



Escuela de Ingeniería

Microcontroladores y sistemas embebidos

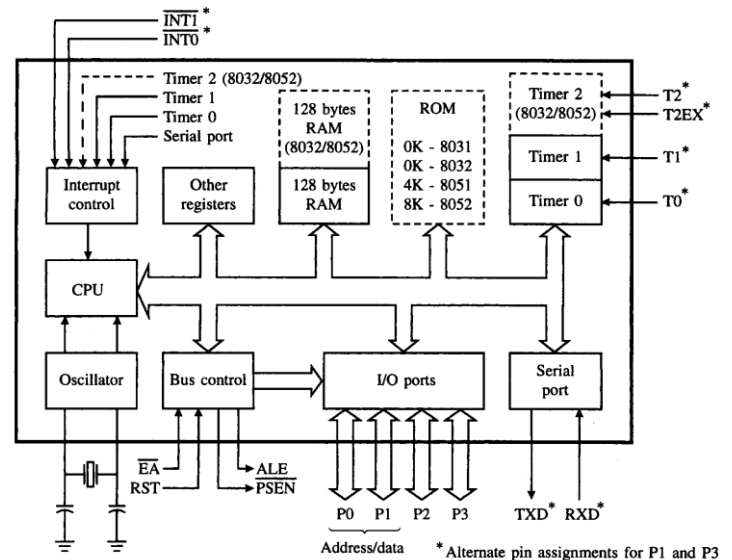
Manual del 8051

Presenta:

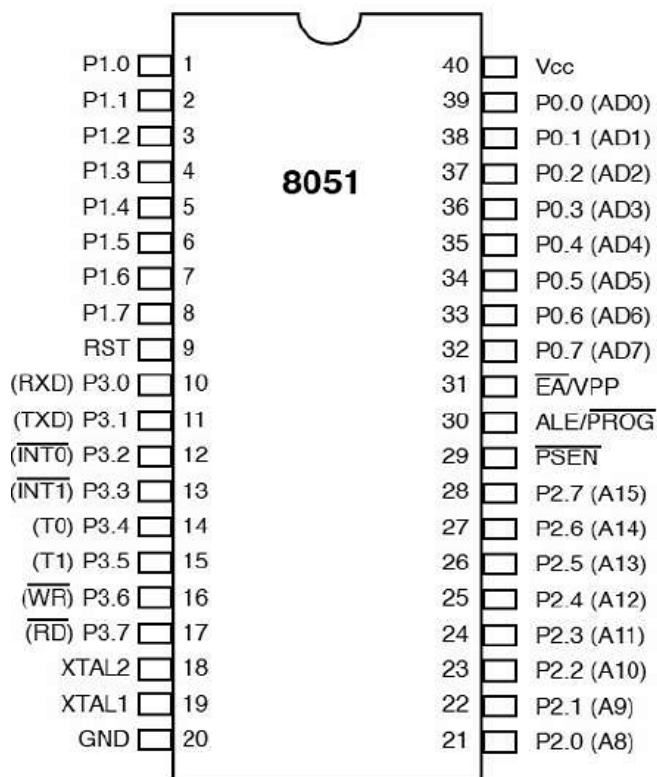
Jorge Tomás Araujo González

Tijuana, B. C., 5 de febrero de 2023

Arquitectura Interna



Arquitectura Externa



Información de los puertos

32 de los 40 pines funcionan como entrada / salida; 24 de ellos son de propósito dual.

Las 8 líneas de cada puerto pueden ser tratadas como unidad cuando se interactúa con dispositivos paralelos como impresoras. De igual manera, cada línea puede operar independientemente como un solo bit.

Puerto 0. Es un puerto de doble propósito en los pines 32-39. En diseños con componentes mínimos, es utilizado como puerto de entrada / salida de propósito general. Para diseños con memoria externa, se convierte en una dirección multiplexada y bus de datos.

Puerto 1. Es un puerto dedicado a entradas / salidas en los pines 1-8. Los pines están disponibles para interactuar con dispositivos externos como sea requerido.

Puerto 2. (pines 21-28) puerto de doble propósito, que funciona como entrada / salida, o como el byte mayor del bus de direcciones para diseños con código de memoria externa o más de 256 bytes de datos de memoria externa.

Puerto 3. Es un puerto de doble propósito (pines 10-17). Puede funcionar como entrada / salida y otras funciones presentadas en la tabla 1.

