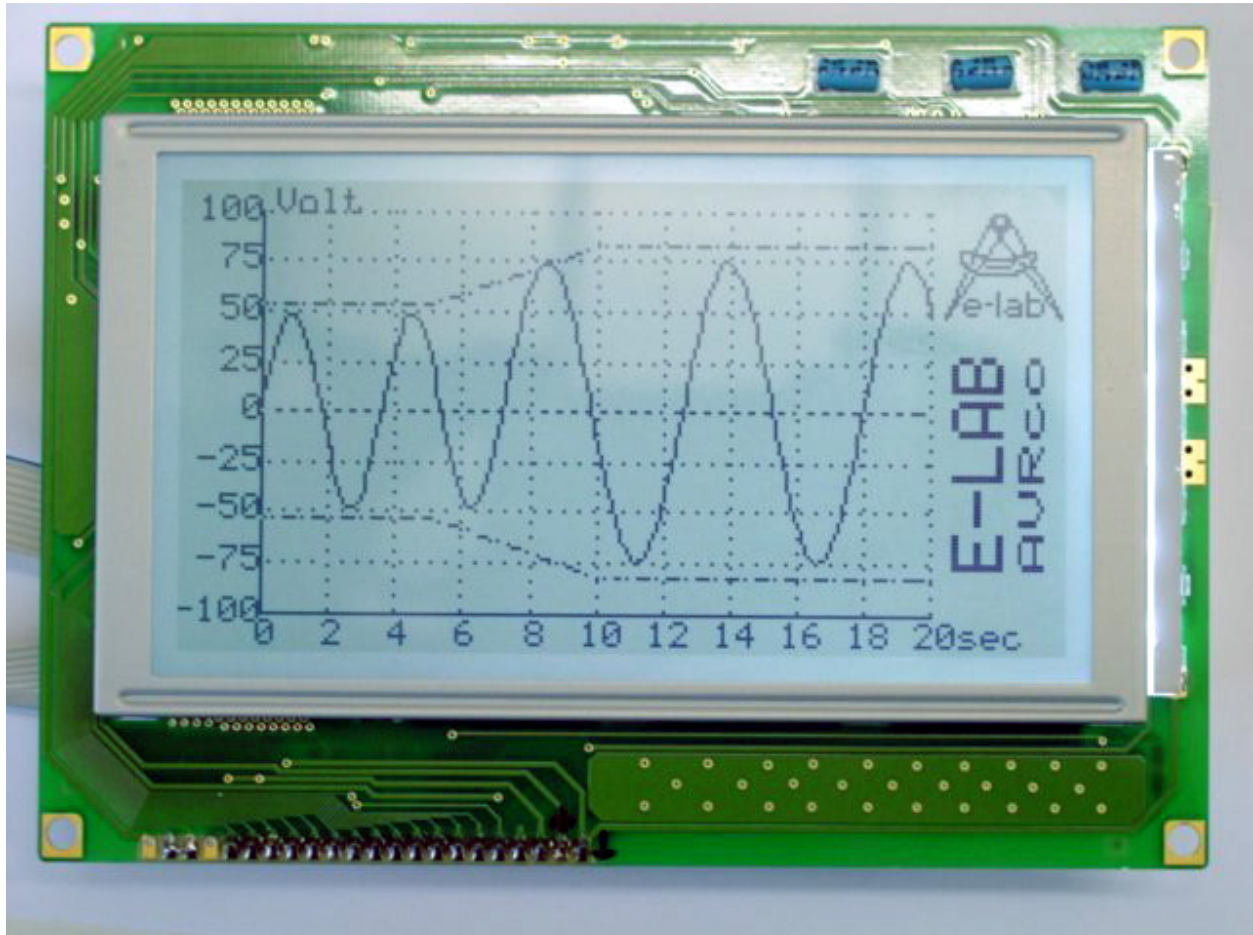


به نام خدا

آموزش میکروکنترلر AVR به زبان C (WinAVR)

(جلسه سوم)



مقدمه:

در جلسه قبل به راه اندازی LCD کاراکتری ۲ در ۱۶ معمولی پرداختیم. در این جلسه تصمیم داریم به معرفی توابع، کتابخانه ها و درایورهای مورد نیاز برای راه اندازی LCD های گرافیکی تک رنگ ۱۲۸ در ۶۴ مبتنی بر چیپ KS0108 و HD61202 که معمولاً با نام تجاری TS12864A-2 شناخته می شود بپردازیم. این کتابخانه با LCD نازک ۱۲۸ در ۶۴ با چیپ SAMSUNG نیز همخوانی داشته و تست شده است.



