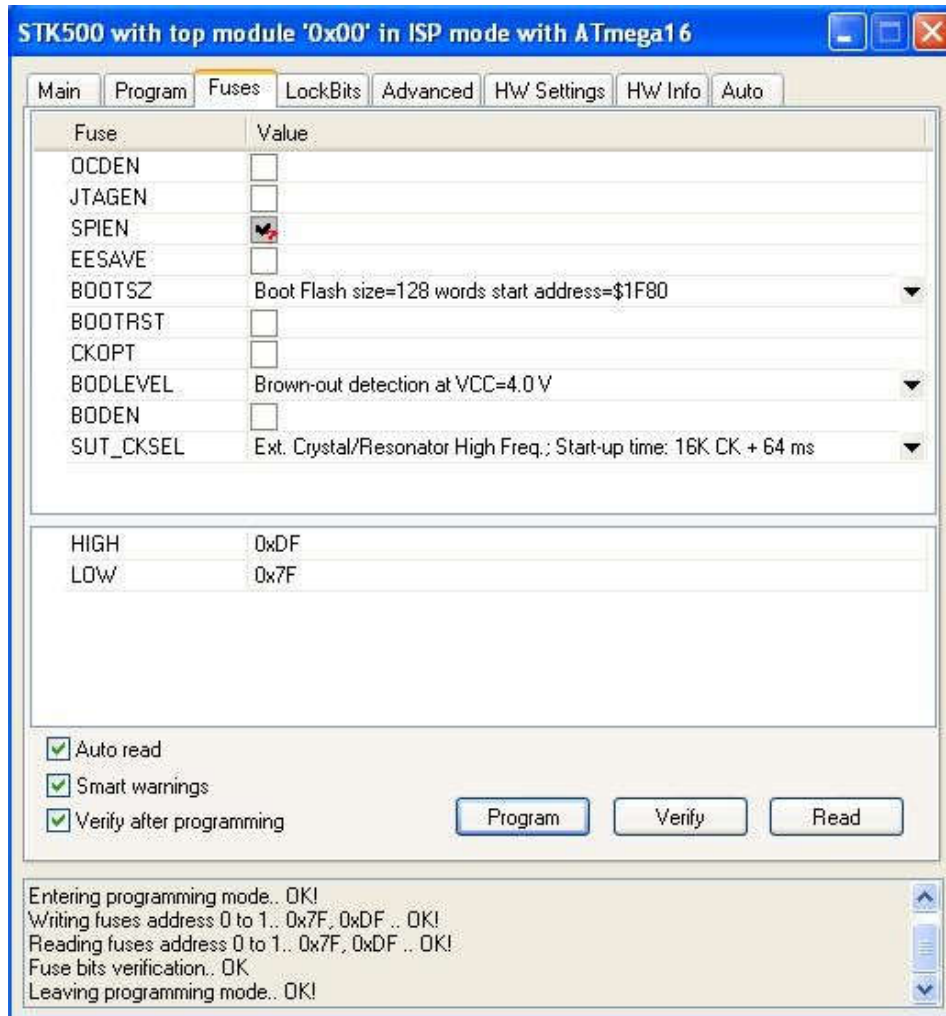
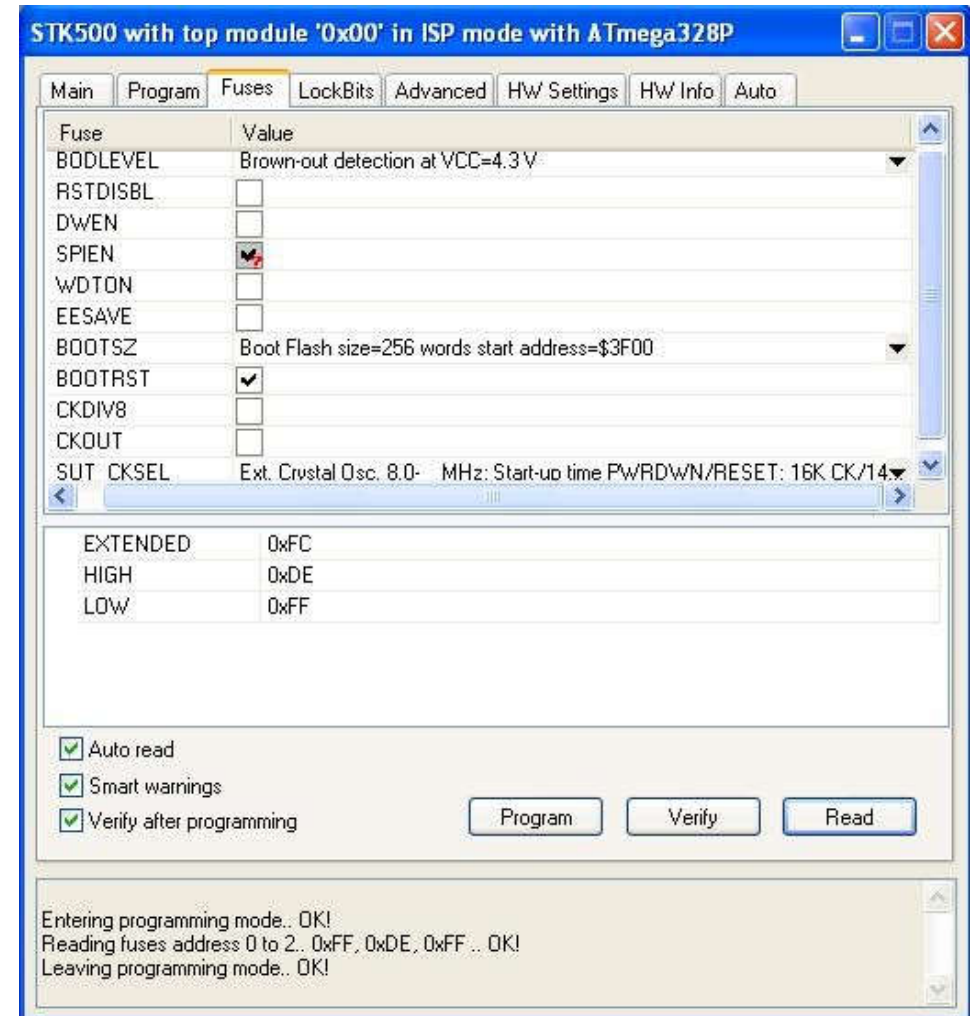