Bit	Name	Bit Address	Alternate Function
P3.0	RXD	B0H	Receive data for serial port
P3.1	TXD	B1H	Transmit data for serial port
P3.2	INT0	B2H	External interrupt 0
P3.3	INT1	B3H	External interrupt 1
P3.4	T0	B4H	Timer/counter 0 external input
P3.5	T1	B5H	Timer/counter 1 external input
P3.6	WR	B6H	External data memory write strobe
P3.7	RD	B7H	External data memory read strobe
P1.0	T2	90H	Timer/counter 2 external input
P1.1	T2EX	91H	Timer/counter 2 capture/reload

Tabla 1. Funciones alternas de los puertos 1 y 3.

PSEN (Program Store Enable). Es una señal de salida en el pin 29. Esta señal habilita el programa de una memoria externa. Usualmente se conecta al pin Output Enable (OE) de la EPROM para permitir la lectura de los bytes del programa.

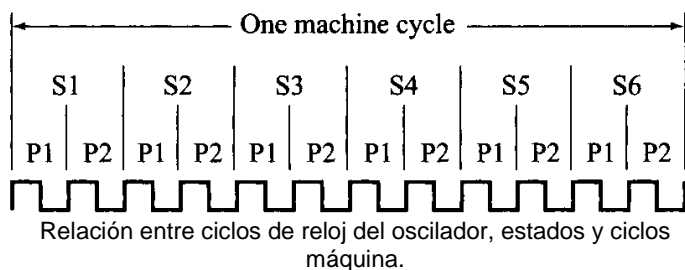
ALE (Address Latch Enable). Señal de salida en el pin 30 para demultiplexado de direcciones y bus de datos. Cuando el puerto 0 está en su modo alternado, ALE es la señal que hace un latch de la dirección a un registro externo durante la primera mitad del ciclo de memoria.

EA (External Access). Esta señal (pin 31), generalmente se encuentra en estado alto o bajo. Si está en alto, el 8051 ejecuta programas de la ROM interna cuando se está ejecutando en la parte inferior 4K/8K de la memoria. Si está en bajo, los programas solo se ejecutan de la memoria externa. Si EA se encuentra en estado bajo, la ROM interna se deshabilita y los programas se ejecutan de la EPROM externa.

RST (Reset). La entrada de pulso de reset se encuentra en el pin 9. Cuando la señal se torna en alto por al menos dos ciclos máquina, los registros internos se cargan con valores apropiados para volver a iniciar el sistema. Para funcionamiento normal, RST se está en bajo.

Entradas de oscilador. Las entradas 18 y 19 son para conectar un cristal oscilador de 12MHz y se necesitan capacitores estabilizadores. El reloj interno no necesariamente tiene que ser controlado por un cristal, se puede conectar una fuente de reloj TTL.

Conexiones de alimentación. El 8051 opera con tan solo 5V. La conexión de Vcc se encuentra en el pin 40, y tierra en el pin 20.



Estructura de Memoria del 8051

Byte address	Bit address	Byte address	Bit address
7F	General purpose RAM	FF	
		F0	F7 F6 F5 F4 F3 F2 F1 F0 B
		E0	E7 E6 E5 E4 E3 E2 E1 E0 ACC
		D0	D7 D6 D5 D4 D3 D2 D1 D0 PSW
		B8	- - - BC BB BA B9 B8 IP
		B0	B7 B6 B5 B4 B3 B2 B1 B0 P3
		A8	AF - - AC AB AA A9 A8 IE
		A0	A7 A6 A5 A4 A3 A2 A1 A0 P2
30		99	not bit addressable SBUF
2F		98	9F 9E 9D 9C 9B 9A 99 98 SCON
2E		90	97 96 95 94 93 92 91 90 P1
2D		8D	not bit addressable TH1
2C		8C	not bit addressable TH0
2B		8B	not bit addressable TL1
2A		8A	not bit addressable TL0
29		89	not bit addressable TMOD
28		88	8F 8E 8D 8C 8B 8A 89 88 TCON
27		87	not bit addressable PCON
26		83	not bit addressable DPH
25		82	not bit addressable DPL
24		81	not bit addressable SP
23		80	87 86 85 84 83 82 81 80 P0
22			
21			
20			
1F	Bank 3		
1E	Bank 2		
1D			
1C			
1B	Bank 1		
1A			
19			
18	Default register bank for R0-R7		
17			
16			
15			
14			
13			
12			
11			
10			
0F			
0E			
0D			
0C			
0B			
0A			
09			
08			
07			
06			
05			
04			
03			
02			
01			
00			

