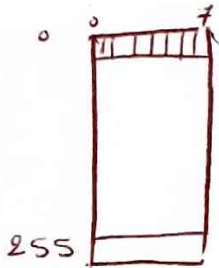


میکروکنترلرهای AVR :

میکروکنترلر keyboard - 884 (صفحه ۱۱۹-۱۲۰) بیان صفحات هر یک از این (صفحه ۱۱۹-۱۲۰) بیان صفحات هر یک از این

یادداشت : حافظه وقت
یادداشت : حافظه وقت
یادداشت : حافظه وقت



حافظه 256 بیت یعنی : ما از 0 تا 255 خانه داریم که هر خانه 8 بیت است
برای AVR دو تا بیان داریم
سیک (Bascom)

AVR میکروکنترلرهای 8 بیت ، ساخت شرکت Atmel ، می باشند . این میکروکنترلرها ، بهترین حجم بازار ایران را به خود اختصاص داده اند . از جمله مزایای این میکروکنترلرهای می توان به موارد زیر اشاره کرد :

۱. کارایی بالا در توان مصرفی کم . (توان مصرفی بسیار کم تا ۱ میلی وات)

۲. دایره 100 تا 140 کیلوهرتز ، که البته آن ها ، در یک سیکل اجرا می شوند .
سرعت بیشتر → ۲۱

۳. قابلیت کار در فرکانس 1 تا 10 مگاهرتز . (به جز مدل های ساخت)

۴. حافظه E2PROM ، فلش ، SRAM با دقت فرکانس بالا
سخت → SRAM ← DRAM

۵. تایمر / شمارنده (ATM8)

۶. تایمر / شمارنده PWM

۷. تایمر / شمارنده ADC ، ۱۰ بیت ، (3 تا 12 بیت) ، ما خواهیم دید که از این میکروکنترلرها ، می توانیم

۸. دایره RTC (تایمر ساعت است که زمان را تا بی نهایت می شمارد و می تواند با سیلانور مجزا ، خاموش می کنند .)

۹. خاموش کننده کانال های داخل

۱۰. usart قابل برنامه ریزی

۱۰ wachdog تابع برنامه‌ریزی است بلا تور داخلی
۱۱ ارتباط سریال ISP برای برنامه‌ریزی (Program کردن داخلی مدار)

۱۲ هنگامی که می‌خواهیم داخل مدار است با Program کردن ISP و توان می‌خواهیم برنامه‌ریزی کنیم.
برای برنامه‌ریزی از چهار خط Miso و Mdi و Sck و Reset استفاده می‌شود.

۱۳ قابلیت ارتباط سریال ISP به صورت Master یا Slave (راستی یا چپ)

۱۴ قابلیت ارتباط Jtag (یک نوع ارتباط است، که از طریق آن می‌توانیم قطعه‌ها را تست کنیم)
برنامه‌ریزی می‌کند و راه‌اندازی می‌کند.

۱۵ Reset کردن می‌کند، بعد از روشن شدن.

۱۶ دارای چندین مد بیکاری.

۱۷ چندین منبع تولید ولتس.

۱۸ توان ساز داخلی.

۱۹ تنوع در امکانات به بندسی و قیمت و غیره.

۲۰ دو گلد سوار می‌کند، این می‌کند و راه‌اندازی می‌کند.

شما می‌توانید دیگر اطلاعات را از دست‌نویس استخراج کنید.

- برخی از عبارات که در می‌کند و کار می‌رود:

۱۱ V_{cc} و Ground . ستفوا از این دو قطعه پایه‌های تغذیه می‌باشند . که عموماً $V_{cc} = 5V$ و $GND = 0V$
شما می‌توانید این تغذیه را از پورت USB یا سیروی جزو کنید . (USB یک پایه 5V و یک پایه GND دارد)
۱۲ پورت 8 که می‌کند در ای چندین پورت می‌باشند و هر پورت دارای چندین پایه است . (عموماً 8 پایه)
برای مثال می‌کند ATM 32 ، دارای 4 پورت ، A, B, C, D می‌باشند . که هر پورت

