I²C to HD44780 compatible LCD bridge

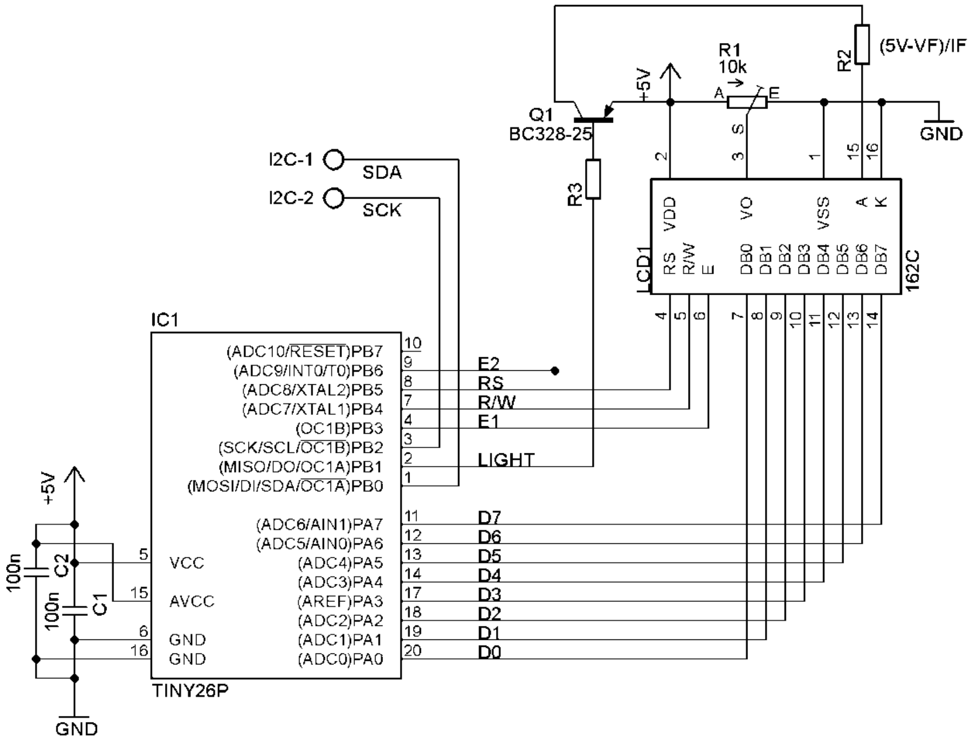
Overview

An AtTiny26 serves as a serial to parallel converter connecting a HD44780 LCD to an I2C bus. Compared to a PCF8574 it provides

- Full 8-bit bus read & write operation. No need to switch the LCD to 4-bit mode.
- Tags (RS, R/W, E) are automatically generated for each byte based on the I²C address. Tags need not be set in multiple write operations.
- Automatic clock stretching and write buffering on LCD busy. No need for timed operations.
- Pre-initializes the LCD on power-up with custom reset values and an optional splash message.
- Automatic PWM backlight control.

The converter operates at 400 kHz maximum I²C clock. 100kHz operation avoids clock stretching completely.

Hardware



E2 allows for dual controller LCDs (4*40, 2*80). R2 is calculated based on the required forward current of the backlight LEDs and the maximum current of Q1. R3 must be calculated to provide sufficient current to drive Q1. 1k should work in most cases.

Firmware

The software consists of 2 files:

- I2C_to_LCD.asm - the ATtiny26 program
- sam.inc - structured assembly macros

It should be easy to port the program to the newer ATtiny261A or even other 8-bit AVR models that have a USI module. **It will not work with a TWI module because these do not allow multiple I^{2}C addresses!**

Fuse bits

When programming the ATtiny26 for the first time, you must also set the correct fuse bits. Refer to your flash tool's documentation on how to set AVR fuses and to the ATtiny26 manual, memory programming, fuse bits. The settings are:

- Clock source: internal RC oscillator - 8MHz
- Startup time: 6 CK + 64 ms *
- BOD must be enabled, 4.0V *

  * This is required to allow the LCD controller to properly power up prior to going through the reset sequence.

Caution! Enable means programmed = 0, disable means unprogrammed = 1.