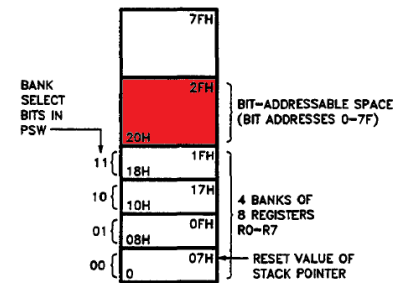
RAM

SPECIAL FUNCTION REGISTERS

Mapeo de memoria

El 8051 tiene un mapeo de memoria de 64 KB, pero solo 32 KB son accesibles debido a limitaciones en su arquitectura. La memoria se divide en varios bloques, cada uno con un propósito específico:

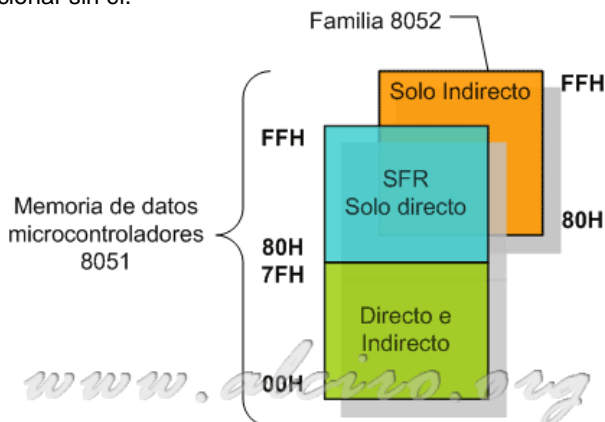
- 0-7Fh (128 bytes): Registros de sistema y de control del microcontrolador
- 80h-FFh (128 bytes): Registros de usuario y RAM
- 100h-1FFh (256 bytes): RAM extendida
- 2000h-FFFFh (49664 bytes): Memoria externa



El registro de sistema incluye registros como A (registro acumulador), B, R0-R7 (registros generales), DPTR (Dirección de Puntero de Datos), PC (Contador de Programa), PSW (Palabra de Estado) y otros.

El registro de usuario y la RAM se utilizan para almacenar variables y datos intermedios durante la ejecución del programa. La RAM extendida permite al microcontrolador acceder a una cantidad adicional de 256 bytes de memoria.

La memoria externa se utiliza para almacenar programas y datos más grandes. Este bloque de memoria se accede a través del registro DPTR y es opcional, ya que el microcontrolador puede funcionar sin él.



Set de instrucciones

Mnemonic	Instruction	Description	Addressing Mode	# of Bytes	# of Cycles
MOV	A, #Data	$A \leftarrow \text{Data}$	Immediate	2	1
	A, Rn	$A \leftarrow R_n$	Register	1	1
	A, Direct	$A \leftarrow (\text{Direct})$	Direct	2	1
	A, @Ri	$A \leftarrow @R_i$	Indirect	1	1
	Rn, #Data	$R_n \leftarrow \text{data}$	Immediate	2	1
	Rn, A	$R_n \leftarrow A$	Register	1	1
	Rn, Direct	$R_n \leftarrow (\text{Direct})$	Direct	2	2
	Direct, A	$(\text{Direct}) \leftarrow A$	Direct	2	1
	Direct, Rn	$(\text{Direct}) \leftarrow R_n$	Direct	2	2
	Direct1, Direct2	$(\text{Direct1}) \leftarrow (\text{Direct2})$	Direct	3	2
	Direct, @Ri	$(\text{Direct}) \leftarrow @R_i$	Indirect	2	2
	Direct, #Data	$(\text{Direct}) \leftarrow \text{#Data}$	Direct	3	2
	@Ri, A	$@R_i \leftarrow A$	Indirect	1	1
	@Ri, Direct	$@R_i \leftarrow \text{Direct}$	Indirect	2	2
	@Ri, #Data	$@R_i \leftarrow \text{#Data}$	Indirect	2	1
	DPTR, #Data16	$\text{DPTR} \leftarrow \text{#Data16}$	Immediate	3	2
MOVC	A, @A+DPTR	$A \leftarrow \text{Code Pointed by A+DPTR}$	Indexed	1	2
	A, @A+PC	$A \leftarrow \text{Code Pointed by A+PC}$	Indexed	1	2
	A, @Ri	$A \leftarrow \text{Code Pointed by Ri (8-bit Addr)}$	Indirect	1	2
MOVB	A, @DPTR	$A \leftarrow \text{External Data Pointed by DPTR}$	Indirect	1	2
	@Ri, A	$@R_i \leftarrow A \text{ (External Data 8-bit Addr)}$	Indirect	1	2
	@DPTR, A	$@\text{DPTR} \leftarrow A \text{ (External Data 16-bit Addr)}$	Indirect	1	2
PUSH	Direct	$\text{Stack Pointer SP} \leftarrow (\text{Direct})$	Direct	2	2
POP	Direct	$(\text{Direct}) \leftarrow \text{Stack Pointer SP}$	Direct	2	2
XCH	Rn	Exchange ACC with Rn	Register	1	1
	Direct	Exchange ACC with Direct Byte	Direct	2	1
	@Ri	Exchange ACC with Indirect RAM	Indirect	1	1
XCHD	A, @Ri	Exchange ACC with Lower Order Indirect RAM	Indirect	1	1

