



Лекции

Микроконтроллеры

периферия (дисплей)

2016

Дисплеи



Подключение WH1602

Pin No.	Symbol	Level	Description
1	V _{SS}	0V	Ground
2	V _{DD}	5.0V	Supply Voltage for logic
3	VO	(Variable)	Operating voltage for LCD
4	RS	H/L	H: DATA, L: Instruction code
5	R/W	H/L	H: Read(MPU→ Module) L: Write(MPU→ Module)
6	E	H,H→ L	Chip enable signal
7	DB0	H/L	Data bit 0
8	DB1	H/L	Data bit 1
9	DB2	H/L	Data bit 2
10	DB3	H/L	Data bit 3
11	DB4	H/L	Data bit 4
12	DB5	H/L	Data bit 5
13	DB6	H/L	Data bit 6
14	DB7	H/L	Data bit 7
15	A	-	LED +
16	K	-	LED -

Подключение MT1602

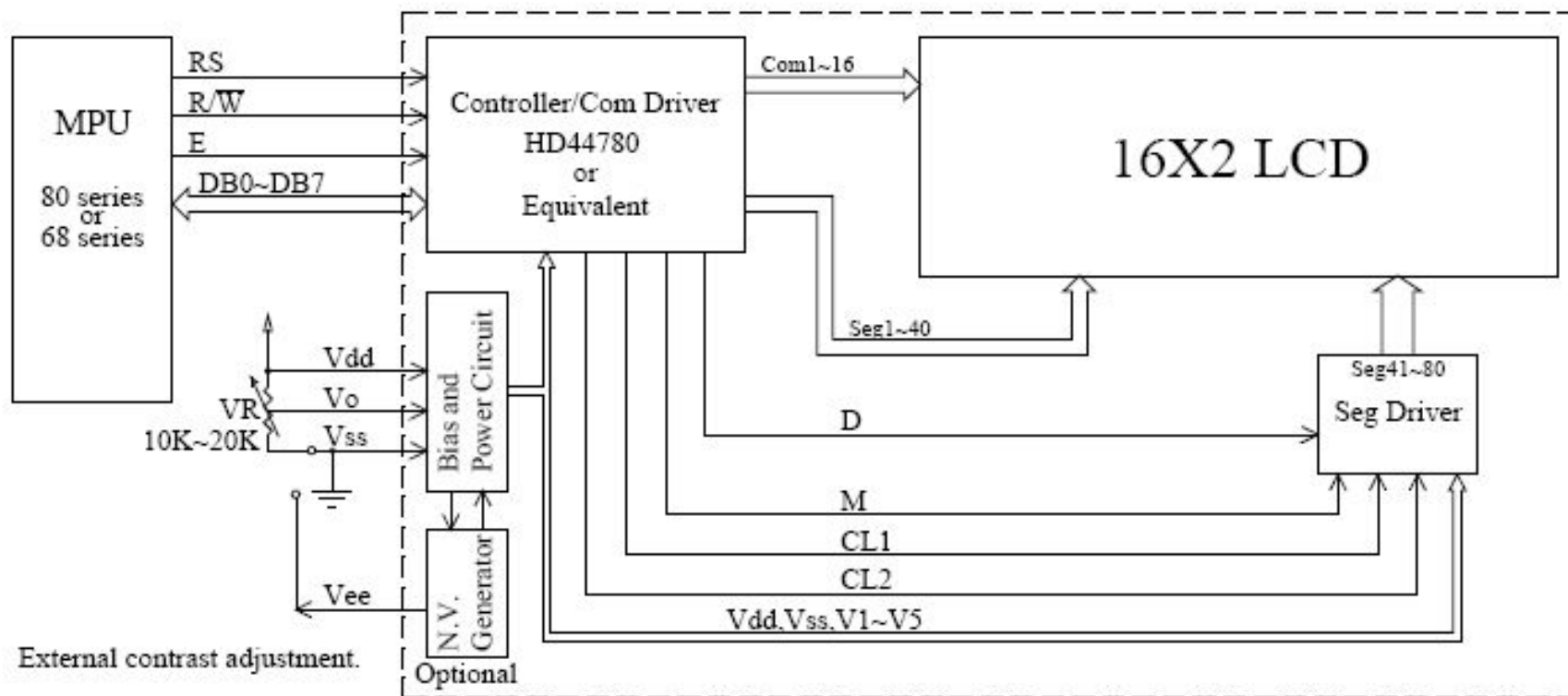
Вывод	Обозначение	Назначение вывода
1	GND	Общий вывод (0В)
2	UCC	Напряжение питания (5В/3В)
3	Uo	Управление контрастностью
4	A0	Адресный сигнал — выбор между передачей данных и команд управления
5	R/W	Выбор режима записи или чтения
6	E	Разрешение обращений к индикатору (а также строб данных)
7	DB0	Шина данных (8-ми битный режим)(младший бит в 8-ми битном режиме)
8	DB1	Шина данных (8-ми битный режим)
9	DB2	Шина данных (8-ми битный режим)
10	DB3	Шина данных (8-ми битный режим)
11	DB4	Шина данных (8-ми и 4-х битные режимы)(младший бит в 4-х битном режиме)
12	DB5	Шина данных (8-ми и 4-х битные режимы)
13	DB6	Шина данных (8-ми и 4-х битные режимы)
14	DB7	Шина данных (8-ми и 4-х битные режимы) (старший бит)

