

Arduino Sketches (05-tue22-12). The ones of interest are highlighted in color. There are others but these are the main ones.

<CODE1--12.5.1: 8pin mode-2ports>:

```
// different from the textbook*****

#define F_CPU 8000000UL

#include <avr/io.h>
#include <util/delay.h>

#define LCD_DPRT PORTC // does NOT work with port A
#define LCD_DDDR DDRC // ONLY with all B or C or a
#define LCD_DPIN PINC // combination of the two!!!
#define LCD_CPRT PORTB
#define LCD_CDDR DDRB
#define LCD_CPIN PINB
#define LCD_RS 0
#define LCD_RW 1
#define LCD_EN 2

//*****
void delay_us(unsigned int d)
{
    _delay_us(d);
}

//*****
void lcdCommand( unsigned char cmdnd )
{
    LCD_DPRT = cmdnd;
    LCD_CPRT &= ~(1<<LCD_RS);
    LCD_CPRT &= ~(1<<LCD_RW);
    LCD_CPRT |= (1<<LCD_EN);
    delay_us(1);
    LCD_CPRT &= ~(1<<LCD_EN);
    delay_us(100);
}

//*****
void lcdData( unsigned char data )
{
    LCD_DPRT = data;
    LCD_CPRT |= (1<<LCD_RS);
    LCD_CPRT &= ~(1<<LCD_RW);
    LCD_CPRT |= (1<<LCD_EN);
    delay_us(1);
    LCD_CPRT &= ~(1<<LCD_EN);
    delay_us(100);
}

//*****

void lcd_init()
{
    LCD_DDDR = 0xFF;
    LCD_CDDR = 0xFF;

    LCD_CPRT &= ~(1<<LCD_EN);
    delay_us(2000);
    lcdCommand(0x38);
    lcdCommand(0x0E);
    lcdCommand(0x01);
    delay_us(2000);
    lcdCommand(0x06);
}
```

```

//*****
void lcd_gotoxy(unsigned char x, unsigned char y)
{
    unsigned char firstCharAdr[]={0x80,0xC0,0x94,0xD4}; //table 12-5
    lcdCommand(firstCharAdr[y-1] + x - 1);
    delay_us(100);
}

//*****
void lcd_print( char * str )
{
    unsigned char i = 0 ;
    while(str[i]!=0)
    {
        lcdData(str[i]);
        i++ ;
    }
}

//*****
int main(void)
{
    lcd_init();
    lcd_gotoxy(1,1);
    lcd_print("The world is but");
    lcd_gotoxy(1,2);
    lcd_print("one country");

    while(1);
    return 0;
}
</CODE1--12.5.1: 8pin mode-2ports>

```

CODE2--12.5.2: 8pin mode-1port possibility1 >

```

// different from the textbook*****
#define F_CPU 8000000UL
#include <avr/io.h>
#include <util/delay.h>
#define LCD_DPRT PORTB // does NOT work with port A
#define LCD_DDDR DDRB // ONLY with all B or C or a
#define LCD_DPIN PINB // combination of the two!!!
#define LCD_CPRT PORTB
#define LCD_CDDR DDRB
#define LCD_CPIN PINB
#define LCD_RS 0
#define LCD_RW 1
#define LCD_EN 2

//*****
void delay_us(unsigned int d)
{
    _delay_us(d);
}

//*****
void lcdCommand( unsigned char cmd )
{
    LCD_DPRT = cmd;
    LCD_CPRT &= ~(1<<LCD_RS);
    LCD_CPRT &= ~(1<<LCD_RW);
    LCD_CPRT |= (1<<LCD_EN);
    delay_us(1);
    LCD_CPRT &= ~(1<<LCD_EN);
    delay_us(100);
}

```

```

}

//*****
void lcdData( unsigned char data )
{
    LCD_DPRT = data;
    LCD_CPRT |= (1<<LCD_RS);
    LCD_CPRT &= ~(1<<LCD_RW);
    LCD_CPRT |= (1<<LCD_EN);
    delay_us(1);
    LCD_CPRT &= ~(1<<LCD_EN);
    delay_us(100);
}

//*****
void lcd_init()
{
    LCD_DDDR = 0xFF;
    LCD_CDDR = 0xFF;

    LCD_CPRT &= ~(1<<LCD_EN);
    delay_us(2000);
    lcdCommand(0x38);
    lcdCommand(0x0E);
    lcdCommand(0x01);
    delay_us(2000);
    lcdCommand(0x06);
}

//*****
void lcd_gotoxy(unsigned char x, unsigned char y)
{
    unsigned char firstCharAdr[]={0x80,0xC0,0x94,0xD4}; //table 12-5
    lcdCommand(firstCharAdr[y-1] + x - 1);
    delay_us(100);
}

//*****
void lcd_print( char * str )
{
    unsigned char i = 0 ;
    while(str[i] != 0)
    {
        lcdData(str[i]);
        i++;
    }
}

//*****
int main(void)
{
    lcd_init();
    lcd_gotoxy(1,1);
    lcd_print("The world is but");
    lcd_gotoxy(1,2);
    lcd_print("one country");
    while(1);
    return 0;
}
</CODE2--12.5.2: 8pin mode-1port possibility1 >

```

.....

