machinecode en -assembly:

- A10D00 MOV EAX,13
  zet de inhoud van register EAX op 13
- 87F3 XCHG EBX,ESI
  verwissel de inhoud van registers EBX en ESI

Berekenen

- bereken en schrijf het resultaat in register
- vlaggen (zero, sign, carry, overflow) zetten: afhankelijk van resultaat

- 00E3 ADD BL,AH
  tel AH bij BL op
- F6E1 MUL CL
  vermenigvuldig AL met CL en sla het resultaat in EAX op
- 09D2 OR EDX,EDX
  bereken bit-or van EDX met zichzelf, sla resultaat in EDX op

accumulator = impliciet register

EDX blijft gelijk, vlaggen veranderen

Programmaflow

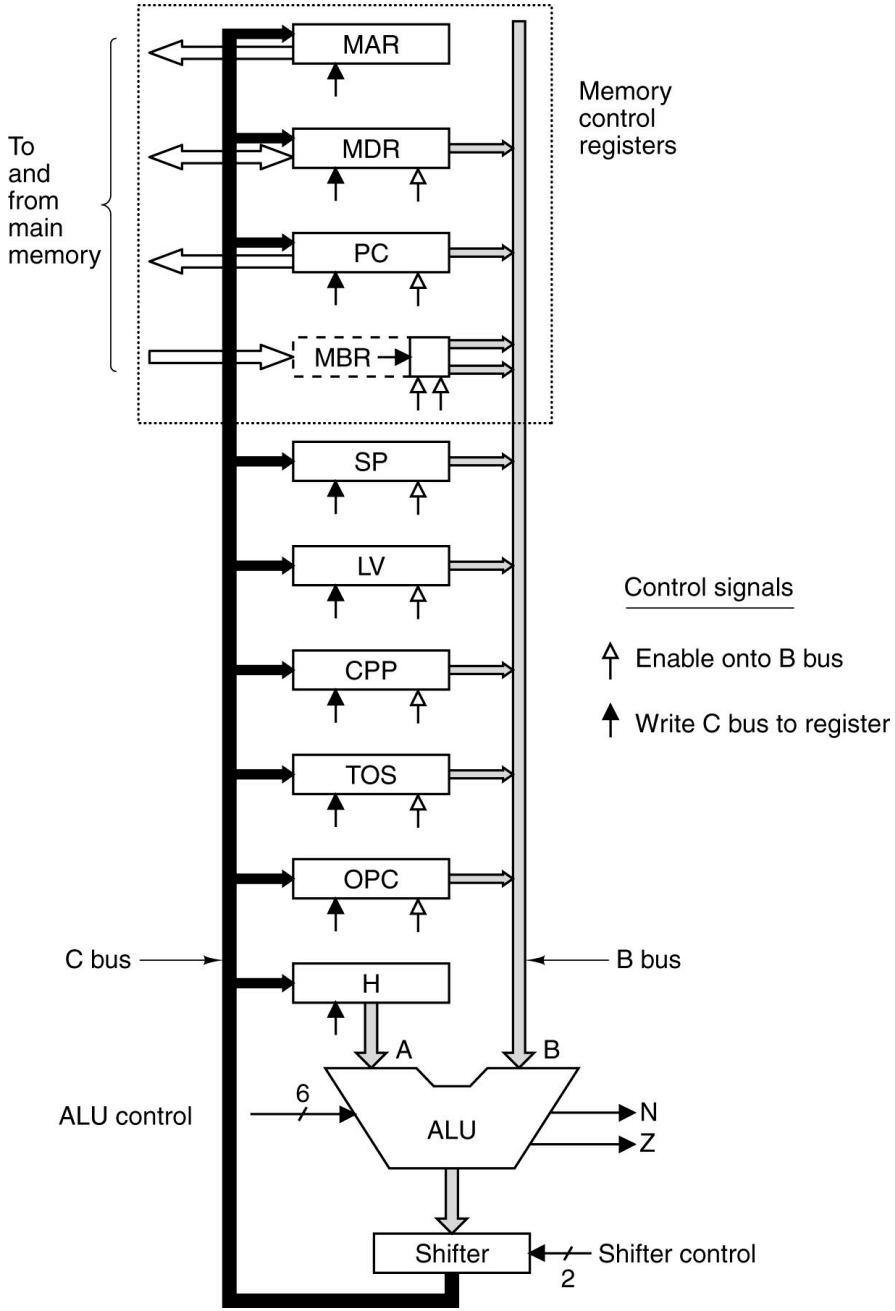
- normaal:
  instructies staan op volgorde in het geheugen
- programmaflow-instructies geven afwijkende volgorde aan

