

به نام خدا

## اتصال کیبرد کامپیوتر به AVR

نویسنده

یعقوب جان محمدی

کلمات کلیدی

کیبرد کامپیوتر، AVR

چکیده

در این مقاله نحوه اتصال کیبرد کامپیوتر به میکرو کنترلر های AVR بیان شده است.



## پرتکل کی برد - ارتباط کی برد با میکروکنترلر AVR:

در کی بردهای نوع قدیمی AT میکروپرسورهای 8042 و در نوع XT میکروپرسورهای 8044 کار اسکن صفحه کلید و ارتباط با PC را بر عهده دارند .



این ارتباط بر دو حالت است :

### ۱- دیتا

### ۲- دستور

دیتا وقتی ایجاد می شود که شما دکمه ای را فشار داده اید و پرسور کی برد یک کد (معمولا ) یک بایتی را که MAKE CODE نام دارد را به PC منتقل می کند و تا هنگامی که کلید فشرده شده است این کد به PC فرستاده می شود و وقتی که شما دکمه را رها کنید پرسور کی برد یک کد ( معمولا ) دو بایتی را به نام BREAK CODE به PC می فرستد به این معنی که دکمه مورد نظر رها شده است. دستور تنها از PC به کی برد انتقال پیدا می کند.

برای اطلاع از نحوه ی ارتباط کی برد و PC یعنی بررسی پرتکل کی برد ابتدا باید پورت کی برد را بشناسیم. باتوجه به شکل، پایه ی 1، DATA می باشد و پایه ی 3، GND، و پایه ی 4 VCC(4.5-5.5V) و پایه ی 5، CLOCK، می باشد. ماکسیمم جریانی که می توان از پورت PS2 کامپیوتر کشید 275mA می باشد، فرکانس کلاک از 10kHz تا 16.7kHz می باشد .

	
DIN41524, Female at Computer, 5-pin DIN 180°	6-pin Mini DIN PS2 Style Female at Computer

از وجود یک خط دیتا و یک خط کلاک نتیجه می شود که این ارتباط دو طرفه ، سریال و سنکرون ( یعنی پایه ی دیتا نسبت به پایه کلاک خوانده می شود ) می باشد و در این ارتباط PC مستر ( MASTER ) و کی برد

اسلیو (SLAVE) می باشد . پس کنترل نهایی بر روی این دو خط توسط PC انجام می شود . کلاک توسط پرسسور کی برد تولید می شود . حال حالت های چهار گانه ی CLOCK و DATA را بررسی می کنیم :

۱.  $CLOCK = HIGH$  و  $DATA = HIGH$  : حالت بیکار

۲.  $CLOCK = LOW$  و  $DATA = HIGH$  : ارتباط توسط PC ممنوع شده است

۳.  $CLOCK = HIGH$  و  $DATA = LOW$  : درخواست PC از کی برد برای فرستادن دیتا و تولید پالس

۴.  $CLOCK = LOW$  و  $DATA = LOW$  : بین کی برد و PC دیتا در حال رفت و آمد است

هر قاب یا فریمی که بین PC و کی برد تبادل می شود شامل 11 تا 12 بیت می باشد ، بصورت زیر :

۱. بیت شروع که صفر است

۲. هشت بیت دیتا که کم ارزش ترین بیت اول فرستاده می شود

۳. بیت پریتی ( پریتی فرد)