8 پایه دارد. پورت A، از پایه 33 تا 40 میسر می باشد. در شماره 40، سن A.0

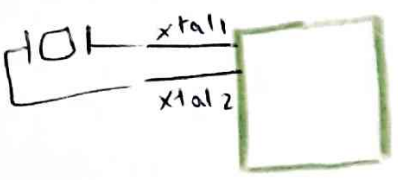
(Pin a.0) میگویند. پایه 39 (Pin a.1) و پایه 33 (Pin a.7) می باشد

برای دیدن سایر اطلاعات میگردید در سایت آن ها، مراجعه کنید.

3 Pin به هم پایه میگرد، Pin نیز گفته می شود مثلاً Pin a.1، یعنی پایه شماره پورت A

مدرک میگرد ATM16، پایه شماره 39 می باشد.

4 میکروکنترلر، میگرد برای تنظیم زمان برای انجام کارها، یک زمان ساز، نیاز دارد. که به این زمان ساز میکروکنترلر گفته می شود. حالته میکروکنترلر مورد استفاده برای AVR 16MHz می باشد.



همچنین میگرد برای AVR - دارای زمان ساز داخلی می باشد که در پورت

سازیم تنظیم از این استفاده کنیم. میکروکنترلر به 2 پایه xtal1 و xtal2 متصل می شوند. این پایه ها برای میگرد ATM16، پایه های 12 و 13 می باشد.

5، 6 پایه های MR دارای چندین نقش می باشند.

Pin a.0 (پایه شماره 140 در ATM16، علاوه بر این که می تواند به عنوان ورودی یا خروجی محسوب می شود

می تواند به عنوان مبدل، ADC استفاده گردد. شما می توانید در این از نقش یک Pin یا Port استفاده کنید. برای همین نقش دیگر این ها به در سایت آن ها مراجعه نمایید.

6، حسگر حافظه 8 مکان های از حافظه می باشد. که می تواند در سنسور ها در آن جا ذخیره می شود.

به تابه میگرد که می تواند استفاده است. اما در بیشتر میگرد میگرد، یک مقفول است و مقفول AC، چه میگرد میگرد.

7، آدرس حافظه

برای دیگر کارها فقط، آن کارها به بخش های کوچکتر تقسیم می کنند. شما می توانید این کارها را آدرس دهی و سنسور مناسب های مورد نظر را ذخیره کنید.

آشنایی مختصه با محیط Bascom (زمان بسیار)

نقشه: گذرین زمان ۲۰

۱- منوی File

۱-۱ new: باز کردن این گزینه یک صفحه جدید برای نوشتن برنامه جدید باز می شود. این صفحه مجزبه در براس می باشد، دستورات می باشد. پیش از هر کاری که دستور در دست وارد شود به رنگ آبی در می آید ولی اگر دستور اشتباه وارد شود به رنگ سفید (مسلک) است. این مورد برای تعداد کمی از دستورات اجرا نمی شود.

۱-۲ open: باز کردن این گزینه می تواند برنامه ای که از قبل ذخیره کرده اید. باز می شود. برنامه ها با پسوند .bas ذخیره می شود. با هم می توانید برنامه خود را در داخل برنامه های دیگر مثل کامپایلر مانند `notepad` بزنید. در این پنجره باز کنید.

۱-۳ close: با انتخاب این گزینه، پنجره ای که برای نوشتن برنامه باز شده، بسته می شود. در هر پنجره که یک صفحه برنامه ای را که در پنجره جدید نوشته اید ذخیره نکرده باشید. از لحظه در مورد ذخیره کردن برنامه سوال می شود.

۱-۴ save, save as: این دوگزینه ای ذخیره کردن پرونده به کار می روند.

۱-۵ print و print preview: این دوگزینه برای چاپ کردن برنامه، استفاده می شوند. باز کردن این گزینه می تواند نسخه کامل چاپ را قبل از چاپ مشاهده نماید.