ПОДКЛЮЧЕНИЕ

RS	R/W	Operation
0	0	IR write as an internal operation (display clear, etc.)
0	1	Read busy flag (DB7) and address counter (DB0 to DB7)
1	0	Write data to DDRAM or CGRAM (DR to DDRAM or CGRAM)
1	1	Read data from DDRAM or CGRAM (DDRAM or CGRAM to DR)

Character located	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
DDRAM address	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
DDRAM address	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F

ПОДКЛЮЧЕНИЕ



Upper 4 bit Lower 4 bit	LLLL	LLLH	LLHL	LLHH	LHLL	LHLH	LHHL	LHHH	HLLL	HLLH	HLHL	HLHH	HHLL	HHLH	HHHL	HHHH
LLLL	CG RAM (1)			0	1	2	3	4			5	6	7	8	9	A
LLLH	CG RAM (2)		.	0	1	2	3	4			5	6	7	8	9	A
LLHL	CG RAM (3)		"	0	1	2	3	4			5	6	7	8	9	A
LLHH	CG RAM (4)		#	0	1	2	3	4			5	6	7	8	9	A
LHLL	CG RAM (5)		\$	0	1	2	3	4			5	6	7	8	9	A
LHLH	CG RAM (6)		%	0	1	2	3	4			5	6	7	8	9	A
LHHL	CG RAM (7)		&	0	1	2	3	4			5	6	7	8	9	A
LHHH	CG RAM (8)		'	0	1	2	3	4			5	6	7	8	9	A
HLLL	CG RAM (1)		(0	1	2	3	4			5	6	7	8	9	A
HLLH	CG RAM (2))	0	1	2	3	4			5	6	7	8	9	A
HLHL	CG RAM (3)		*	0	1	2	3	4			5	6	7	8	9	A
HLHH	CG RAM (4)		+	0	1	2	3	4			5	6	7	8	9	A
HHLL	CG RAM (5)		,	0	1	2	3	4			5	6	7	8	9	A
HHLH	CG RAM (6)		-	0	1	2	3	4			5	6	7	8	9	A
HHHL	CG RAM (7)		.	0	1	2	3	4			5	6	7	8	9	A
HHHH	CG RAM (8)		/	0	1	2	3	4			5	6	7	8	9	A