پس از ویرایش فایلها در پوشه glcd به منوی start و سپس WinAVR رفته و Programmer's Notepad را باز کنید. سپس کدهای زیر را در پنجره پروژه جدید تایپ کرده و برنامه را با نام main.c در پوشه glcd ذخیره و کامپایل نمایید. در صورتی که همه چیز به خوبی پیش رفته باشد فایل hex در پوشه glcd ایجاد می شود که می توانید از طریق کامپایلر Bascom و یا هر نرم افزار دیگری آن را داخل میکروکنترلر پروگرام کنید. توجه داشته باشد که فرکانس میکرو را روی ۸ مگاهرتز RC داخلی تنظیم کنید. همچنین پورت JTAG را غیر فعال نمایید تا پورت C قابل استفاده شود.

```
main.c
#include <avr/io.h>
#include <avr/interrupt.h>
#include "rprintf.h"
#include "glcd.h"

int main (void)
{
    glcdInit();
    rprintfInit(glcdwriteChar);
    rprintfProgStrM("GLCD,ic");

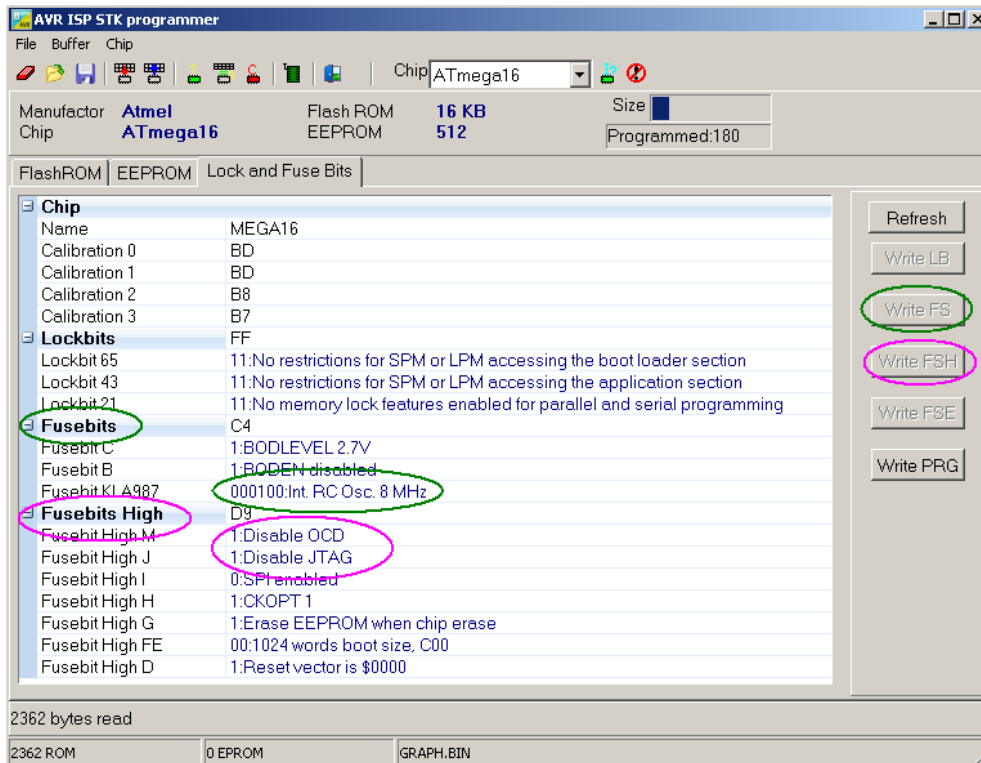
    glcdSetAddress(4,LINE2);
    glcdwriteChar('H');

    glcdSetAddress(4,LINE3);
    rprintfProgStrM("line 3");

    //glcdClearScreen();
    //glcdClearDot(0,0);
    //glcdSetDot(1,1);
    //glcdSetAddress(4,LINE3);
    //glcdPutStr("HD61202/3 controller");

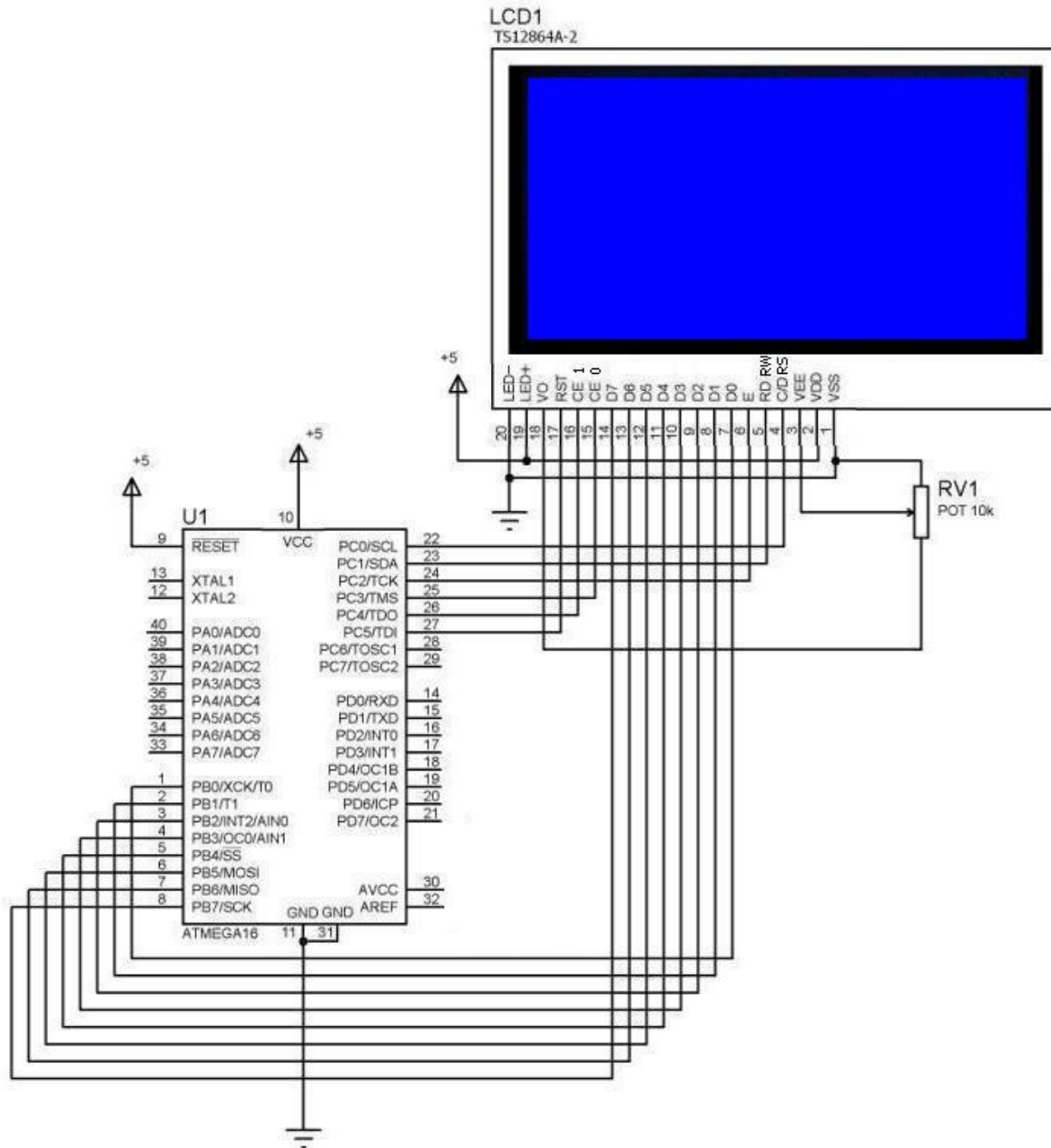
    glcdRectangle(100, 30, 20, 25);
    glcdCircle(30,45,10);
    //glcdStartLine(5);
}
```

