

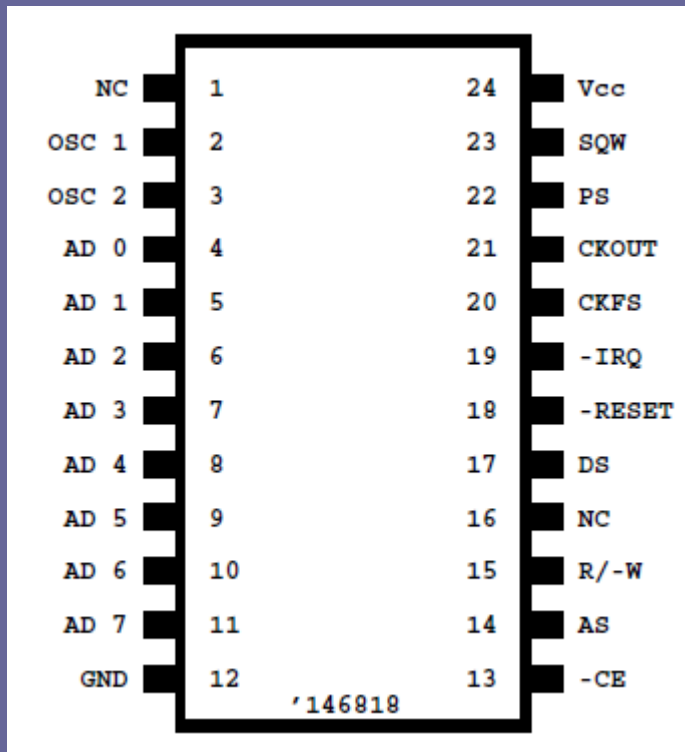
UNIVERSIDAD
NACIONAL DE
SAN MARTÍN



El reloj de Tiempo Real (RTC) de la PC

David López

Motorola MC 146818



Funciones

- Reloj
- Calendario
- Alarma
- Generador de onda cuadrada
- 64 bytes de RAM estática bajo consumo

Chip MC 146818
(Ver Apartado 12.12 apunte
Universo Digital p. 368)

Motorola MC146818

00	Segundos
01	Segundos Alarma
02	Minutos
03	Minutos alarma
04	Horas
05	Horas alarma
06	Dia de la semana
07	Dia del mes
08	Mes
09	Año
0A	Registro A
0B	Registro B
0C	Registro C
0D	Registro D
0E..3F	50 bytes libres

64 bytes de RAM:

10 para fecha y hora

- 00 al 09

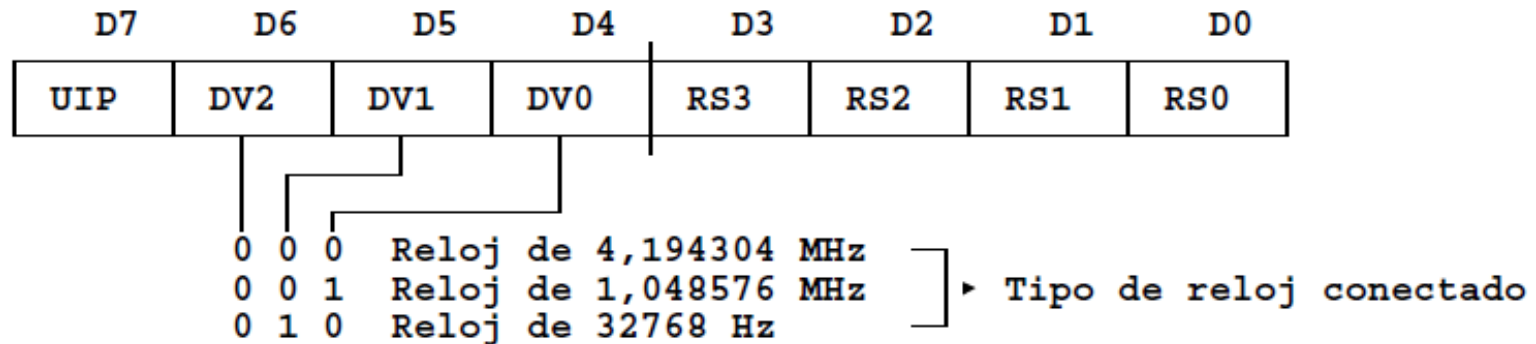
4 registros de configuración

- A, B, C, D

50 bytes disponibles para otros usos

- 0x32: primeras 2 cifras del año
- 49 disponibles para otros usos

Registro A



- Es de lectura/escritura excepto el bit UIP
 - UIP: Indica si se está actualizando el reloj
 - Si está en 0 tenemos al menos 244μs para trabajar
 - Si está en 1 hay que esperar 1984μs

Registro B

D7	D6	D5	D4	D3	D2	D1	D0
SET	PIE	AIE	UIE	SQWE	DM	24/12	DSE

- Es de lectura/escritura
- Se usa para configurar el chip

Registro C

D7	D6	D5	D4	D3	D2	D1	D0
IRQF	PF	AF	UF	0	0	0	0

- Es de sólo lectura
- Se borra al leerlo
- Contiene 4 flags que indican si se generaron interrupciones

Registro D

D7	D6	D5	D4	D3	D2	D1	D0
VRT	0	0	0	0	0	0	0

- Es de sólo lectura
- Contiene 1 flag que indica si la hora es válida (no hubo fallo en alimentación)

E/S mediante ports en Linux

- Se utilizan 2 ports para leer o escribir la memoria del chip
 - 0x70: Se coloca el nro. de posición a leer/escribir
 - 0x71: Se lee o escribe el valor

Bibliotecas y funciones

● **unistd.h**

- `usleep (unsigned long usecs)`
 - Interrumpe por al menos 10ms aprox. porque cambia de contexto.
- Para delays más chicos y precisos existe `nanosleep()`

● **sys/io.h – unistd.h según versión de libc**

- `int ioperm(unsigned long from, unsigned long num, int turn_on)`
- `unsigned char inb(unsigned short int port)`
- `void outb(unsigned char value, unsigned short int port)`