

```

#include <avr\io.h>
#include "delays_xmega.h"
#define lcdPort PORTK.OUT
#define lcdPORT_DIR PORTK.DIR
#define lcdE    0x02
#define lcdRS   0x04
#define lcdRW   0x08

// -----
// This function sends a command to the LCD controller.
// -----
void cmd2LCD(unsigned char cmd)
{
    unsigned char temp;
    temp    = cmd;
    cmd    &= 0xF0;
    lcdPort &= ~(lcdRS+lcdRW); // select IR register & pull R/W low
    lcdPort |= lcdE;          // pull E input high
    lcdPort = cmd | lcdE;     // output upper 4 command bits, E, and RS
    cmd    = temp;           // lengthen E signal
    cmd    = cmd >> 3;       // lengthen E signal
    lcdPort &= ~lcdE;        // pull E low to terminate write of upper 4 bits
    cmd    >>= 3;
    cmd    = temp << 4;      // extract lower 4 bits
    lcdPort |= lcdE;        // pull E high
    lcdPort = cmd | lcdE;    // output lower 4 cmd bits, E, and RS
    cmd    = temp;
    cmd    = cmd >> 3;       // length E signal
    lcdPort &= ~lcdE;        // pull E low to terminate write of lower 4 bits
    delay50us();
}
// -----
// This function configures LCD to select 4-bit data bus, 2-line display, 5x8 font,
// turn on display, cursor, and cursor blinking, move cursor right, clear screen.
// move cursor to upper left corner.
// -----
void openLCD(void)
{
    lcdPORT_DIR = 0xFF; // configure Port K for output
    delayby100ms(3);    // wait for LCD internal initialization to complete
    cmd2LCD(0x33);
    delayby1ms(5);
    cmd2LCD(0x32);
    delayby1ms(2);
    cmd2LCD(0x28);      // select 4-bit data, 2-line display, 5 x 8 font
    cmd2LCD(0x0F);      // turn on display, cursor, and cursor blinking
    cmd2LCD(0x06);      // move cursor right
    cmd2LCD(0x01);      // clear screen, move cursor to home position
    delayby1ms(2);     // wait until "clear display" command is completed
}
// -----
// This function sends a byte of data to the LCD controller to be displayed on the
// LCD panel.
// -----
void putc2LCD(unsigned char cx)
{
    unsigned char temp;
    temp    = cx;
    lcdPort |= lcdRS;   // select LCD data register
    lcdPort &= ~lcdRW; // select write operation
}

```

```
lcdPort |= lcdE;    // pull E to high
cx      &= 0xF0;    // clear lower 4 bits
lcdPort = cx | (lcdRS + lcdE); // output upper 4 data bits, RS, and E
cx      = temp;    // lengthen E signal
cx      = cx >> 3; // lengthen E signal
lcdPort &= ~lcdE;  // terminate upper 4 bits write
cx      >>= 3;    // make sure tC is longer than 500 us
cx      = temp << 4; // extract lower 4 bits
lcdPort |= lcdE;   // pull E to high
lcdPort = cx | (lcdRS + lcdE); // output lower 4 data bits, RS, and E
cx      = temp;    // lengthen E signal
cx      = cx >> 3; // "
lcdPort &= ~lcdE;  // pull E low to terminate write of lower 4 bits
delay50us();
}
// -----
// This function outputs a string.
// -----
void puts2LCD(unsigned char *ptr)
{
    while(*ptr) {
        putc2LCD(*ptr);
        ptr++;
    }
}
```