۱-۶ Exit: باز کردن این گزینه، برنامه Bascom به طور کامل بسته می شود. اما اگر برنامه ای، ذخیره نشده باشد در مورد ذخیره برنامه از شما پرسیده می شود.

۱-۷ زیرگزینه Exit، چندگزینه دیگر وجود دارد. که برای دستورات سریع، اغلب قابل چاپ می باشد که باز کردن می باشد.

۲- منوی Edit

۲-۱ undo, redo: این گزینه برای دستوراتی که آخرین گزیننده تغییرات اجرا کرده می باشد.

۲-۲ copy, cut, past: این گزینه برای برداشتن، یا کپی کردن قسمتی از متن به جای دیگری می باشد.

۲-۳ Find, Find next : برای پیدا کردن تسمی از متن در برنامه‌هاست. چگونه : بعد از انتخاب گزینه Find پیغام عددی باز شده به هر text to find ، متن مدنظر را تایپ کرده بعد از کلیک ok ، متن مورد نظر در برنامه انتخاب می‌شود.

Find next : متن‌هایی که در برنامه پیدا شده خط‌های بعدی برنامه وجود دارد انتخاب می‌کند.

۲-۴ replace : برای تغییر تسمی‌ها می‌توانید متن را جایگزین متن جدید در برنامه کنید. یعنی در قسمت text to find متن یا کلمه‌ای که باید توسط متن یا کلمه دیگری جایگزین شود را تایپ کنید. در قسمت ^{with} replace ~~text~~ متن را که باید جایگزین شود را تایپ می‌کنیم.

۲-۵ در گزینه‌های برای تراشیدن علامت در خطوط مختلف در پرانتزها می‌توانید.

۲-۶ indent block , out indent block : متن انتخاب شده را به اندازه یک tab به چپ یا راست منتقل می‌کند.

۴- منوی View :

۱۱ Pin Layout : با انتخاب این گزینه، پیغامی برای باز کردن پنل‌ها می‌آید که می‌توانید هر کدام را ببندید. آن‌ها بسته می‌شوند. با بردن موس روی هر یک از آنها، نقش آن در زیر تصویر، نوشته می‌شود.

۱۲ pdf viewer : با انتخاب این گزینه، می‌توانید صفحات می‌تواند هر کدام را ببینید. به غیر از آن‌ها download کنید. با کلیک روی گزینه، خود download به شما گفته می‌شود. در همین برنامه Adobe Acrobat برای باز کردن pdf ها نیاز دارید.

۱۳ error panel : با انتخاب این گزینه پیغام‌ها باز می‌شود. این پیغام بعد از compile کردن برنامه، در صورت وجود خطا، به شما خواهد گفت باز می‌شود.

۲ tib : باز کردن این گزینه پنجره ای باز می شود که حاوی نکات برای بهره گیری از باسcom است

۳-۱ : Program

۳-۲ : compile : انتخاب این گزینه برنامه در سطح سازه زبان ماشین ترجمه می شود و فایل های از تبدیل می شود

و تراسرین در حین ساخته می شوند. اگر در این مرحله نامی خطا باشد پنجره ای باز می شود که در آن خطاها نمایش داده می شود. با کلیک کردن روی هر خطا می توانیم در آن خطا کلیک کنیم تا آن خطا را توضیح دهد.

۳-۳ : syntax check : انتخاب این گزینه برنامه از نظر خلط املایی چک می شود. (باز کردن compile در این مرحله)

۳-۴ : show result : انتخاب این گزینه پنجره ای باز می شود که در آن نتایج اجرای برنامه وجود دارد.