1 Fuses



AVR Studio Fuses Master (ATmega16)

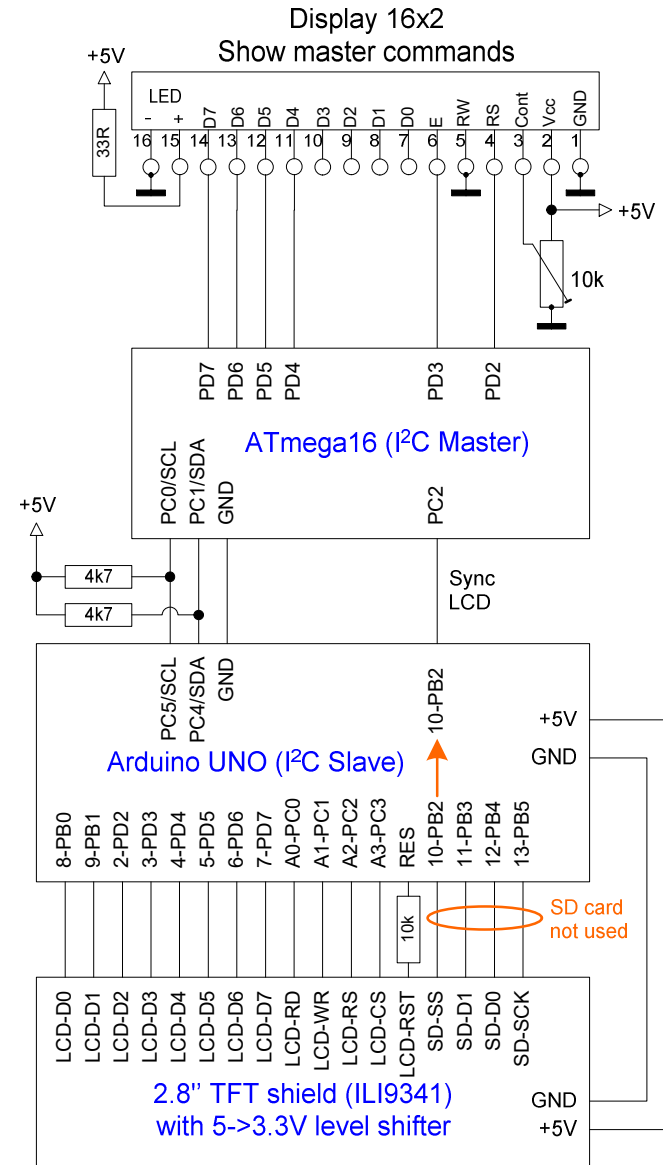
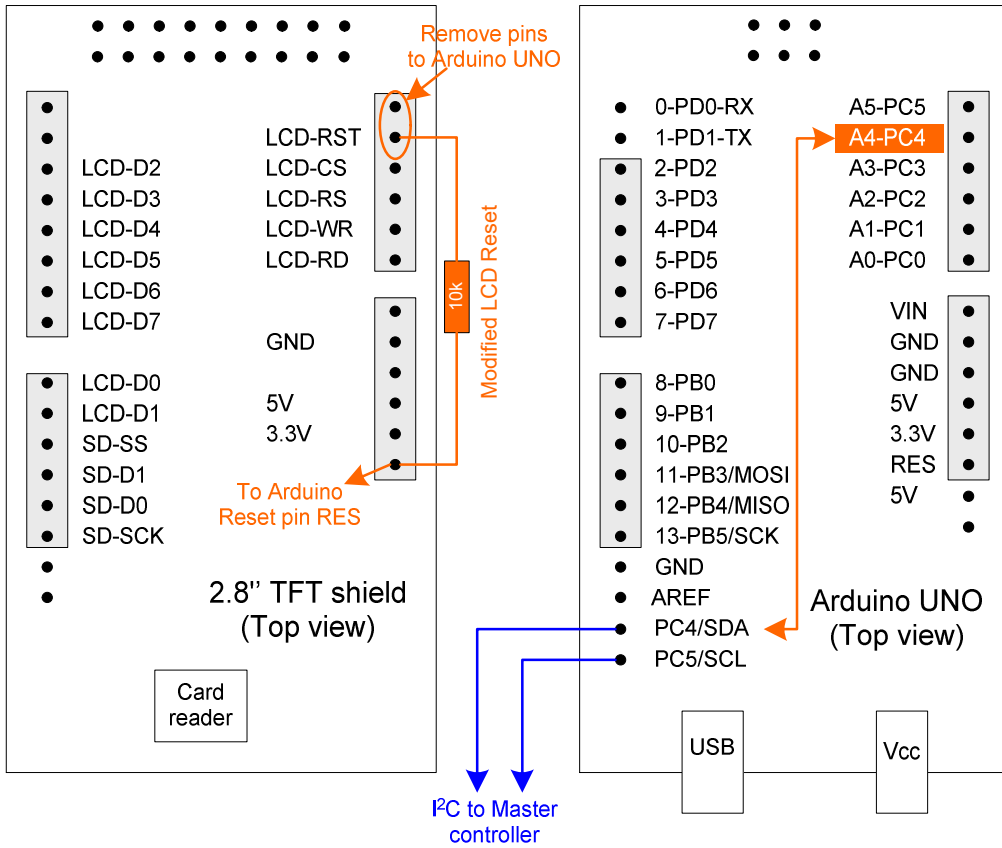