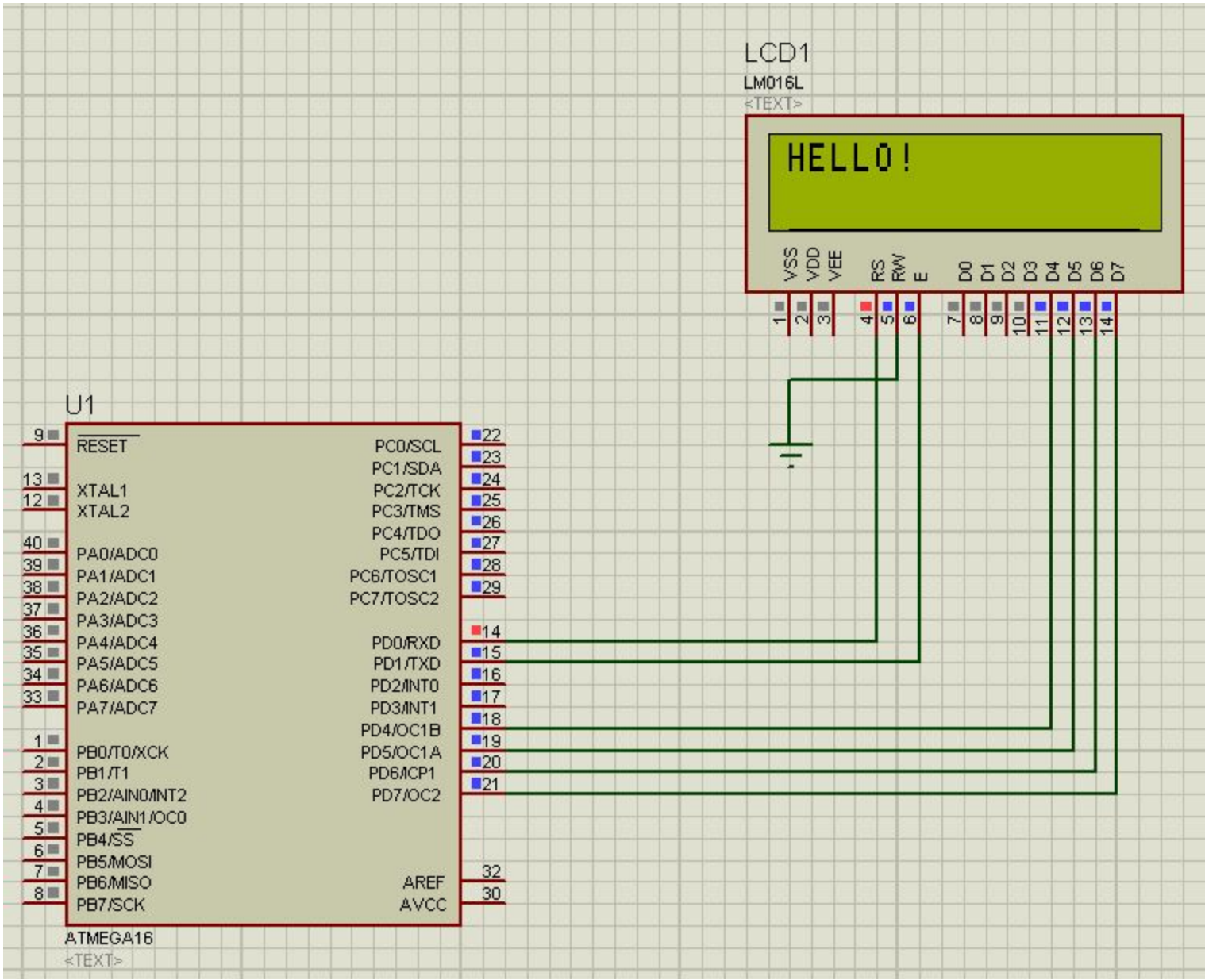


Диаграмма обмена по 4-х битному интерфейсу