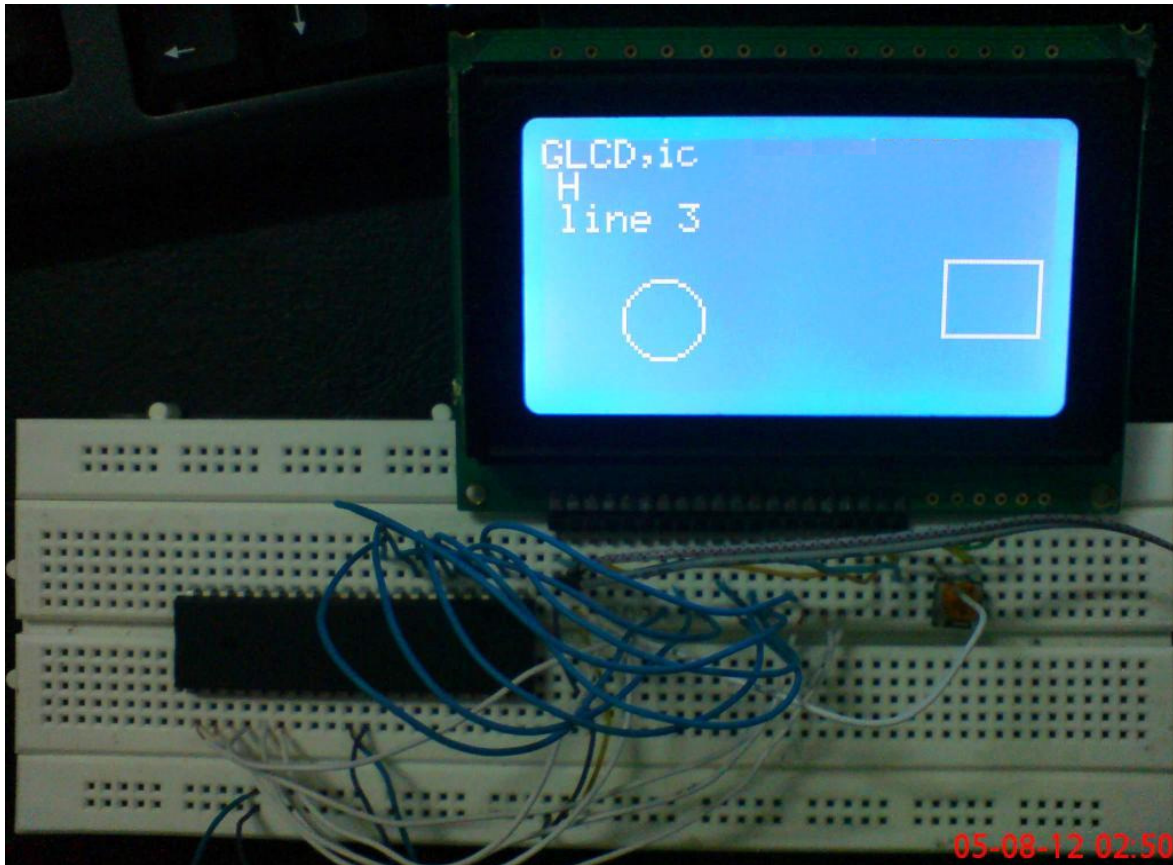
برای غیر فعال کردن پورت JTAG و تنظیم فرکانس CPU میکرو روی ۸ مگاهرتز داخلی RC اگر از محیط بیسکام برای پروگرام کردن میکرو استفاده می کنید میتوانید مطابق شکل زیر پس از پروگرام کردن، وارد سربرگ فیوزبیت ها شده و تنظیمات را انجام دهید. پس از انجام تنظیمات هر گروه روی نام آن گروه کلیک کنید و دکمه مربوطه به آن را که فعال خواهد شد کلیک کنید تا فیوز بیت مربوطه در میکرو تنظیم شود. موقع تنظیم فیوزبیت کلاک دقت کنید که سایر منابع رزرو شده را انتخاب نکنید چرا که میکرو بلا استفاده شده و براحتی قابل بازگشت نخواهد بود.