Mnemonic	Instruction	Description	Addressing Mode	# of Bytes	# of Cycles
ADD	A, #Data	$A \leftarrow A + \text{Data}$	Immediate	2	1
	A, Rn	$A \leftarrow A + Rn$	Register	1	1
	A, Direct	$A \leftarrow A + (\text{Direct})$	Direct	2	1
	A, @Ri	$A \leftarrow A + @Ri$	Indirect	1	1
ADDC	A, #Data	$A \leftarrow A + \text{Data} + C$	Immediate	2	1
	A, Rn	$A \leftarrow A + Rn + C$	Register	1	1
	A, Direct	$A \leftarrow A + (\text{Direct}) + C$	Direct	2	1
	A, @Ri	$A \leftarrow A + @Ri + C$	Indirect	1	1
SUBB	A, #Data	$A \leftarrow A - \text{Data} - C$	Immediate	2	1
	A, Rn	$A \leftarrow A - Rn - C$	Register	1	1
	A, Direct	$A \leftarrow A - (\text{Direct}) - C$	Direct	2	1
	A, @Ri	$A \leftarrow A - @Ri - C$	Indirect	1	1
MUL	AB	Multiply A with B ($A \leftarrow \text{Lower Byte of } A*B$ and $B \leftarrow \text{Higher Byte of } A*B$)	--	1	4
DIV	AB	Divide A by B ($A \leftarrow \text{Quotient}$ and $B \leftarrow \text{Remainder}$)	--	1	4
DEC	A	$A \leftarrow A - 1$	Register	1	1
	Rn	$Rn \leftarrow Rn - 1$	Register	1	1
	Direct	$(\text{Direct}) \leftarrow (\text{Direct}) - 1$	Direct	2	1
	@Ri	$@Ri \leftarrow @Ri - 1$	Indirect	1	1
INC	A	$A \leftarrow A + 1$	Register	1	1
	Rn	$Rn \leftarrow Rn + 1$	Register	1	1
	Direct	$(\text{Direct}) \leftarrow (\text{Direct}) + 1$	Direct	2	1
	@Ri	$@Ri \leftarrow @Ri + 1$	Indirect	1	1
	DPTR	$DPTR \leftarrow DPTR + 1$	Register	1	2
DA	A	Decimal Adjust Accumulator	--	1	1