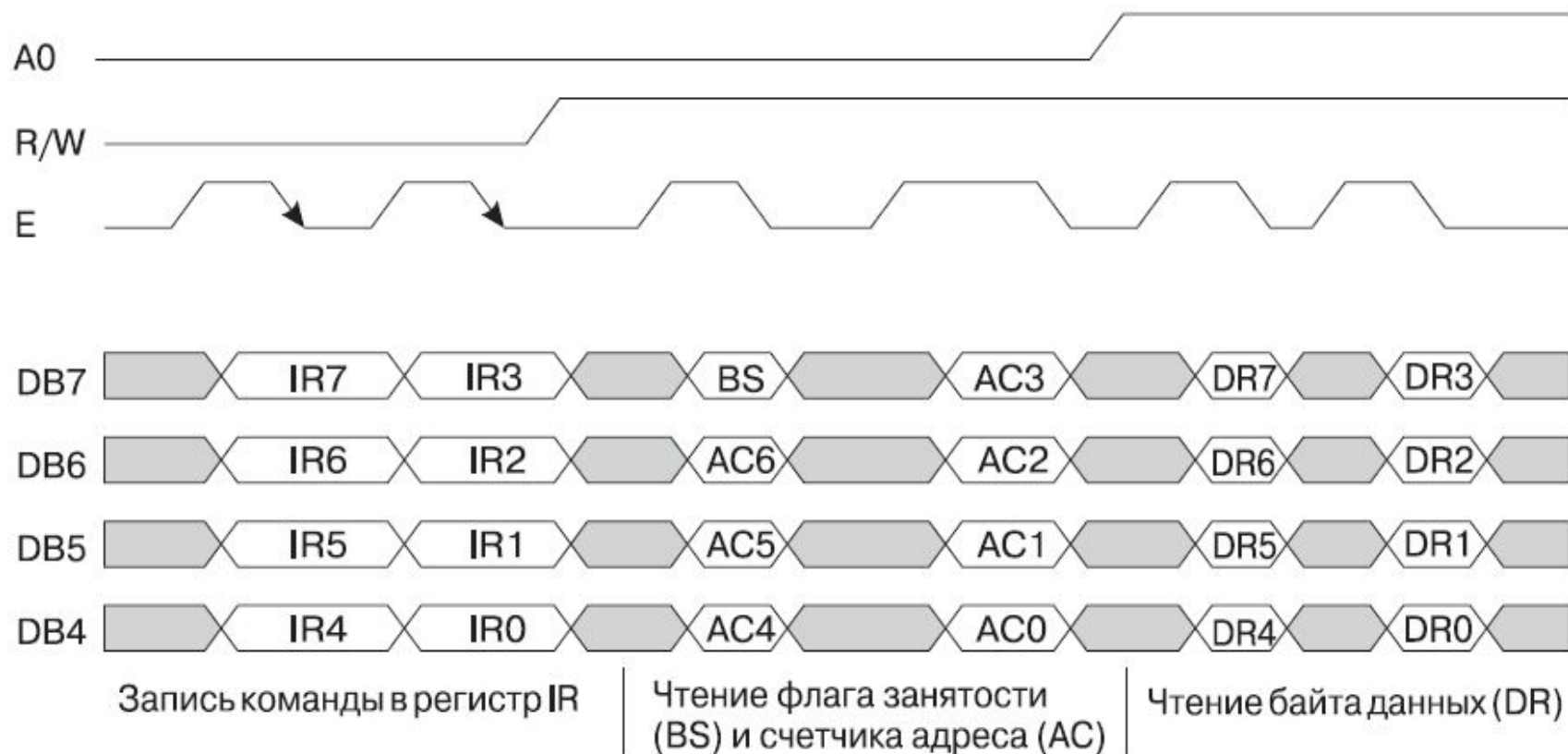
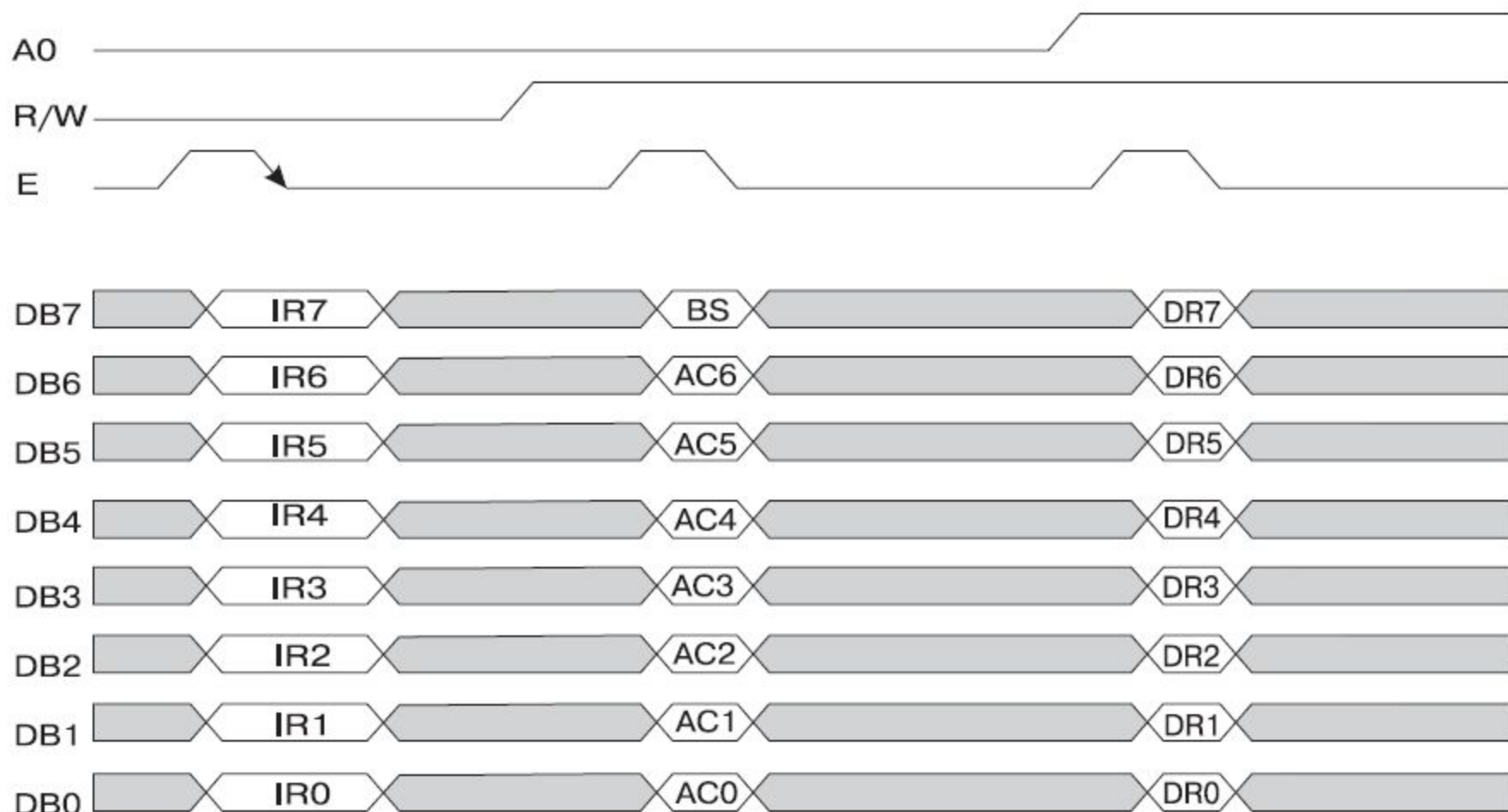