۳-۵ : simulate : انتخاب این گزینه پنجره شبیه سازی باز می شود. در این پنجره می توانیم LCD

keyboard مدل آنالوگ به دیجیتال درگیر می باشد. می توانیم برنامه خود را شبیه سازی کنیم

۳-۶ : send to chip : دارد محیط برنامه کردن می کند و می توانیم

۵-۱ : tools : ابزار کار با میکروکنترلر
۵-۲ : کیبورد
۵-۳ : پنجره نمایش LCD
۵-۴ : پنجره شبیه سازی
۵-۵ : پنجره نتایج اجرای برنامه

مراحل نوشتن یک برنامه جدید (بند تک برنامه)

۱- برای نوشتن برنامه در باسcom باید new را در منوی File انتخاب می کنیم. پنجره جدیدی باز می شود که می توانیم نام برنامه را در آنجا وارد کنیم. پس از آن می توانیم بر روی فایل کلیک کنیم و آن را در مکان مورد نیاز خود ذخیره کنیم.

۱ & regfile = نام فایل

۲- در مرحله بعد می توانیم نام پروژه را در قسمت پروژه نام وارد کنیم. در این مرحله می توانیم نام پروژه را در قسمت پروژه نام وارد کنیم. در این مرحله می توانیم نام پروژه را در قسمت پروژه نام وارد کنیم.

۳- در این مرحله می توانیم نام میکروکنترلر را در قسمت میکروکنترلر نام وارد کنیم. در این مرحله می توانیم نام میکروکنترلر را در قسمت میکروکنترلر نام وارد کنیم.

۴- در این مرحله می توانیم نام میکروکنترلر را در قسمت میکروکنترلر نام وارد کنیم. در این مرحله می توانیم نام میکروکنترلر را در قسمت میکروکنترلر نام وارد کنیم.

"Attiny 12.dat"
 15.dat
 ... 22.dat
 ~~~~~  
 Attiny میکروکنترلر

"1200.def.dat"  
 ~~~~~  
 AT90S میکروکنترلر

