### КАНАЛИ І ПОРТИ

**Канали та порти вводу-виводу**

**Канали** застосовуються для підключення к ПК периферійних пристроїв: це **фізичні лінії**. Є паралельні і послідовні канали (розглядаються далі окремо).

**Порти** – уніфіковані засоби для забезпечення взаємодії з периферійними пристроями. Для усіх ресурсів виділяються свої порти – **логічні адреси вводу-виводу**, за допомогою яких здійснюється передача даних і управлінь до периферійних пристроїв. Прийом фізичних кодів від клавіатури здійснюється через порт 96 (60h), управління динаміком - 97 (61h) і т.і.

Порти з номерами 0-FFh призначені для материнської плати, номери 100h-3FFh декодуються на шине вводу-виводу для використання різними адаптерами. Порти з номерами 400h і вище – недоступні.

|  |  |
| --- | --- |
| **Порти** | **Призначення** |
| 000-01F | Порти контролера прямого доступа в пам’ять |
| 020-03F | Порти 1 контролера обробки АП |
| 040-05F | Порти таймера |
| 060-06F | Порти контролера клавіатури |
| 0A0-0BF | Порти 2 контролера обробки АП |
| 170-177 | Порти 2 ЖД |
| 1F0-1F7 | Порти 1 ЖД |
| 0F0-0FF | Порти сопроцесора |
| 378-37F | Порти паралельного каналу |
| 3F8-3FF | Порти послідовного каналу |

Уточнення щодо таблиці даних BIOS.

0:0400 – адреса порту СОМ1

0:0402 - адреса порту СОМ2

0:0404 - адреса порту СОМ3

0:0406 - адреса порту СОМ4

0:0408 - адреса порту LPT1

0:0410 – встановлення обладнання

0:0413 – обсяг ОП (в КБ)

0:041А - адреса початкового буферу клавіатури

0:041С - адреса кінцевого буферу клавіатури

0:041Е – (32б) – буфер клавіатури

0:0442 – (7б) – стан контролеру Накопичувача Гнучких МД

0:046C – (4б) лічильник імпульсів таймера

0:0472 – (2б) 1234h (Ctrl+Alt+Del)

Засоби для роботи з портами **1) out, in**

**2) outport, inport**

**outportb, inportb**

*Канали: послідовні і паралельні*

Паралельні: лінії даних, лінії управління

Послідовні: 1 лінія – асинхронні;

2 лінії - синхронні

Порти – логічні адреси ОП

*Структура пакета даних у послідовному каналі вводу-виводу.*

Старт біт

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | < ДАНІ > |  | 1 | 2 |

5 – 8 біт БП Стоп біти

контролер парності = біт парності (БП)

**Мікросхема CMOS-memory**

64(128)регістрів:

Порт 07016 – номер регістра CMOS

Порт 07116 – дані

Регістри:

0 – секунди

1 – сигнал секунд

2 – хвилини

3 – сигнал хвилин

4 - години

5 – сигнал годин

6 – день тижня

7 – день місяця

8 - місяць

9 - рік

А, В, С, Д – регістри статусу

7 біт(А) – виконується модифікація часу

4(В) = 1 – кінець модифікації IRQ8

5(В) = 1 – ALARM

2 біт(В) = 1 – час у 2-ій системі

0 – BCD

1 біт(В) = 1 – 24 год.

0 – 12 год.

ПРИКЛАД 1:

# include <stdio.h>

# include <dos.h>

void main()

{

outport b(0x70, 4);

printf(“Часы : % х \n”, inport b(0x71));

outport b(0x70, 2);

printf(“Минуты : % x\n”, inport b(0x71));

outport b(0x70, 0);

printf(“Секунды : % x\n”, inport b(0x71));

}

**Мікросхема паралельного інтерфейсу INTEL 8255**

3 одно-байтні регістри:

6016 – лише для читання

6116 – читається і записується

6216 – лише для читання

7 біт 6116 порту = 0 → у порт занесений скен-код натиснутої клавіші

= 1 → скен-код прийнятий і його можна обробляти

0 біт 6116 порту = 0 – м/с безпосередньо сама керує системним динаміком

1 біт = 1 – дифузор втягнутий

= 0 - відпущено

= 1 – динаміком керує 2-й канал таймеру

1 біт = 0 – нема управління (сигнали не пропускаються)

=1 – сигнали пропускаються на вихідній лінії

6116 – порт діагностики

6216 – порт в залежності від ПК

**ТАЙМЕР 8253/8254**

3 канали: 1, 2, 3 (з номерами 0,1,2): в кожному по 3 16-бітних регістрів

3 16-бітні регістри: регістр вводу - виводу

регістр буферний

регістр лічильник

3 лінії для кожного з каналів:

1 – вхідна лінія

2 – вхідна лінія управління, яка дозволяє або заперечує проходження імпульсів (канали 0, 1 завжди: так, 2 – 0 або 1 )

3 - вихідна

3 порти для роботи каналів:

0 – 4016 – визначення поточного часу (зберігається у Таблиці BIOS 0:046C)

1 – 4116 – для регенерації пам’яті

2 – 4216 – управління динаміком

Типовий генератор 1193180 імпульсів/сек.

Режими роботи: 0,4 – одноразове виконання

1,5 – робота з перезапуском (0 /1 на виході генерують старт)

2,3 – робота з автозавантаженням (новий старт автоматично)

Схема таймера зв’язана з командним 8-бітним регістром через порт 4316

***КОМАНДНИЙ РЕГІСТР 043h****:*

0 біт – 0 – відлік двійковий, 1 - BCD

1 – 3 – режим роботи таймеру (000 – 101,0 – 5)

4 – 5 – операція 00: – передати значення лічильника в буфер

01 – читати тільки молодший байт

10 - читати тільки старший байт

11 – завантаження молодшого, а потім старшого

6 – 7 – номер каналу (00 – 10 , 0 – 2 канали:

00=0, 01=1, 10=2)

B6h – типовий зміст командного регістра (10 11 011 0 – двійковий облік, реж. 3, завантаження мол. + старш., 2 канал)

ПРИКЛАД 2:

# include <conio.h>

extern int myasm1 (int);

void main()

{

int b; char c;

while ((c=getch())!=27)

{

b=c\*100;

zvuk(b);

}

}

.MODEL SMALL

.CODE

PUBLIC \_myasm1

\_myasm1 PROC

push bp

mov bp, sp

mov cx, [bp+4] ;аргумент→cx

pop bp

mov al, 0B6H

out 43H, al

mov ax, cx

out 42H, al

mov al, ah

out 42H, al

in al, 61H

mov ah, al

or al, 3 ; в порт 61h записуємо 11

out 61H, al

sub cx, cx ;обнулити сх

tm:

loop tm ; в момент СХ=0 – IRQ0

mov al, ah

out 61H, al

ret

\_myasm1 ENDP

END