Рис. 5

Примечание. В каждом цикле обмена необходимо передавать (читать или писать) все 8 бит — два раза по 4 бита. Передача старших 4-х бит без последующей передачи младших 4-х бит **не допускается**.

Диаграмма обмена по 8-ми битному интерфейсу



Запись команды в регистр IR

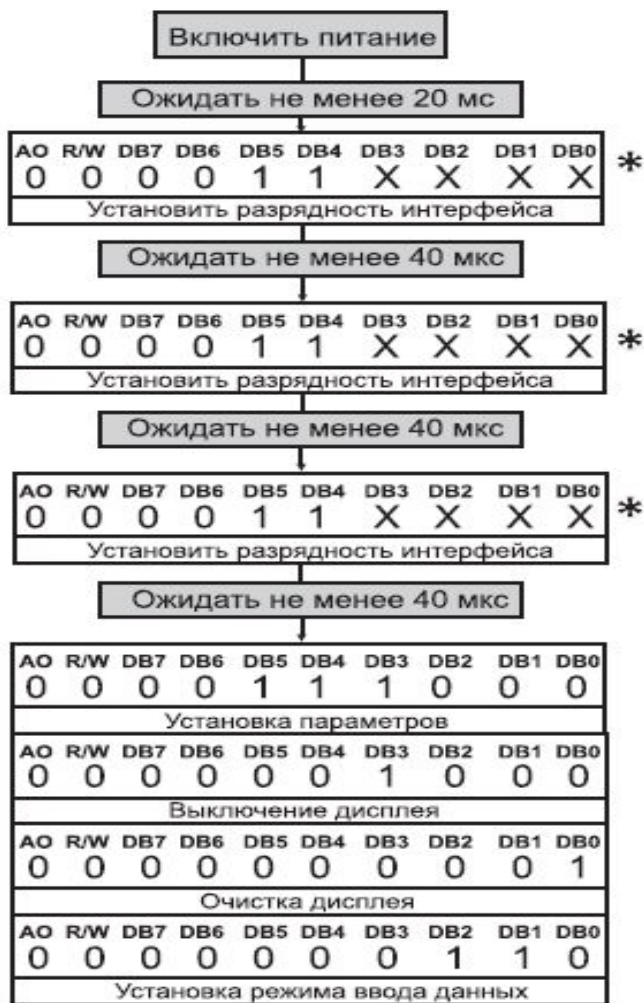
Чтение флага занятости (BS) и счетчика адреса (AC)

Чтение байта данных (DR)