"M16def.dat"
 ~~~~~  
 Mega میکروکنترلر

سؤال

2)  $\phi \text{ crystal} = \frac{8000000 \text{ Hz}}{8 \text{ MHz}} = X \text{ Hz}$  خط بعدی معرف کریستال می باشد.

که  $X$  کریستال مورد استفاده بر حسب هدف است. شما همچنین می توانید مقدار کریستال را بزرگ کرد و تقسیم کنید. (این کار را می توانید کریستال داشته باشید و تقسیم کنید)

config clock Div = X (constant)

if  $X=8 \Rightarrow \frac{8000000}{8} = 1000000 = 1 \text{ MHz}$

که  $X$  constant، برای این مقدار است: 128, 256, 1, 2, 4, 8, 16, 32, 64

$\phi \text{ crystal} = 8000000$

config clock Div = 8

این به بعد میگردیم با فرکانس 1MHz کار می کند.

3) بجای معرف کریستال، ترتیب به معرف امکانات می باشد: Keypad، LED، ...

(رشته آخرین دستور End است. همچنین از شماره ها به ابتدا end نوشته می شود تا به شماره های خردمان، بقیه باید داخل loop قرار بگیرد و تا حالات را مرتباً چک کند.)

امکانات شامل: تایمرها، ADC، و ... با فرکانس کار کردن می باشد.

معرف امکانات با دستور شروع می شود.

### 3) Config

config LED = 16 \* 2

config portb = input

عدد عددی دیگر در امکانات جانبی ترتیب به استفاده از آن ها می باشد. مثلاً برای استفاده از امکانات باید آن ها را در حلقه قرار دهیم. در نهایت شماره با 5 به بیان می آید. همیشه چاقوی بر نامه به زبان basic برای AVR، استفاده می باشد.

# - دستورات مربوط به پین ها (بخش عملی و پایه ها)

در Bascom ، برای استناد به یک پورت باید آن را به صورت یک ورودی یا خروجی قرار داد. یک پورت در هنگام خواندن خروجی یک ورودی را میخواند که خروجی آن در یک پورت هنگام به خواندن ورودی نیز میخواند که خروجی به آن در کتابخانه هم دادن در کتابخانه مندرک لیدری که برآورد آن به S دات است. در گرفتن دیتا به مندرک LED ، که برآورد آن به GND متصل است ، در نقطه تکسیم

config port x = (input / output)

config port b.0 = input

config port b.1 = output

که پورت X ، یک از پایه های میکروکنترلر باشد ، مانند config port A = output ، پورت A به عنوان خروجی تکوین شده است / پورت B به عنوان ورودی تکوین شده است. در همین ترتیب می توانیم از پایه های پورت A به عنوان ورودی یا خروجی تکوین کنیم. مانند config pin a.0 = input ، پایه 0 یک از پورت A به عنوان ورودی تکوین شده است. config pin c.5 = output ، پایه 5 از پورت C به عنوان خروجی تکوین شده است.

## - دستورات مربوط به پورت ها

1 دستور toggle : این دستور ، یک bit را برعکس میکند.  $\text{toggle } \text{bit}$  هر کد یک پورت یا پین 8 bit دیگر باشد.

2 دستور set : این دستور ، یک bit را یک میکند.  $\text{set } \text{bit}$  بیت هر کد یک Pin از پورت یا پین دیگر باشد.

3 دستور Reset : این دستور ، یک bit را صفر میکند.  $\text{Reset } \text{bit}$  بیت هر کد یک Pin از پورت یا پین دیگر باشد.

4 دستور Alias : این دستور برای تغییر نام استناد در کد است.

DIM  $\varphi$  as bit

$\varphi$  Alias pin B.0



این دستور cpu میخواند، برای یک بار پس از آن میخواند. هر بار که پایه یک یا صفر میخواند (صفر یا یک) (تغییر میخواند)  
 (تغییر میخواند) دستورات زیر خطی را به همراه میخواند. این دستورات همگی بر اساس

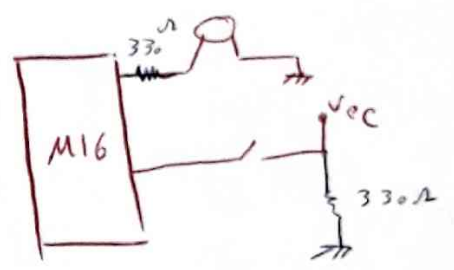
B.t wait X, set / Reset

X یا پایه است که قرار است، یک شود. مثل Port A.0 یا Port B.7 یا فید. set / Reset. در صورت  
 قرار می set. هنگامی که پایه یک میخواند، زمانهای زیر دستورات اجرا می شود. در صورت قرار می Reset.  
 هنگامی که پایه صفر میخواند، دستورات زیر خطی، اجرا می شود. مانند

مثال: زمانهای تنظیمی که از یک مایکرو ۸۰۸۵ (۸۰۸۵) استفاده می شود. پورت B.6 و خروجی B.7  
 به عنوان خروجی استفاده می شود. در صورتی که پایه B.7 صفر می شود به LED متصل می شود. پایه B.6 یک مایکرو می خواند

```

$ regfile = 'M16 def.dat'
$ crystal = '8000000'
config port b.7 = input
config port b.6 = output
bit wait port b.7, Reset
set port b.6
B.t wait port b.7, set
Reset port b.6
end
    