<CODE3--12.5.3: YES!!!8pin mode-1port, utilizing 12.7 features>

// same as in the text.

#define F_CPU 8000000UL

#include <avr/io.h>

#include <util/delay.h>

#define LCD_PRT PORTB

#define LCD_DDR DDRB

#define LCD_PIN PINB

#define LCD_RS 0

#define LCD_RW 1

#define LCD_EN 2

void delay_us(int d)

```
{
    _delay_us(d);
}
```

void lcdCommand(unsigned char cmd)

```
{
    LCD_PRT = (LCD_PRT & 0x0F) | (cmd & 0xF0);
    LCD_PRT &= ~(1<<LCD_RS);
    LCD_PRT &= ~(1<<LCD_RW);
    LCD_PRT |= (1<<LCD_EN);
    delay_us(1);
    LCD_PRT &= ~(1<<LCD_EN);
    delay_us(100);
}
```

void lcdData(unsigned char data)

```
{
    LCD_PRT = (LCD_PRT & 0x0F) | (data & 0xF0);
    LCD_PRT |= (1<<LCD_RS);
    LCD_PRT &= ~(1<<LCD_RW);
    LCD_PRT |= (1<<LCD_EN);
    delay_us(1);
    LCD_PRT &= ~(1<<LCD_EN);
    delay_us(100);
}
```

void lcd_init()

```
{
    LCD_DDR = 0xFF;
    LCD_PRT &= ~(1<<LCD_EN);
    delay_us(2000);
    lcdCommand(0x38);
    lcdCommand(0x0E);
    lcdCommand(0x01);
    delay_us(2000);
    lcdCommand(0x06);
}
```

void lcd_gotoxy(unsigned char x, unsigned char y)

```
{ //table 12-5
    unsigned char firstCharAdr[] = {0x80, 0xC0, 0x94, 0xD4};
    lcdCommand(firstCharAdr[y-1] + x - 1);
    delay_us(100);
}
```

void lcd_print(char * str)

```
{
    unsigned char i = 0 ;

    while(str[i]!=0)
    {
        lcdData(str[i]);
        i++ ;
    }
}
```

```

int main(void)
{
    lcd_init();
    lcd_gotoxy(1,1);
    lcd_print("The world is but");
    lcd_gotoxy(1,2);
    lcd_print("one country");
    while(1);
    return 0;
}

```

</CODE3--12.5.3: YES!!!8pin mode-1port, utilizing 12.7 features>

.....

<CODE4--12.5.4: 8pin mode-2ports DON>

// different from the textbook*****

```

#define F_CPU 8000000UL

```

```

#include <avr/io.h>

```

```

#include <util/delay.h>

```

```

#define LCD_DPRT PORTD // does NOT work with port A

```

```

#define LCD_DDDR DDRD // ONLY with all B or C or a

```

```

#define LCD_DPIN PIND // combination of the two!!!

```

```

#define LCD_CPRT PORTB

```

```

#define LCD_CDDR DDRB

```

```

#define LCD_CPIN PINB

```

```

#define LCD_RS 0

```

```

#define LCD_RW 1

```

```

#define LCD_EN 2

```

//*****

```

void delay_us(unsigned int d)

```

```

{
    _delay_us(d);
}

```

//*****

```

void lcdCommand( unsigned char cmnd )

```

```

{
    LCD_DPRT = cmnd;
    LCD_CPRT &= ~(1<<LCD_RS);
    LCD_CPRT &= ~(1<<LCD_RW);
    LCD_CPRT |= (1<<LCD_EN);
    delay_us(1);
    LCD_CPRT &= ~(1<<LCD_EN);
    delay_us(100);
}

```

//*****

```

void lcdData( unsigned char data )

```

```

{
    LCD_DPRT = data;
    LCD_CPRT |= (1<<LCD_RS);
    LCD_CPRT &= ~(1<<LCD_RW);
    LCD_CPRT |= (1<<LCD_EN);
    delay_us(1);
    LCD_CPRT &= ~(1<<LCD_EN);
    delay_us(100);
}

```

//*****

```

void lcd_init()