Mnemonic	Instruction	Description	Addressing Mode	# of Bytes	# of Cycles
ANL	A, #Data	$A \leftarrow A \text{ AND } \text{Data}$	Immediate	2	1
	A, Rn	$A \leftarrow A \text{ AND } Rn$	Register	1	1
	A, Direct	$A \leftarrow A \text{ AND } (\text{Direct})$	Direct	2	1
	A, @Ri	$A \leftarrow A \text{ AND } @Ri$	Indirect	1	1
	Direct, A	$(\text{Direct}) \leftarrow (\text{Direct}) \text{ AND } A$	Direct	2	1
	Direct, #Data	$(\text{Direct}) \leftarrow (\text{Direct}) \text{ AND } \text{#Data}$	Direct	3	2
ORL	A, #Data	$A \leftarrow A \text{ OR } \text{Data}$	Immediate	2	1
	A, Rn	$A \leftarrow A \text{ OR } Rn$	Register	1	1
	A, Direct	$A \leftarrow A \text{ OR } (\text{Direct})$	Direct	2	1
	A, @Ri	$A \leftarrow A \text{ OR } @Ri$	Indirect	1	1
	Direct, A	$(\text{Direct}) \leftarrow (\text{Direct}) \text{ OR } A$	Direct	2	1
	Direct, #Data	$(\text{Direct}) \leftarrow (\text{Direct}) \text{ OR } \text{#Data}$	Direct	3	2
XRL	A, #Data	$A \leftarrow A \text{ XRL } \text{Data}$	Immediate	2	1
	A, Rn	$A \leftarrow A \text{ XRL } Rn$	Register	1	1
	A, Direct	$A \leftarrow A \text{ XRL } (\text{Direct})$	Direct	2	1
	A, @Ri	$A \leftarrow A \text{ XRL } @Ri$	Indirect	1	1
	Direct, A	$(\text{Direct}) \leftarrow (\text{Direct}) \text{ XRL } A$	Direct	2	1
	Direct, #Data	$(\text{Direct}) \leftarrow (\text{Direct}) \text{ XRL } \text{#Data}$	Direct	3	2
CLR	A	$A \leftarrow 00H$	--	1	1
CPL	A	$A \leftarrow \overline{A}$	--	1	1
RL	A	Rotate ACC Left	--	1	1
RLC	A	Rotate ACC Left through Carry	--	1	1
RR	A	Rotate ACC Right	--	1	1
RRC	A	Rotate ACC Right through Carry	--	1	1
SWAP	A	Swap Nibbles within ACC	--	1	1

Mnemonic	Instruction	Description	# of Bytes	# of Cycles
CLR	C	$C \leftarrow 0$ (C = Carry Bit)	1	1
	Bit	$\text{Bit} \leftarrow 0$ (Bit = Direct Bit)	2	1
SET	C	$C \leftarrow 1$	1	1
	Bit	$\text{Bit} \leftarrow 1$	2	1
CPL	C	$C \leftarrow \overline{C}$	1	1
	Bit	$\text{Bit} \leftarrow \overline{\text{Bit}}$	2	1
ANL	C, /Bit	$C \leftarrow C \cdot \overline{\text{Bit}}$ (AND)	2	1
	C, Bit	$C \leftarrow C \cdot \text{Bit}$ (AND)	2	1
ORL	C, /Bit	$C \leftarrow C + \overline{\text{Bit}}$ (OR)	2	1
	C, Bit	$C \leftarrow C + \text{Bit}$ (OR)	2	1
MOV	C, Bit	$C \leftarrow \text{Bit}$	2	1
	Bit, C	$\text{Bit} \leftarrow C$	2	2
JC	rel	Jump is Carry (C) is Set	2	2
JNC	rel	Jump is Carry (C) is Not Set	2	2
JB	Bit, rel	Jump is Direct Bit is Set	3	2
JNB	Bit, rel	Jump is Direct Bit is Not Set	3	2
JBC	Bit, rel	Jump is Direct Bit is Set and Clear Bit	3	2

Mnemonic	Instruction	Description	# of Bytes	# of Cycles
ACALL	ADDR11	Absolute Subroutine Call $PC + 2 \rightarrow (SP); \text{ADDR11} \rightarrow PC$	2	2
LCALL	ADDR16	Long Subroutine Call $PC + 3 \rightarrow (SP); \text{ADDR16} \rightarrow PC$	3	2
RET	--	Return from Subroutine $(SP) \rightarrow PC$	1	2
RETI	--	Return from Interrupt	1	2
AJMP	ADDR11	Absolute Jump $\text{ADDR11} \rightarrow PC$	2	2
LJMP	ADDR16	Long Jump $\text{ADDR16} \rightarrow PC$	3	2
SJMP	rel	Short Jump $PC + 2 + \text{rel} \rightarrow PC$	2	2
JMP	@A + DPTR	$A + \text{DPTR} \rightarrow PC$	1	2
JZ	rel	If A=0, Jump to PC + rel	2	2
JNZ	rel	If A ≠ 0, Jump to PC + rel		
CJNE	A, Direct, rel	Compare (Direct) with A. Jump to PC + rel if not equal	3	2
	A, #Data, rel	Compare #Data with A. Jump to PC + rel if not equal	3	2
	Rn, #Data, rel	Compare #Data with Rn. Jump to PC + rel if not equal	3	2
	@Ri, #Data, rel	Compare #Data with @Ri. Jump to PC + rel if not equal	3	2
DJNZ	Rn, rel	Decrement Rn. Jump to PC + rel if not zero	2	2
	Direct, rel	Decrement (Direct). Jump to PC + rel if not zero	3	2
NOP		No Operation	1	1

Modos de direccionamiento

El modo de direccionamiento es una combinación de direccionamiento directo e indirecto.