- E9FE0F JMP (EIP+0ffeh)
  tel 0FFE hexadecimaal bij de instructiepointer op
- 75EC JNZ (EIP–20)
  als de ZERO-vlag 0 is, trek 20 van de instructiepointer af

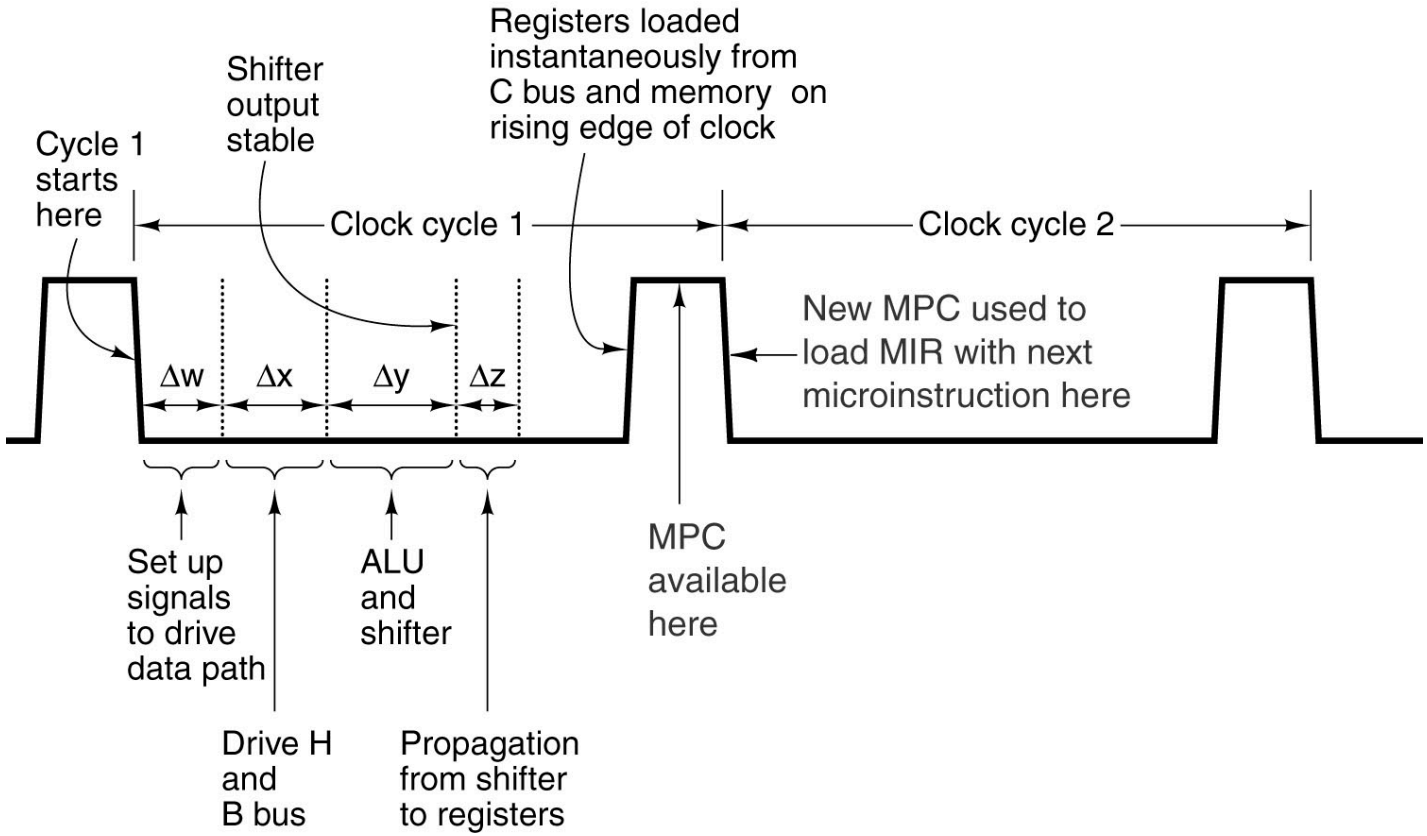
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	ADD Eb, Gb	ADD Ev, Gv	ADD Gb, Eb	ADD Gv, Ev	ADD AL, b	ADD AX, Iv	PUSH ES	POP ES	OR Eb, Gb	OR Ev, Gv	OR Gb, Eb	OR Gv, Ev	OR AL, Ib	OR AX, Iv	PUSH CS	--
1	ADC Eb, Gb	ADC Ev, Gv	ADC Gb, Eb	ADC Gv, Ev	ADC AL, Ib	ADC AX, Iv	PUSH SS	POP SS	SBB Eb, Gb	SBB Ev, Gv	SBB Gb, Eb	SBB Gv, Ev	SBB AL, Ib	SBB AX, Iv	PUSH DS	POP DS
2	AND Eb, Gb	AND Ev, Gv	AND Gb, Eb	AND Gv, Ev	AND AL, Ib	AND AX, Iv	ES:	DAA	SUB Eb, Gb	SUB Ev, Gv	SUB Gb, Eb	SUB Gv, Ev	SUB AL, Ib	SUB AX, Iv	CS:	DAS
3	XOR Eb, Gb	XOR Ev, Gv	XOR Gb, Eb	XOR Gv, Ev	XOR AL, Ib	XOR AX, Iv	SS:	AAA	CMP Eb, Gb	CMP Ev, Gv	CMP Gb, Eb	CMP Gv, Ev	CMP AL, Ib	CMP AX, Iv	DS:	AAS
4	INC AX	INC CX	INC DX	INC BX	INC SP	INC BP	INC SI	INC DI	DEC AX	DEC CX	DEC DX	DEC BX	DEC SP	DEC BP	DEC SI	DEC DI
5	PUSH AX	PUSH CX	PUSH DX	PUSH BX	PUSH SP	PUSH BP	PUSH SI	PUSH DI	POP AX	POP CX	POP DX	POP BX	POP SP	POP BP	POP SI	POP DI
6	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