```

```

{

```

```

LCD_DDDR = 0xFF;
LCD_CDDR = 0xFF;

LCD_CPRT &=~(1<<LCD_EN);
delay_us(2000);
lcdCommand(0x38);
lcdCommand(0x0E);
lcdCommand(0x01);
delay_us(2000);
lcdCommand(0x06);
}

//*****
void lcd_gotoxy(unsigned char x, unsigned char y)
{
    unsigned char firstCharAdr[]={0x80,0xC0,0x94,0xD4}; //table 12-5
    lcdCommand(firstCharAdr[y-1] + x - 1);
    delay_us(100);
}

//*****
void lcd_print( char * str )
{
    unsigned char i = 0 ;
    while(str[i]!=0)
    {
        lcdData(str[i]);
        i++ ;
    }
}

//*****
int main(void)
{
    lcd_init();
    lcd_gotoxy(1,1);
    lcd_print("The world is but");
    lcd_gotoxy(1,2);
    lcd_print("one country");
    while(1);
    return 0;
}
<CODE4--12.5.4: 8pin mode-2ports DON>

```

<CODE5—12.6: 4bit mode, 1port>

// NOT!!! in text as far as using only PORTC is concerned!!!

```
#include <avr/io.h>
```

```
#define F_CPU 7372800UL
```

```
#include <util/delay.h>
```

```
#define LCD_DPRT PORTC
```

```
#define LCD_DDDR DDRC
```

```
#define LCD_DPIN PINC
```

```
#define LCD_CPRT PORTC // addition. different frm text
```

```
#define LCD_CDDR DDRC // addition. different frm text
```

```
#define LCD_CPIN PINC // addition. different frm text
```

```
#define LCD_RS 0
```

```
#define LCD_RW 1
```

```
#define LCD_EN 2
```

```
void delay_us(int d)
```

```
{
    _delay_us(d);
}
```

```

}

void lcdCommand( unsigned char cmnd )
{
    LCD_DPRT = (LCD_DPRT&0x0F)|(cmnd & 0xF0); //different from text
    LCD_CPRT &= ~(1<<LCD_RS);
    LCD_CPRT &= ~(1<<LCD_RW);
    LCD_CPRT |= (1<<LCD_EN);
    delay_us(1);
    LCD_CPRT &= ~(1<<LCD_EN);
    delay_us(100);
    LCD_DPRT = (LCD_DPRT&0x0F)|(cmnd<<4);    //different from text
    LCD_CPRT |= (1<<LCD_EN);
    delay_us(1);
    LCD_CPRT &= ~(1<<LCD_EN);
    delay_us(100);
}

void lcdData( unsigned char data )
{
    LCD_DPRT = (LCD_DPRT&0x0F)|(data & 0xF0); //different from text
    LCD_CPRT |= (1<<LCD_RS);
    LCD_CPRT &= ~(1<<LCD_RW);
    LCD_CPRT |= (1<<LCD_EN);
    delay_us(1);
    LCD_CPRT &= ~(1<<LCD_EN);
    LCD_DPRT = (LCD_DPRT&0x0F)|(data<<4);    //different from text
    LCD_CPRT |= (1<<LCD_EN);
    delay_us(1);
    LCD_CPRT &= ~(1<<LCD_EN);
    delay_us(100);
}

void lcd_init()
{
    LCD_DDDR = 0xFF;
    LCD_CDDR = 0xFF;
    LCD_CPRT &= ~(1<<LCD_EN);
    lcdCommand(0x33);
    lcdCommand(0x32);
    lcdCommand(0x28);
    lcdCommand(0x0e);
    lcdCommand(0x01);
    delay_us(2000);
    lcdCommand(0x06);
}

void lcd_gotoxy(unsigned char x, unsigned char y)
{
    unsigned char firstCharAdr[]={0x80,0xC0,0x94,0xD4} ;

    lcdCommand(firstCharAdr[y-1] + x - 1);
    delay_us(100);
}

void lcd_print(char * str )
{
    unsigned char i = 0 ;

    while(str[i]!=0)
    {
        lcdData(str[i]);
        i++ ;
    }
}

int main(void)
{

```

```

lcd_init();
lcd_gotoxy(1,1);
lcd_print("The world is but");
lcd_gotoxy(1,2);
lcd_print("one country");

```

```

while(1);

```

```

return 0;
}

```

</CODE5—12.6: 4bit mode, 1port>

<CODE6—12.7.1: 4bit mode, 1port>

// different as in the text.

```

#define F_CPU 8000000UL
#include <avr/io.h>
#include <util/delay.h>
#define LCD_PRT PORTD // using port D instead of A.
#define LCD_DDR DDRD
#define LCD_PIN PIN_D
#define LCD_RS 0
#define LCD_RW 1
#define LCD_EN 2