## شماتیک پروژه و راه اندازی فیزیکی

برای تست این پروژه مداری را مطابق شکل زیر بر روی بُرد پیاده کنید و فایل هگز را داخل میکرو بارگزاری نمایید.





تابع `glcd.h` شامل توابع کاربردی بسیار زیادی برای کار با انواع LCD های گرافیکی می باشد. در این پروژه این تابع به همراه درایور `KS0108` استفاده شده است. در واقع `KS0108.h` در سطح سخت افزار عمل کرده و `glch.h` در لایه بالاتر و `rprintf.h` در لایه کاربر فعالیت می کند. کاربری که با دستورات `rprintf` کار می کند انتظار دارد که متن هایش در نمایشگر نشان داده شود. در این پروژه همچنین می توان به طور مستقیم از توابع `glcd.h` نیز استفاده کرد. توابعی مثل تابع رسم دایره و چهار ضلعی از این دسته اند.



برای مشاهده سایر توابع، هدرفایلهای پروژه را بررسی نمایید.



از مزیت های زبان C باز بودن معماری آن است؛ بدین صورت که بر خلاف Bascom و یا سایر کامپایلرهای "بسته" به راحتی می توانید توابع را ویرایش کرده و در صورت نیاز از توابع دیگران در پروژه های خود استفاده نمایید، البته قبل از استفاده از توابع سایرین حتماً به قوانین کپی رایت توجه کنید و لایسنس آن را مطالعه نمایید.

\*\*\*

جلسه چهاردهم در این قسمت به پایان می رسد. در ۷ جلسه بعدی که جلسات پایانی WinAVR می باشد به ترتیب به موضوعات زیر خواهیم پرداخت:

- ۴- پورت سریال و توابع مربوطه
- ۵- مبدل آنالوگ به دیجیتال و توابع
- ۶- تایمر / کانتر و توابع کاربردی
- ۷- مموری کارت SD/MMC
- ۸- سیستم فایل و توابع کاربردی
- ۹- مباحث تکمیلی وین ای وی آر
- ۱۰- مثال ها و پروژه های کاربردی

ادامه دارد...