El direccionamiento directo implica que la dirección de la memoria se especifica explícitamente en el código. Por ejemplo, el código `MOV A, #34` significa que se mueve el valor 34 al registro A.

Por otro lado, el direccionamiento indirecto implica que la dirección de la memoria se almacena en un registro, y se utiliza ese registro para acceder a los datos. Por ejemplo, el código `MOV R0, #30` significa que se mueve el valor 30 al registro R0. Después, el código `MOV A, @R0` significa que se mueve el valor almacenado en la dirección apuntada por R0 al registro A.

El 8051 también tiene un registro DPTR (Dirección de Puntero de Datos) que se utiliza para acceder a datos en la memoria externa. Este registro es de 16 bits, lo que permite acceder a direcciones de memoria de hasta 65.536 bytes.

En resumen, el modo de direccionamiento del 8051 es una combinación de direccionamiento directo, indirecto a través de los registros R0, R1 y R2, y direccionamiento indirecto a través del registro DPTR. Esto permite al 8051 acceder a la memoria de forma flexible y eficiente, lo que es esencial para muchas aplicaciones.

- **Immediate Addressing (MOV A, #20H)**
- **Direct Addressing (MOV A, 30H)**
- **Indirect Addressing (MOV A, @R0)**
- **External Direct Addressing (MOVX A, @DPTR)**
- **Code Indirect Addressing (MOV C A, @A+DPTR)**

Funciones principales del 8051

- Interrupciones externas y temporizador/contador: el 8051 es capaz de responder a interrupciones externas, así como a interrupciones generadas por un temporizador/contador interno.
- Comunicación serial: el 8051 es capaz de comunicarse con otros dispositivos mediante una interfaz de comunicación serial, como, por ejemplo, el protocolo RS-232.
- Memoria: el 8051 cuenta con una memoria RAM de 128 bytes y una memoria ROM de 4Kb.
- Procesamiento de datos: el 8051 es capaz de realizar operaciones aritméticas básicas, lógicas y de desplazamiento.
- Puertos paralelos: el 8051 cuenta con dos puertos paralelos programables para la conexión de periféricos externos.

Registros

Los registros del 8051 son nombres cortos para almacenar valores en un microcontrolador 8051. Estos registros sirven para guardar datos y direcciones. Están compuestos por 8 registros de 8 bits: A, B, R0, R1, R2, R3, R4, R5, R6, R7 y 3 registros de 16 bits: DPTR, PC y SP. Estos registros están compuestos de bits, los cuales se usan para controlar variables, procesar datos y controlar la ejecución de instrucciones.

Comunicación

El 8051 es un microcontrolador con una variedad de opciones de comunicación, como la comunicación serial, la comunicación paralela, la comunicación por infrarrojos, la comunicación por radiofrecuencia, la comunicación por Bluetooth, la comunicación por USB y la comunicación por Ethernet. Estas opciones permiten que el 8051 se conecte a otros dispositivos para transferir y recibir datos.