```

```

void delay_us(int d)
{
    _delay_us(d);
}

```

```

void delay_ms(int d)
{
    _delay_ms(d);
}

```

```

void lcdCommand( unsigned char cmnd ){
    LCD_PRT = (LCD_PRT & 0x0F) | (cmnd & 0xF0);
    LCD_PRT &= ~(1<<LCD_RS);
    LCD_PRT &= ~(1<<LCD_RW);
    LCD_PRT |= (1<<LCD_EN);
    delay_us(1);
    LCD_PRT &= ~(1<<LCD_EN);

```

```

    delay_us(20);

```

```

    LCD_PRT = (LCD_PRT & 0x0F) | (cmnd << 4);
    LCD_PRT |= (1<<LCD_EN);
    delay_us(1);
    LCD_PRT &= ~(1<<LCD_EN);
}

```

```

void lcdData( unsigned char data )
{
    LCD_PRT = (LCD_PRT & 0x0F) | (data & 0xF0);
    LCD_PRT |= (1<<LCD_RS);
    LCD_PRT &= ~(1<<LCD_RW);
    LCD_PRT |= (1<<LCD_EN);
    delay_us(1);
    LCD_PRT &= ~(1<<LCD_EN);

    LCD_PRT = (LCD_PRT & 0x0F) | (data << 4);
    LCD_PRT |= (1<<LCD_EN);
    delay_us(1);
    LCD_PRT &= ~(1<<LCD_EN);
}

```



```

void lcd_init(){
    LCD_DDR = 0xFF;

    LCD_PRT &=~(1<<LCD_EN);
    delay_us(2000);
    lcdCommand(0x33);
    delay_us(100);
    lcdCommand(0x32);
    delay_us(100);
    lcdCommand(0x28);
    delay_us(100);
    lcdCommand(0x0e);
    delay_us(100);
    lcdCommand(0x01);
    delay_us(2000);
    lcdCommand(0x06);
    delay_us(100);
}

void lcd_gotoxy(unsigned char x, unsigned char y)
{ //table 12-5
    unsigned char firstCharAdr[] = {0x80, 0xC0, 0x94, 0xD4};

    lcdCommand(firstCharAdr[y-1] + x - 1);
    delay_us(100);
}

void lcd_print( char * str )
{
    unsigned char i = 0 ;

    while(str[i]!=0)
    {
        lcdData(str[i]);
        i++ ;
    }
}

int main(void)
{
    lcd_init();
    while(1)
    {
        lcd_gotoxy(1,1);
        lcd_print("The world is but");
        lcd_gotoxy(1,2);
        lcd_print("one country  ");
        delay_ms(1000);
        lcd_gotoxy(1,1);
        lcd_print("and mankind its ");
        lcd_gotoxy(1,2);
        lcd_print("citizens  ");
        delay_ms(1000);
    }
    return 0;
}

```

</CODE6—12.7.1: 4bit mode, 1port>

.....

<CODE7: Hello World>

```
/*  
LiquidCrystal Library - Hello World
```

Demonstrates the use a 16x2 LCD display. The LiquidCrystal library works with all LCD displays that are compatible with the Hitachi HD44780 driver. There are many of them out there, and you can usually tell them by the 16-pin interface.

This sketch prints "Hello World!" to the LCD and shows the time.

The circuit:
* LCD RS pin to digital pin 12
* LCD Enable pin to digital pin 11
* LCD D4 pin to digital pin 5
* LCD D5 pin to digital pin 4
* LCD D6 pin to digital pin 3
* LCD D7 pin to digital pin 2
* LCD R/W pin to ground
* 10K resistor:
* ends to +5V and ground
* wiper to LCD VO pin (pin 3)

Library originally added 18 Apr 2008
by David A. Mellis
library modified 5 Jul 2009
by Limor Fried (<http://www.ladyada.net>)
example added 9 Jul 2009
by Tom Igoe
modified 22 Nov 2010
by Tom Igoe

This example code is in the public domain.

<http://www.arduino.cc/en/Tutorial/LiquidCrystal>
*/

```
// include the library code:  
#include <LiquidCrystal.h>
```

```
// initialize the library with the numbers of the interface pins  
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
```

```
void setup() {  
  // set up the LCD's number of columns and rows:  
  lcd.begin(16, 2);  
  // Print a message to the LCD.  
  lcd.print("hello, world!");  
}
```

```
void loop() {  
  // set the cursor to column 0, line 1  
  // (note: line 1 is the second row, since counting begins with 0):  
  lcd.setCursor(0, 1);  
  // print the number of seconds since reset:  
  lcd.print(millis()/1000);  
}
```

</CODE7: Hello World>

.....