```



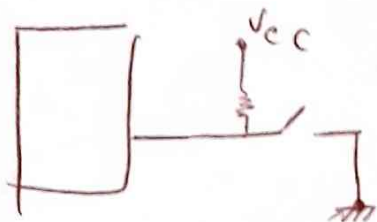
این مثال: همان دیده شده مشاهده می شود از یک مایکرو ۸۰۸۵  
 و قطعه LED استفاده شده است. در خط اول

زمانی که مایکرو در حالت میخواند (مایکرو = M16) در دستورات (8 MHz)

در خط دوم Port B.7، به عنوان خروجی تنظیم شده است و قطعه همان  
 متصل است. در خط چهارم Port B.6، به عنوان خروجی تنظیم شده است.

LED و آن متصل می شود. در خط پنجم، دستورات B.t wait و Port B.7 که میخواند  
 اگر B.7 = 0، cpu میخواند خط بیسی میخواند (تغییر میخواند) بار میخواند cpu میخواند  
 B.t wait، متصل می شود. زمانی که شرط صفر یا یک میخواند، این متصل می شود (این خط میخواند)  
 پایه B.6 = 1 میخواند و LED روشن می شود. در خط بیسی، دوباره پایه B.7 میخواند (تغییر میخواند) بار میخواند،  
 میخواند حالت متصل است. یعنی اگر B.7 = 1 میخواند، cpu میخواند خط بیسی میخواند و LED روشن می شود.

دو تا به ما درس اول encl بین می‌نویسد



مهندسی برای دست مهندسی است

کلید باز ← منطق یک می‌باشد

کلید بسته ← منطق صفر می‌باشد

در شکل می‌توان دانست وقتی کلید باز است، نوندرقه و منطق یک می‌باشد.

دو تا به ما درس اول

Port c ، Port b ، m16

\$ reg h:1 = "16 def.dat"

\$ crystal = 8000000

config portc = output

set pin c.0

Reset pin c.1

set pin c.2

Reset pin c.3

set pin c.4

Reset pin c.5

set pin c.6

8 پورت  
C کن 7 تا به  
گرد  
Reset pin c.7

wait 1

toggle portc

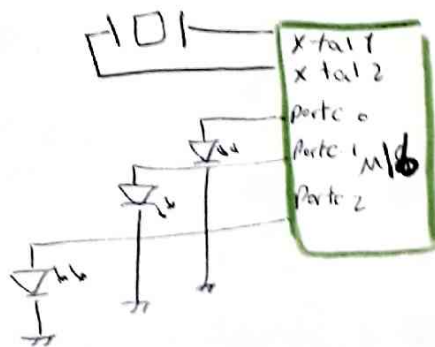
wait 1

loop Loop

end

(به جای toggle می‌توان دوباره به دست)

(برگشتن دهنده‌ها بالا را می‌توانیم)



دستورات تأخیر

برای ایجاد تأخیر در برنامه، دستور wait استفاده می‌شود. دستور wait به شکل زیر است.

wait x  
این دستور برای ایجاد تأخیر مشخصه‌ای می‌باشد. مقدار تأخیر به ثانیه است که بین ۱ تا ۹۹۹۹۹۹۹۹ باشد.

تأخیر به مدت 500ms  
\* wait 500

wait ms x  
تأخیر به مدت 720ms  
مقدار تأخیر می‌باشد. مقدار تأخیر به ثانیه است که بین ۱ تا ۹۹۹۹۹۹۹۹ باشد.

تأخیر به مدت 720ms  
wait ms 720

wait x  
این دستور برای ایجاد تأخیر نامتناهی می‌باشد. مقدار تأخیر است که عددی بی‌نهایت باشد.

تأخیر به مدت 1000s  
wait 1000

- delay

این دستور همواره استفاده می‌شود. یک تأخیر یک ثانیه می‌باشد. ایجاد می‌شود.

نوع : هرگاه دستور wait، کار بود، برنامه در آن جا به اندازه زمان مورد نظر متوقف می‌شود.

دستورات حلقه در پس :

گام‌ها و مات نیز است که برنامه مدام اجرا می‌شود یا در زمان اجرا، از یک خط به خط می‌پرسد. برای این کار حلقه‌ها در دستورات پس استفاده می‌شود.

دستور do-loop : شروع این حلقه با Do در پایان آن با Loop است.

در پس از یک قسمت برنامه به قسمت دیگر، هر گاه از دستور jump یا goto استفاده کرد. از دستورات نوع به عنوان حلقه نیز می‌توان استفاده کرد.