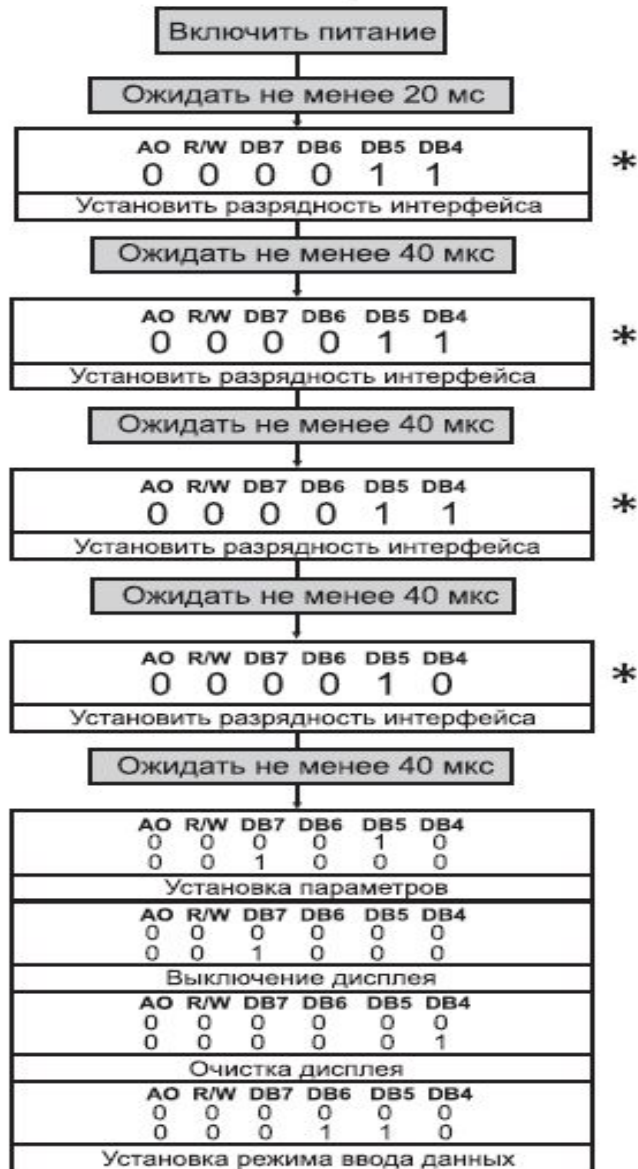
Начальная установка индикатора

Индикатор войдет в нормальный режим работы только после подачи на него следующих команд:

8-ми битный режим



4-х битный режим



* — после этих команд флаг занятости "BS" не проверять! Остальные команды выдавать в соответствии с таблицей 4 и примечаниями под ней.

Пример кода

```
void DisplayValue(unsigned long value)
{
    char i;
    unsigned long V;
    unsigned long digit;
    unsigned int code;
    unsigned char code_low, code_high;
    V = AcpValues[AcpChannel];//

    PORTB = 0;
    PORTB = 0b00010000;
    PORTB = 0;
```

Пример кода

```
for(i = 0; i < 5; i++)
{
    digit = V % 10;
    V = V / 10;
    code = mas8[digit];

    code_low = (code >> 0) & 0xF;
    code_high = (code >> 4) & 0xF;

    PORTB = code_low | 0b01010000;
    PORTB = code_low | 0b01000000;

    PORTB = code_high | 0b01010000;
    PORTB = code_high | 0b01000000;
}
```

Пример кода

```
PORTB = 0b01010000;//пропуск сегментаов  
PORTB = 0b01000000;
```

```
PORTB = 0b01010000;  
PORTB = 0b01000000;
```

```
    PORTB = 0b01010000;  
PORTB = 0b01000000;
```

```
PORTB = 0b01010000;  
PORTB = 0b01000000;//пропуск сегментаов
```

Пример кода

```
PORTB = 0b01011110;//запись буквы ... в дисплей
    PORTB = 0b01001110;
    PORTB = 0b01011000;
    PORTB = 0b01001000;
PORTB = 0b01010011;//запись буквы ... в дисплей
    PORTB = 0b01000011;
    PORTB = 0b01011110;
    PORTB = 0b01001110;
digit = (AscChannel + 1);//записываем номер канала в дисплей
code = mas8[digit];
code_low = (code >> 0) & 0xF;
code_high = (code >> 4) & 0xF;

PORTB = code_low | 0b01010000;
PORTB = code_low | 0b01000000;
PORTB = code_high | 0b01010000;
PORTB = code_high | 0b01000000;
}
```