7	JO Jb	JNO Jb	JB Jb	JNB Jb	JZ Jb	JNZ Jb	JBE Jb	JA Jb	JS Jb	JNS Jb	JPE Jb	JPO Jb	JL Jb	JGE Jb	JLE Jb	JG Jb
8	GRP1 Eb, Ib	GRP1 Ev, Iv	GRP1 Eb, Ib	GRP1 Ev, Ib	TEST Gb, Eb	TEST Gv, Ev	XCHG Gb, Eb	XCHG Gv, Ev	MOV Eb, Gb	MOV Ev, Gv	MOV Gb, Eb	MOV Gv, Ev	MOV Ew, Sw	LEA Gv, M	MOV Sw, Ew	POP Ev
9	NOP	XCHG CX, AX	XCHG DX, AX	XCHG BX, AX	XCHG SP, AX	XCHG BP, AX	XCHG SI, AX	XCHG DI, AX	CBW	CWD	CALL Ap	WAIT	PUSHF	POPF	SAHF	LAHF
A	MOV AL, Ob	MOV AX, Ov	MOV Ob, AL	MOV Ov, AX	MOVSB	MOVSW	CMPSB	CMPSW	TEST AL, Ib	TEST AX, Iv	STOSB	STOSW	LODSB	LODSW	SCASB	SCASW
B	MOV AL, Ib	MOV CL, Ib	MOV DL, Ib	MOV BL, Ib	MOV AH, Ib	MOV CH, Ib	MOV DH, Ib	MOV BH, Ib	MOV AX, Iv	MOV CX, Iv	MOV DX, Iv	MOV BX, Iv	MOV SP, Iv	MOV BP, Iv	MOV SI, Iv	MOV DI, Iv
C	--	--	RET Iw	RET	LES Gv, Mp	LDS Gv, Mp	MOV Eb, Ib	MOV Ev, Iv	--	--	RETF Iw	RETF	INT 3	INT Ib	INTO	IRET
D	GRP2 Eb, 1	GRP2 Ev, 1	GRP2 Eb, CL	GRP2 Ev, CL	AAM I0	AAD I0	--	XLAT	--	--	--	--	--	--	--	--
E	LOOPNZ Jb	LOOPZ Jb	LOOP Jb	JCXZ Jb	IN AL, Ib	IN AX, Ib	OUT Ib, AL	OUT Ib, AX	CALL Jv	JMP Jv	JMP Ap	JMP Jb	IN AL, DX	IN AX, DX	OUT DX, AL	OUT DX, AX
F	LOCK	--	REPNZ	REPZ	HLT	CMC	GRP3a Eb	GRP3b Ev	CLC	STC	CLI	STI	CLD	STD	GRP4 Eb	GRP5 Ev

The Data Path (1)

The data path of the example microarchitecture used in this chapter.