(Label) φ :

D0

برنامه نوشته شده

برنامه نوشته شده

Loop

jmp φ or goto φ :

شکل : نشان برنامه نویسی

حلیه کا چکارم

- LCD کارکنده :

LCD سے کارکنده سے درجہ اولیہ  
LCD سے کارکنده سے درجہ اولیہ

این نوع LCD در حقیقت نوع سلفی است و دارای چند سطر سونجی است که با سونجی که در آنجا قرار دارد  
سبب سطر سونجی است که با سونجی در سطر نام این LCD اگر در آن سطر است

LCD 16x1 : دارای یک سطر 16 سونجی است

LCD 16x2 : که دارای یک سطر 16 سونجی است که با سونجی در آن سونجی سونجی به بعد را اگر در آن سونجی

LCD 16x2

LCD 16x4

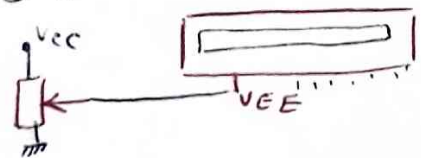
LCD 20x2 : LCD 20x4 : LCD 40x2 : LCD 40x4

LCD های کارکنده دارای 16 پایه هستند که در زیر آورده شده است

پایه 1 :  $V_{SS}$  : پایه GND ، LCD است . پایه بیستم مدار وصل شود . این سونجی دارد به GND

2 :  $V_{DD}$  : این پایه ، پایه  $V_{CC}$  ، LCD است . که باید به 5 وصل شود .  $V_{SS}$  دارد

3 :  $V_{EE}$  : این پایه ، پایه  $V_{EE}$  ، بیکیس های LCD را تعیین میکند . اتصال آن طبق مدار



پایه شماره 4:  $R_5$  ، در LCD در حقیقت برنام رستورالعمل و داده وجود دارد. اگر  $R_5 = 0$  LCD برای گرفتن دستورالعمل آنگاه هر شود. در غیر این صورت برای داده ، آنگاه یک نکل در لغورد  $CLS$  یک دستورالعمل است  $RWA$  یک داده است که باید برای LCD نوشته شود

پایه شماره 5:  $Rw$  : این پایه دو وضعیت دارد.  $Rw = 1$  ← برای خواندن از LCD

$Rw = 0$  ← برای نوشتن در LCD ، آنگاه هر شود. پایه

پایه شماره 6:  $EN$  : این پایه هر زمان ، LCD را انتخاب کرد تا کسی ارسال کنیم. پایه

پایه شماره 7:  $DB_0$  : این پایه برای آدرس  $Data$  (اطلاعات) از LCD می باشد. (پایه ی انتقال)  $DB_1$  : این پایه برای  $Data$  (اطلاعات) از LCD می باشد. (پایه ی انتقال)

پایه شماره 8:  $DB_1$  : این پایه برای  $Data$  (اطلاعات) از LCD می باشد. (پایه ی انتقال)

$DB_2$  : این پایه برای  $Data$  (اطلاعات) از LCD می باشد. (پایه ی انتقال)

پایه شماره 11:  $DB_4$  : این پایه برای  $Data$  (اطلاعات) از LCD می باشد. (پایه ی انتقال)

$DB_5$  : این پایه برای  $Data$  (اطلاعات) از LCD می باشد. (پایه ی انتقال)

پایه شماره 15: این پایه و پایه شماره 16 تغذیه  $LED$  ، در پشت LCD می باشد. که به آن تغذیه می شود.

پایه شماره 16: این پایه و پایه شماره 15 تغذیه  $LED$  ، در پشت LCD می باشد. که به آن تغذیه می شود.

— راه اندازی LCD توسط Bascom

(1) LCD می تواند به طریق 4 بیده ، 4 بیده ، به سه روش تعریف شود.

config LCD Bus = constant

که  $4 = constant$  به معنی 4 بیده می باشد.  $8 = constant$  به معنی 8 بیده می باشد.  $16 = constant$  به معنی 16 بیده می باشد.