AVR Studio Fuses Slave (ATmega328P)

V2.00 (03/2020): If you use the touchscreen set **EESAVE**.

2 Schematics

TFT shield modifications to enable I²C communication



3 Master Commands

Command	Variable	Type	Meaning
GLCD_Clear(Backcolor)			Clear screen and set background color
	Backcolor	Word	background color (predefined, e.g. "Black", "White", "Red",...)
GLCD_Text(Text,X,Y,Font,Forecolor,Backcolor,Bold) Call GLCD_Text("Hello world", 20, 30, 2, White, Black,2)			Show text string
			Show "Hello world" at (20, 30) white text, bgcolor black
	Text	String	Text to be displayed
	X	Word	X-coordinate (horizontal, landscape 0...319)
	Y	Word	Y-coordinate (vertical, landscape 0...239)
			Startposition (X, Y) is the upper left edge of the text
	Font	Byte	Font type, 1 = 6x8, 2 = 12x16 pixel
	Forecolor	Word	Text color (predefined, e.g. "Black", "White", "Red", "Blue"...)
	Backcolor	Word	Background color (predefined, e.g. "Black", "White", "Red",...)
	Bold	Byte	Font style (1 = normal, 2...4 multiples of 2...4)
GLCD_Line(X1, Y1, X2, Y2, Width, Forecolor) Call GLCD_Line(10, 0, 10, 319, 2, Red)			Show line from (x1, y1) to (x2, y2)
			Show horizontal red line from (10, 0) to (10, 319)
	X1	Word	X1 start x-coordinate (horizontal, landscape 0...319)
	Y1	Word	Y1 start y-coordinate (vertical, landscape 0...239)
	X2	Word	X2 end x-coordinate (horizontal, landscape 0...319)
	Y2	Word	Y2 end y-coordinate (vertical, landscape 0...239)
	Width	Byte	Line Width (number of pixels)
	Forecolor	Word	Line color (predefined, e.g. "Black", "White", "Red", "Blue"...)