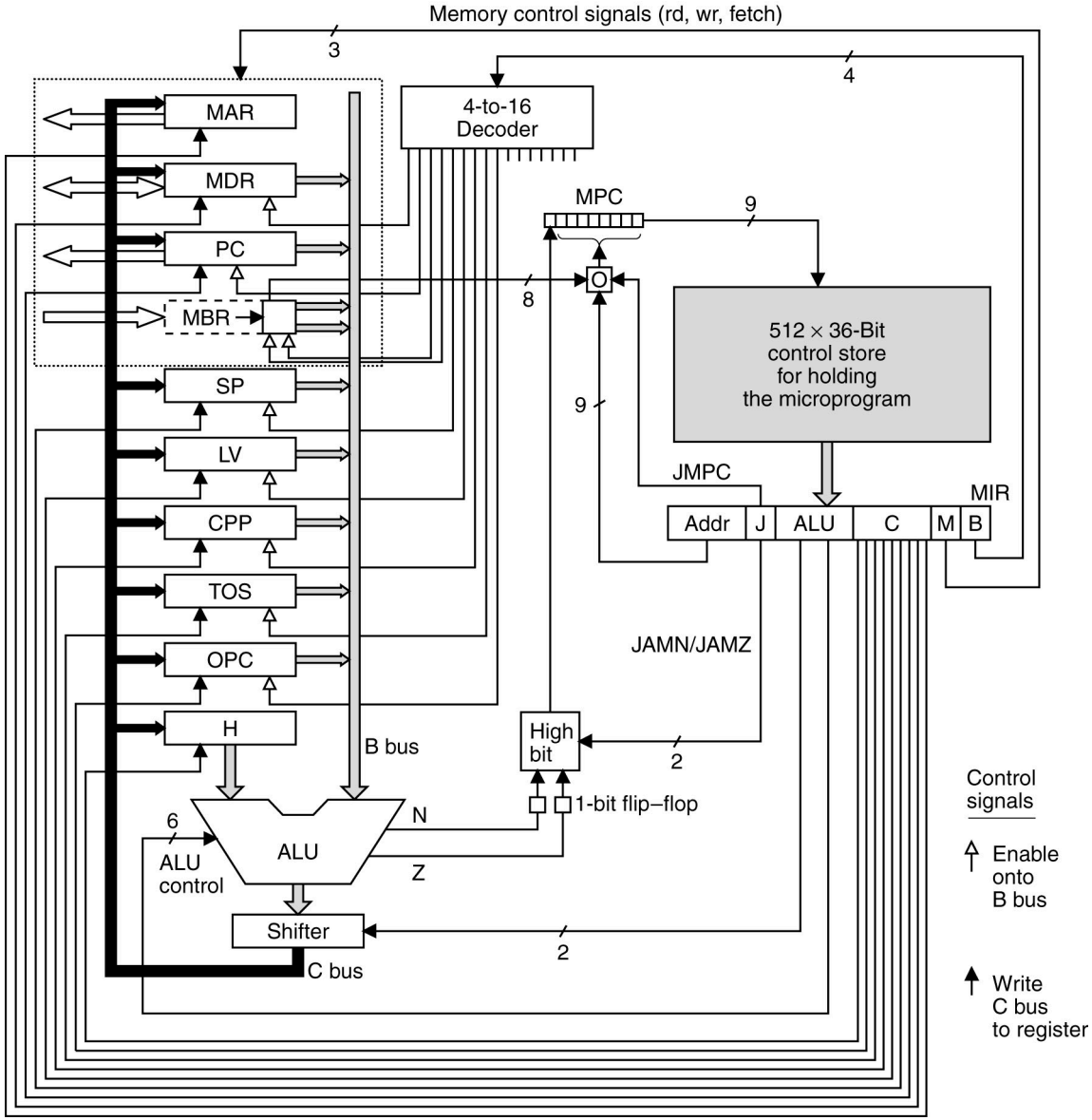
Data Path Timing

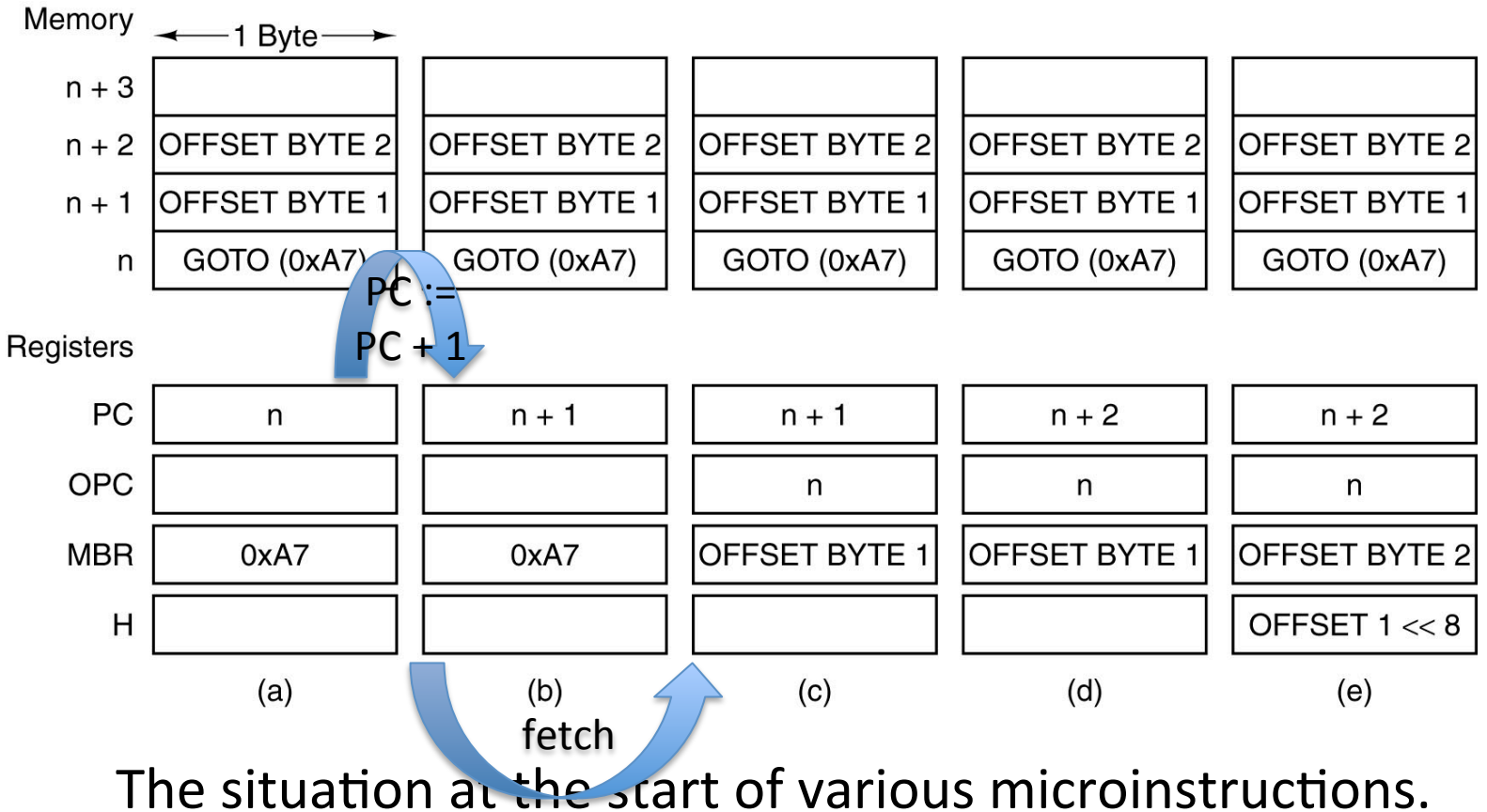


Timing diagram of one data path cycle.

Microinstruction Control: The Mic-1 (1)

The complete block diagram of our example microarchitecture, the Mic-1.





The situation at the start of various microinstructions.

(a) Main1. (b) goto1. (c) goto2. (d) goto3. (e) goto4.

Label	Operations	Comments
Main1	PC = PC + 1; fetch; goto (MBR)	MBR holds opcode; get next byte; dispatch
goto1	OPC = PC – 1	Save address of opcode.
goto2	PC = PC + 1; fetch	MBR = 1st byte of offset; fetch 2nd byte
goto3	H = MBR << 8	Shift and save signed first byte in H
goto4	H = MBRU OR H	H = 16-bit branch offset
goto5	PC = OPC + H; fetch	Add offset to OPC
goto6	goto Main1	Wait for fetch of next opcode

Samenvatting

- von-Neumann-architectuur
- structuur van CPU
- gedrag van CPU
- soorten instructies