در هر 4 بیده ، قطعه ترانس LCD توسط رده 8 بیده هر زمان اطلاعات را که به LCD نوشته شده است را خواند و به بیرون می ارسل کرد. هر کجا که در محیط  $Data_1$  تا  $Data_3$  برای خواندن نوشته شده است را خواند و به بیرون می ارسل کرد.  $Data_1$  تا  $Data_3$  برای خواندن

از LCD در خطوط  $D_4$  تا  $D_7$  برای روشن کردن، استفاده می‌شود.

نکته ۸

در حالت نوشتن LCD باید پایه RS، به پایین نگه داشته شود (مغز شود).  
به خواندن از LCD، به  $R_S$  بالا ~ (مغز شود، 5 ولت)

از آن جا که با وجود حافظه می‌تواند در حالت روشن کار بر نامه نویسی نماید به خواندن از LCD نیز با استفاده از پایه  $D_{B_3}$  استفاده می‌شود. در پایه  $R_{W}$  نیز به GND، متصل است. (مغز شود)

۲

درس سخته برای راه اندازی LCD، معرفی کردن آن است. برای این کار باید عرض می‌دهیم که LCD با استفاده از دستوری، هر کوان LCD، راه اندازی کرد.

config LCD = LCD name

LCD name = 16 \* 2 یا 16 \* 4

config LCD = 16 \* 2

معرفی پایه های میکرو که LCD بیان می‌کند و در این جا توضیح می‌دهیم. برای 4 پایه،

۳

config LCD pin = pin و  $D_{B_4} = pin \times y$  و  $D_{B_5} = pin \times y$  و  $D_{B_6} = pin \times y$

$D_{B_7} = pin \times y$  و  $R_S = pin \times y$  و  $E = pin \times y$

به X فایبرت در ی که پایه آن پورت از میکرو است.  
 $D_7 \_ D_6 \_ D_5 \_ D_4 \_ D_3 \_ D_2 \_ D_1 \_ D_0 = X$   
 $7 \_ 6 \_ 5 \_ 4 \_ 3 \_ 2 \_ 1 \_ 0 = Y$

بعد از راه اندازی LCD ثبت می‌کند در آن است. برای نوشتن روی LCD از دستوری استفاده می‌کنیم.

LCD X که X هر کوانه در می‌تواند باشد. البته در محدود کاراکترهای  
است.

۴

برای پاک کردن LCD از دستوری، استفاده می‌شود. دارد خود نویسه تان، از دستوری

6) Locat X, Y

LCD دہلی، نوٹ

Locat 1, 2  
Lct "123"

\* X اور Y کی پیمائش

نوٹ: LCD کی سائز  $16 \times 2$  ہے  $\Rightarrow X_{max} = 2$  ،  $Y_{max} = 16$  ہے

7

بالترتیب ذیلہ کمانوں کے ساتھ LCD پر لکھ کر دیکھیں کہ جاسے یا نہیں

- upper line : بالترتیب لکھ کر دیکھیں کہ لکھ سکتے ہیں
- lower line : بالترتیب لکھ کر دیکھیں کہ لکھ سکتے ہیں
- Home : بالترتیب لکھ کر دیکھیں کہ لکھ سکتے ہیں
- third line : بالترتیب لکھ کر دیکھیں کہ لکھ سکتے ہیں
- fourth line : بالترتیب لکھ کر دیکھیں کہ لکھ سکتے ہیں

\$ regfile = "M8 def.dat" : لکھیں

\$ crystal = 8000000

config LED = 16 \* 2

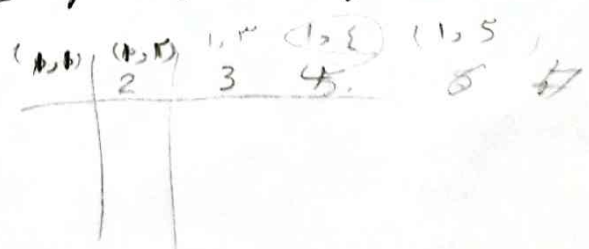
config LED pin = pin , D<sub>B4</sub> = pinC.0 , D<sub>B5</sub> = pinC.1 , D<sub>B6</sub> = pinC.2

D<sub>B7</sub> = pinC.3 , R<sub>S</sub> = pinC.4 , E = pinC.5

```

Do
Locate 1, 2
LED "234"
wait 10
cls

```



Locate 1.5

LED "5678"

wait 4

loop

end