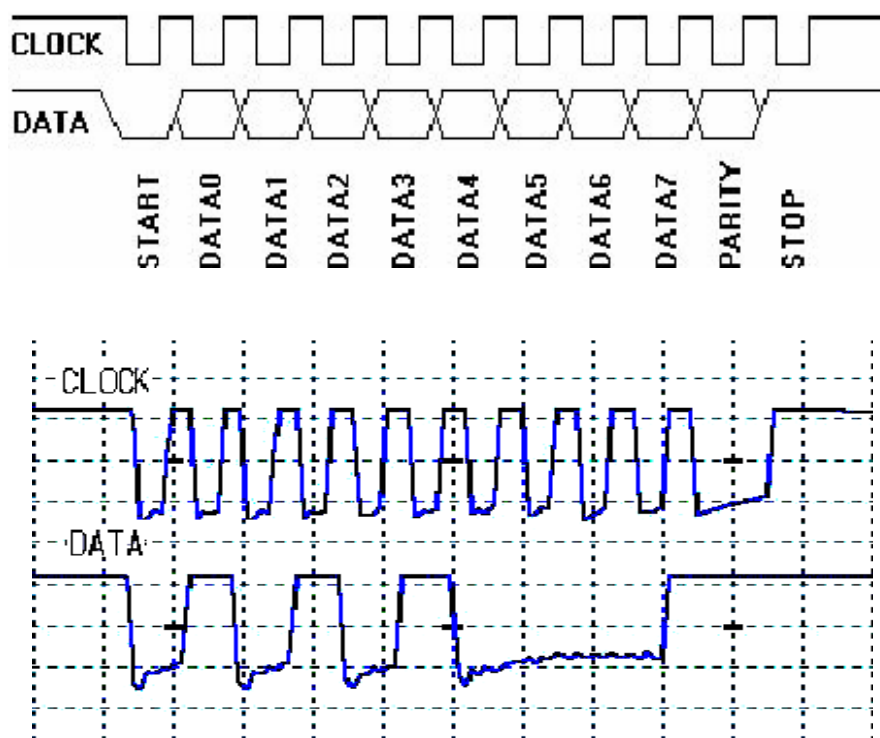
۴. بیت پایان که یک است

۵. بیت ACKNOWLEDGE ( این بیت فقط در ارتباط از PC به کی برد وجود دارد )

### نحوه ارسال دیتا از کی برد به PC :

وقتی که کی برد می خواهد داده ای به PC بفرستد ، ابتدا از HIGH بودن پایه ی CLOCK اطمینان پیدا می کند ، اگر پایه ی CLOCK ، HIGH نبود ، کی برد دیتا را در خود بافر می کند ( تا حداثر 16 کاراکتر ) تا بعد آنها را بفرستد . پایه ی CLOCK باید حداقل 50 میکرو ثانیه HIGH باشد تا کی برد دیتا را بفرستد . این ارتباط شامل 11 بیت بصورت زیر است :

کی برد وضعیت پایه ی DATA را هنگامی که CLOCK در سطح HIGH است تغییر می دهد و PC در لبه ی نزولی CLOCK پایه ی دیتا را می خواند. شکل روبرو نمونه ای از وضعیت پایه های CLOCK و DATA را هنگامی که کی برد در حال فرستادن کاراکتر "Q" ( کد 0X15 ) می باشد نشان می دهد :

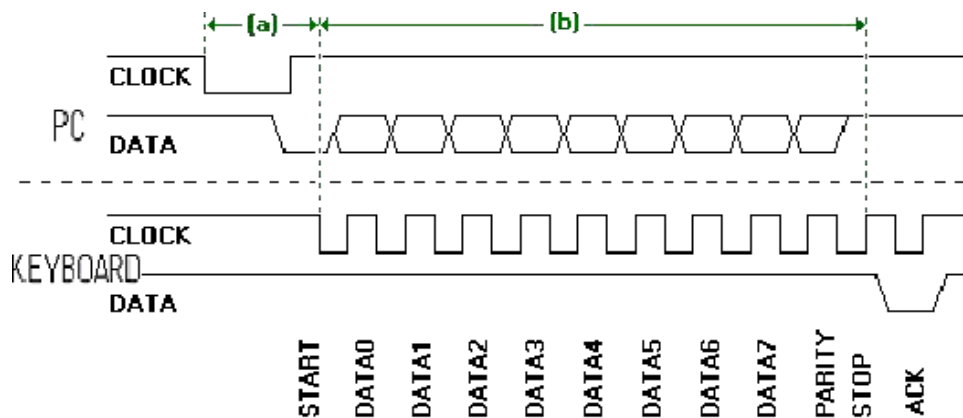


### نحوه ارسال دیتا از PC به کی برد :

هنگامی که PC می خواهد به کی برد دستور بدهد یا دیتا بفرستد یک حالت « در خواست برای فرستادن » را بصورت زیر بوجود می آورد :

۱. ارتباط را با LOW کردن پایه ی CLOCK برای مدت حداقل 100 میکرو ثانیه ممنوع می کند
۲. حالت « در خواست برای فرستادن » را با LOW کردن پایه ی DATA و بعد از آن HIGH کردن پایه ی CLOCK ایجاد می کند

بعد از اینکه این حالت توسط کی برد تشخیص داده شد ، کی برد شروع به تولید پالس در پایه ی CLOCK می کند و PC باید پایه ی دیتا را وقتی که CLOCK در سطح پایین است تغییر دهد ، پرسور کی برد نیز در لبه ی صعودی CLOCK ، پایه ی دیتا را می خواند .