Fuse high byte 0xF4

-	-	-	RSTDISBL	SPIEN	EESAVE	BODLEVEL	BODEN
1	1	1	1	0	1	0	0

Fuse low byte 0xE4

PLLCK	CKOPT	SUT1	SUT0	CKSEL3	CKSEL2	CKSEL1	CKSEL0
1	1	1	0	0	1	0	0

Changed versus default fuses.

Oscillator calibration byte

The ATtiny26 should be O.K. with the default calibration byte loaded at power on. However, this is the calibration for the 1 MHz RC oscillator. You can force replacing the default by changing cf_osccal to the desired 8 MHz calibration value.

```
;   internal
.EQU  cf_osccal   =0xff        ;0xff  oscillator calibration - 0xff = don't set
```

I^{2}C addressing

You may want to change the address range the LCD bridge occupies on the I2C bus. The default is 0x20 - 0x23. The base address must be on a boundary dividable by 4 or 8 (1 or 2 controller LCD).

```
;   I2C
.EQU  cf_lcd_nmbr =1          ;1     1         or 2 LCD controllers
.EQU  cf_i2c_adr  =0x20       ;0x20  0x20-0x23 or 0x20-027 (0x24-0x27 = enable 2)
```

LCD reset initialization

The initialization command sequence for the LCD controller(s) may be altered from the default. Refer to the LCD's datasheet for information on the available commands.

A splash message can be displayed on the 1st line of the display.

```
;   LCD reset (refer to LCD datasheet)
.EQU  cf_lcd_rs_sf=0b00111000 ; 001 set function: 8 bit, 2 lines, font, x, x
.EQU  cf_lcd_rs_do=0b00001100 ; 00001 on/off: display, cursor, blink
.EQU  cf_lcd_rs_em=0b00000110 ; 000001 entry mode: cursor +/-, disp. shift
; splash message to be displayed by reset
; replace with your message or comment out for no message
#define  splash   " - I2C to LCD - "
```

Backlight configuration

The backlight's intensity and timeout values can be configured. The default intensity after the backlight is triggered by all writes to the LCD is

- full on for the first 10 seconds,
- then 1/8 intensity for another 50 seconds
- and full off after that.

```
;   LCD backlight
.EQU  cf_lcd_on   =0xff       ;0xff  backlight full on after data access
.EQU  cf_lcd_dim  =0x20       ;0x20  backlight dimmed after 1st timeout
.EQU  cf_lcd_off  =0x00       ;0x00  backlight off after 2nd timeout
.EQU  cf_lcd_to1  =10         ;10    timeout 1 in seconds (max 255)
.EQU  cf_lcd_to2  =50         ;50    timeout 2 in seconds (max 255)
```

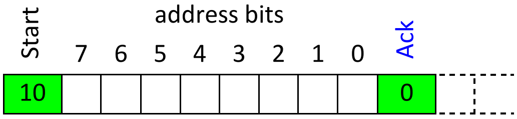
Operation

Addressing

After receiving a start event from the I²C bus the first byte received will be matched with the unit's address. Bits 2-7 will be determined by the configured address in the firmware. The I²C to parallel bus controller will occupy 2 addresses designated by bit 1. It will be used for the LCD's parallel bus and select the proper RS signal to the LCD. Bit 0 will determine the direction of the current I/O and set the R/W control of the parallel bus accordingly.

If the LCD has 2 controller bit 2 of the address selects which controller receives the enable pulse with a byte being transferred.

If the 1st byte received matches the addressing set of the unit and the LCD is not busy with previously transferred bytes it will respond with acknowledge. **Be prepared to retry the I²C bus operation on LCD busy (Ack=1).**



I²C address

7	**6**	**5**	**4**	**3**	**2**	**1**	**0**	Function	Data Bytes
c	c	c	c	c	c			configured fixed address, 1 controller	
						x	x	strobe E1 with any read/write to the LCD	n - to/from LCD
c	c	c	c	c				configured fixed address, 2 controller	
					0	x	x	strobe E1 with any read/write to the LCD	n - to/from LCD CU 1
					1	x	x	strobe E2 with any read/write to the LCD	n - to/from LCD CU 2
					x	0	x	clear RS to LCD	n - command(s) or cursor address
					x	1	x	set RS to LCD	n - display or character data
					x	x	0	clear R/W to LCD	n - write to LCD
					x	x	1	set R/W to LCD	n - read from LCD

LCD registers being accessed with the default base address of 0x20 in a single controller configuration:

- 0x20 = write command register
- 0x21 = read cursor address
- 0x22 = write display data or character generator RAM
- 0x23 = read display data or character generator RAM

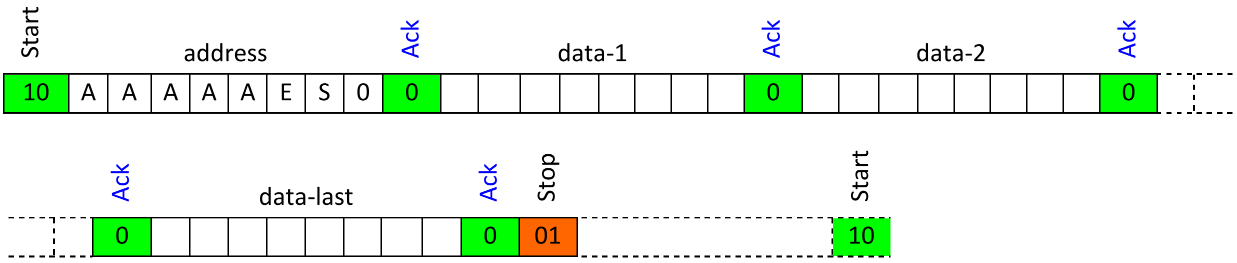
In a dual controller configuration the second controller is accessed at 0x24-0x27. In some environments the I2C address is represented without the R/W-bit (0x10,0x11,...).

Write

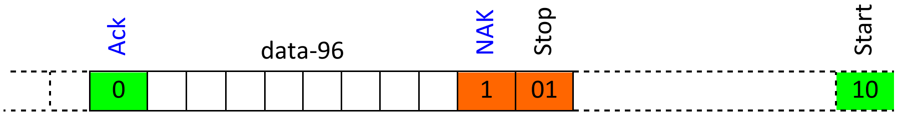
Upon reception of a data byte the unit puts the data into a 96 byte deep buffer and sends acknowledge to the master. After the write of the 96th data byte the bridge sends a NAK to the bus master. Writes > 96 bytes must be split into smaller chunks. The buffer is written to the LCD in the background automatically polling the busy bit. While the buffer is not empty or the LCD signals busy any subsequent start receives a NAK.

When the slave responds with acknowledge to the last byte transferred the master may send a stop or a repeated start condition to end the write to the current LCD register.

Any write will trigger the automatic backlight. A set function command is masked with the default reset configuration to protect the hardware setup of the LCD.



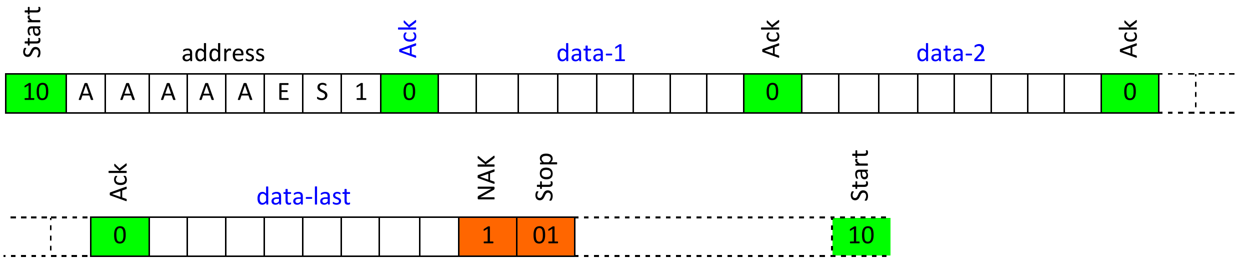
or



Read

- DDRAM / CGRAM address: Upon entering the read state the bridge will poll the busy bit until the LCD controller is no longer busy, then strobe enable, fetch a data byte from the LCD controller and send it to the master.
- LCD status address: Upon entering the read state the bridge will strobe enable, fetch a data byte from the LCD controller and send it to the master. Since you cannot connect to the status register while the LCD is busy the busy bit is always reported as 0.

The master responds with acknowledge if it wants to read more data. A NAK will drop the bridge into idle state and disconnects it from the I²C bus.



Start
address
Ack
data-1
Ack
data-2
Ack
10
A
A
A
A
A
E
S
1
0
0
0
Ack
data-last
NAK
Stop
Start
0
1
01
10