GLCD_Circle(X, Y, Radius, Forecolor)			Show circle, center = (X, Y), with Radius
Call GLCD_Circle(160, 120, 50, Green)			Show circle, center = (160, 120), radius = 50, green line
	X	Word	X-coordinate (horizontal, landscape 0...319)
	Y	Word	Y-coordinate (vertical, landscape 0...239)
	Radius	Word	Circle radius (pixels)
	Forecolor	Word	Line color (predefined, e.g. "Black", "White", "Red", "Blue"...)
GLCD_Fill_Circle(X, Y, Radius, Fillcolor)			Show filled circle, center = (X, Y), with Radius
Call GLCD_Fill_Circle(160, 120, 50, Green)			Show circle, center = (160, 120), radius = 50, filled green
	X	Word	X-coordinate (horizontal, landscape 0...319)
	Y	Word	Y-coordinate (vertical, landscape 0...239)
	Radius	Word	Circle radius (pixels)
	Fillcolor	Word	Color fill (predefined, e.g. "Black", "White", "Red", "Blue"...)
GLCD_Box(X, Y, Width, Height, Fill, Fillcolor, Bordercolor)			Show rectangle box at upper left (X, Y) with width/height
Call GLCD_Box(10, 1, 100, 30, 2, Yellow, Blue)			Show box at upper left (10, 1), filled yellow, border blue
	X	Word	X-coordinate (horizontal, landscape 0...319)
	Y	Word	Y-coordinate (vertical, landscape 0...239)
			Startposition (X, Y) is the upper left edge of the box
	Width	Word	Horizontal dimension (Pixel)
	Height	Word	Vertical dimension (Pixel)
	Fill	Byte	1 = filled, 2 = filled with Border, 3 = not filled, only border
	Fillcolor	Word	Color fill (predefined, e.g. "Black", "White", "Red", "Blue"...)
	Bordercolor		Border color (predefined, e.g. "Black", "White", "Red", "Blue"...)

GLCD_Touch(bytOpt) (New in V2.00)			Handle slave touchscreen
	bytOpt	Byte	bytOpt = 0: Instruct slave to show touchscreen bytTouchNum, = 1: Read slave touch event
	bytTouchNum	Byte	bytTouchNum: Slave touchscreen number (1, 2...), might be more than one touchscreen.
	bytTouchButton	Byte	Returned if bytOpt=1: Number of the touched button = 0: no button has been touched.
TWI_MasterWrite(bytMsg_Size, bytSLA) (New in V2.00)			Send message buffer bytMsg_Buf to slave
	bytMsg_Size	Byte	Number of bytes to send to slave
	bytSLA	Byte	Slave address (hex)
	bytMsg_Buf	Byte	Message buffer, bytMsg_Buf(1)=slave address
TWI_MasterRead(bytMsg_Size, bytSLA) (New in V2.00)			Read message buffer bytMsg_Buf from slave
	bytMsg_Size	Byte	Number of bytes to read from slave
	bytSLA	Byte	Slave address (hex)
	bytMsg_Buf	Byte	Message buffer, bytMsg_Buf(1)=slave address

4 Slave Commands			
Touch panel functions		(New in V2.00)	
CalibTouch	-	-	Touchscreen calibration routine (portrait mode). Calibration data are stored in EEPROM.
Function CheckEEPROM() As Byte			Check if calibration data are stored in EEPROM. CheckEEPROM = 1: Calibration data found, =0 else. If found, calibration data are read, if not found, CalibTouch is to be called.
ReadTouch(Touch_Level , Opt)			Detect touch event and read x/y coordinate
	Touch_Level	Word	ADC value above which a touch event is detected (~20...40) Some noise is measured by ADC without touch event.
	Opt	Byte	=0: No calculation x/y with calibration data =1: calculate real x/y with calibration data.
SetLCDMode(Mode)		(New in V2.00)	
	Mode	Byte	1=Portrait (Orientation 0, Flexboard bottom) 2=Portrait 180° (Orientation 2, Flexboard top) 3=landscape (Orientation 1, Flexboard left) 4=landscape 180° (Orientation 3, Flexboard right)
I2C/TWI functions		(New in V2.00)	
TWI_SlaveRead	-	-	Read data transferred via I2C from master. Data are stored in byte array bytMsg_Buf (message buffer), bytMsg_Buf(1)=slave address.
TWI_GetData	-	-	Convert received message buffer data into GLCD commands.
TWI_SlaveWrite(bytMsg_Size)			Send data back to master if master has sent a request for an answer with data byte #4 = 1.
	bytMsg_Size	Byte	Number of bytes to send incl. bytMsg_Buf(1)=slave address For touchscreen e.g. (bytMsg_Size=2): bytMsg_Buf(1)=slave address bytMsg_Buf(2)=touched button number.

Disabled			
GLCD_Init			Initialize TFT: Is done by Arduino reset (power on or manually)
Not realized			
Card reader functions			SD_SS is used as sync line, see 2-Schematics