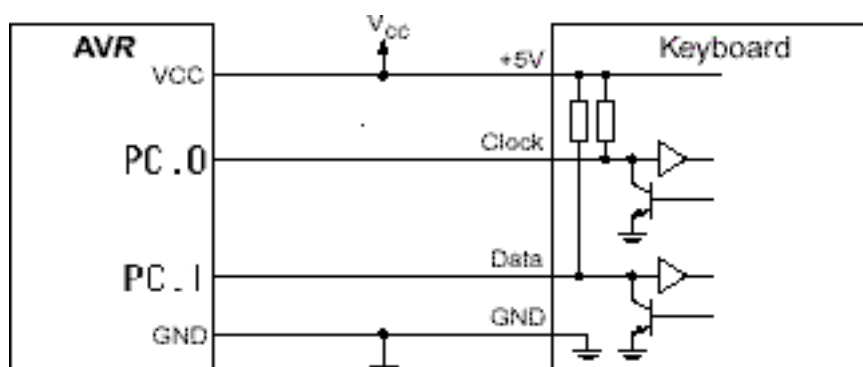
### ارتباط کی برد با میکروکنترلر AVR :

برنامه ی زیر بر نامه ای به زبان BASIC می باشد که توسط این برنامه می توان میکروکنترلر AVR (ATMEGA8) را با کی برد AT یا XT مرتبط کرد و دکمه ی فشار داده شده را بر روی LCD نمایش داد . مزیت اصلی این کار زمانی است که شما می خواهید کاراکترهای مختلفی را بر روی LCD یا تابلو روان نمایش دهید که این کار توسط کی برد های 4x4 بسیار مشکل می باشد.

```

$regfile="m8def.dat"
$crystal = 8000000
Config Lcdpin = Pin , Rs = Portd.0 , E = Portb.6 , D4 = Portd.1 , D5 = Portd.2 , D6
= Portd.3 , D7 = Portd.4
Config Lcd = 16 * 2
Cursor On
Cursor Blink
Home
Config Keyboard = Pinc.0 , Data = Pinc.1 , Keydata = Keydata
Dim A As Byte
Waitms 500
Cls
Main:
A = Getatkbd()
Select Case A:
Case 50
Lowerline
Case 56 :
Home
Case 44
Cls
Home
Case 52 :
Shiftcursor Left
Case 54 :
Shiftcursor Right
Case 13 :
Lowerline
Case 9 :
Shiftcursor Right , 4
Case 8 :
Shiftcursor Left
Lcd " " ;
Shiftcursor Left
Case Else Lcd String(1 , A)
End Select
jmp main
End
Keydata:
Data 0 , 0 , 0 , 0 , 0 , 200 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , &H5E , 0
Data 0 , 0 , 0 , 0 , 0 , 113 , 49 , 0 , 0 , 0 , 122 , 115 , 97 , 119 , 50 , 0

```



شماتیک برنامه روبرو

Data 0 , 99 120 , 100 , 101 , 52 51 , 0 , 0 , 32 , 118 , 102 , 116 , 114 , 53 , 0  
Data 0 , 110 , 98 104 , 103 , 121 , 54 , 7 , 8 , 44 , 109 , 106 , 117 , 55 , 56 , 0  
Data 0 , 44 107 , 105 , 111 , 48 57 , 0 , 0 , 46 , 45 108 , 48 112 , 43 0  
Data 0 , 0 , 0 , 0 , 0 , 92 , 0 , 0 , 0 , 0 , 13 0 , 0 , 92 0 , 0  
Data 0 , 60 0 , 0 , 0 , 0 , 8 , 0 , 0 , 49 , 0 , 52 55 , 0 , 0 , 0  
Data 48 44 , 50 53 , 54 , 56 0 , 0 , 0 , 43 , 51 , 45 42 , 57 , 0 , 0  
Data 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0  
Data 0 , 0 , 0 , 0 , 0 , 81 , 33 0 , 0 , 0 , 90 , 83 , 65 87 34 0  
Data 0 , 67 88 , 68 69 , 0 , 35 , 0 , 0 , 32 , 86 , 70 84 82 37 , 0  
Data 0 , 78 66 , 72 71 , 89 38 0 , 0 , 76 , 77 , 74 85 , 47 , 40 , 0  
Data 0 , 59 75 , 73 79 , 61 41 0 , 0 , 58 , 95 , 76 48 , 80 , 63 , 0  
Data 0 , 0 , 0 , 0 , 0 , 96 , 0 , 0 , 0 , 0 , 13 94 , 0 , 42 , 0 , 0  
Data 0 , 62 0 , 0 , 0 , 8 , 0 , 0 , 49 0 , 52 55 , 0 , 0 , 0 , 0  
Data 48 44 , 50 53 , 54 , 56 0 , 0 , 0 , 43 , 51 , 45 42 , 57 , 0 , 0