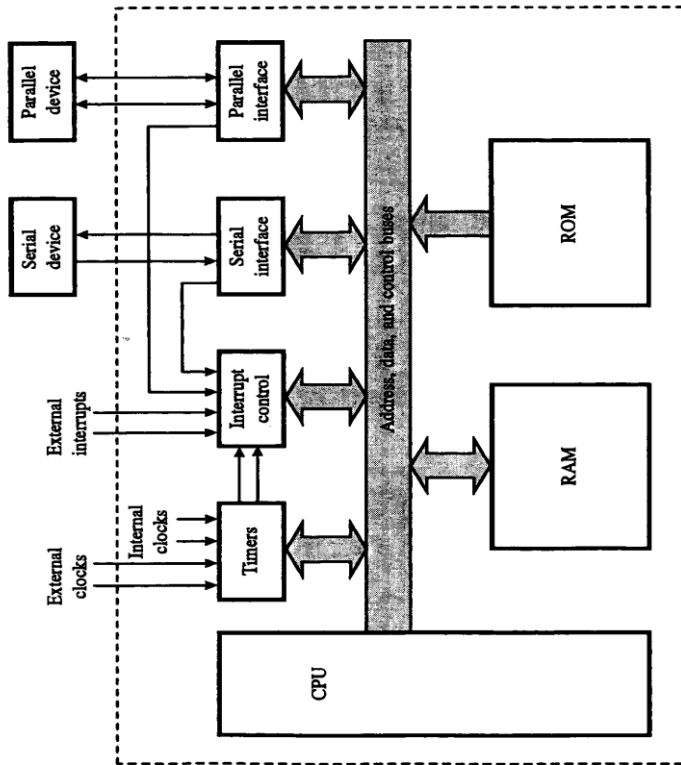


Diagrama de bloques de una microcomputadora

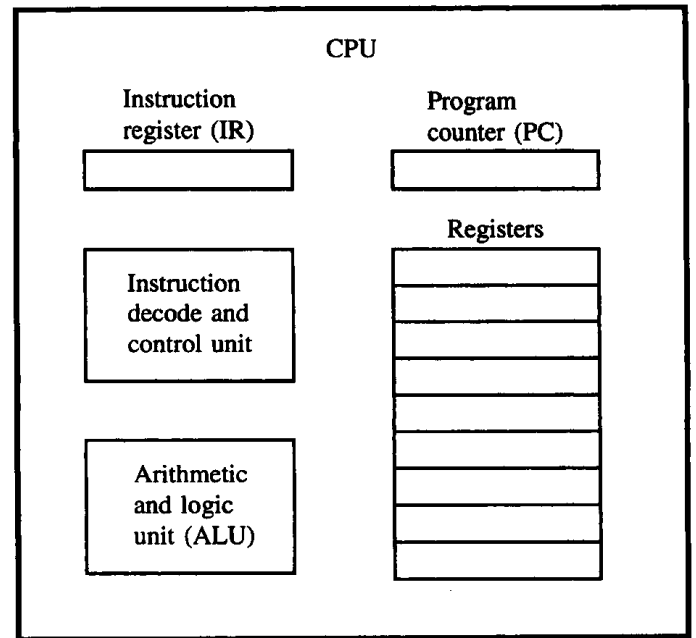
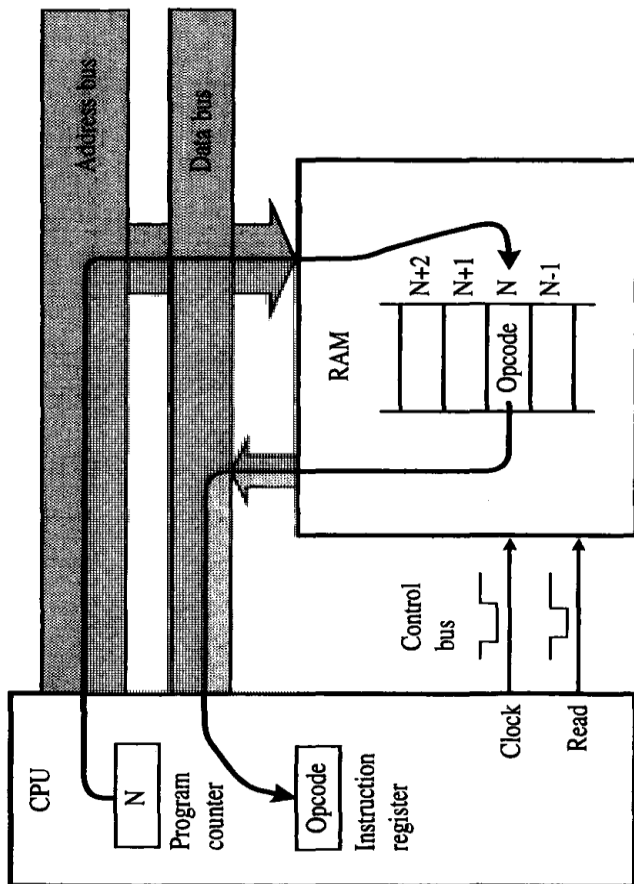


Diagrama del CPU



Actividad del bus en el ciclo fetch.

