

vvvv--search line in original file TFT_eSPI.cpp (Last update 20/03/20) for more Details		Content analyse made by Jörg Fuchs 23.06.2020
LINE	Funktions . Definitions	Description
34	inline void TFT_eSPI::begin_tft_write(void){	Desc : Start SPI transaction for writes and select TFT
51	inline void TFT_eSPI::end_tft_write(void){	Desc : End transaction for write and deselect TFT
71	inline void TFT_eSPI::begin_tft_read(void){	Desc : Start transaction for reads and select TFT
92	inline void TFT_eSPI::end_tft_read(void){	Desc : End transaction for reads and deselect TFT
114	void TFT_eSPI::spi_begin() {begin_tft_write();}	Desc : Start/end transaction
115	void TFT_eSPI::spi_end() { end_tft_write();}	Desc : Start/end transaction
116	void TFT_eSPI::spi_begin_read() {begin_tft_read();}	Desc : Start/end transaction
117	void TFT_eSPI::spi_end_read() { end_tft_read();}	Desc : Start/end transaction
123	TFT_eSPI::TFT_eSPI(int16_t w, int16_t h)	Desc : Constructor , we must use hardware SPI pins
260	void TFT_eSPI::begin(uint8_t tc)	Desc : Included for backwards compatibility
270	void TFT_eSPI::init(uint8_t tc)	Desc : Reset, then initialise the TFT display registers
433	void TFT_eSPI::setRotation(uint8_t m)	Desc : rotate the screen orientation m = 0-3 or 4-7 for BMP drawing
493	void TFT_eSPI::commandList (const uint8_t *addr)	Desc : Get initialisation commands from FLASH and send to TFT
527	void TFT_eSPI::spiwrite(uint8_t c)	Desc : Write 8 bits to SPI port (legacy support only)
539	void TFT_eSPI::writecommand(uint8_t c)	Desc : Send an 8 bit command to the TFT
558	void TFT_eSPI::writedata(uint8_t d)	Desc : Send a 8 bit data value to the TFT
576	uint8_t TFT_eSPI::readcommand8(uint8_t cmd_function, uint8_t index)	Desc : Read a 8 bit data value from an indexed command register
619	uint16_t TFT_eSPI::readcommand16(uint8_t cmd_function, uint8_t index)	Desc : Read a 16 bit data value from an indexed command register
634	uint32_t TFT_eSPI::readcommand32(uint8_t cmd_function, uint8_t index)	Desc : Read a 32 bit data value from an indexed command register
651	uint16_t TFT_eSPI::readPixel(int32_t x0, int32_t y0)	Desc : Read 565 pixel colours from a pixel
751	void TFT_eSPI::setCallback(getColorCallback getCol)	Desc : Read 565 pixel colours from a pixel
761	void TFT_eSPI::readRect(int32_t x, int32_t y, int32_t w, int32_t h, uint16_t *data)	Desc : Read 565 pixel colours from a defined area
866	void TFT_eSPI::pushRect(int32_t x, int32_t y, int32_t w, int32_t h, uint16_t *data)	Desc : push 565 pixel colours into a defined area
878	void TFT_eSPI::pushImage(int32_t x, int32_t y, int32_t w, int32_t h, uint16_t *data)	Desc : plot 16 bit colour sprite or image onto TFT
922	void TFT_eSPI::pushImage(int32_t x, int32_t y, int32_t w, int32_t h, uint16_t *data, uint16_t transp)	Desc : plot 16 bit sprite or image with 1 colour being transparent
994	void TFT_eSPI::pushImage(int32_t x, int32_t y, int32_t w, int32_t h, const uint16_t *data)	Desc : plot 16 bit image
1038	void TFT_eSPI::pushImage(int32_t x, int32_t y, int32_t w, int32_t h, const uint16_t *data, uint16_t transp)	Desc : plot 16 bit image with 1 colour being transparent
1107	void TFT_eSPI::pushImage(int32_t x, int32_t y, int32_t w, int32_t h, uint8_t *data, bool bpp8, uint16_t *cmap)	Desc : plot 8 bit or 4 bit or 1 bit image or sprite using a line buffer
1265	void TFT_eSPI::pushImage(int32_t x, int32_t y, int32_t w, int32_t h, uint8_t *data, uint8_t transp, bool bpp8, uint16_t *cmap)	Desc : plot 8 or 4 or 1 bit image or sprite with a transparent colour
1493	void TFT_eSPI::setSwapBytes(bool swap)	Desc : Used by 16 bit pushImage() to swap byte order in colours
1503	bool TFT_eSPI::getSwapBytes(void)	Desc : Return the swap byte order for colours
1514	void TFT_eSPI::readRectRGB(int32_t x0, int32_t y0, int32_t w, int32_t h, uint8_t *data)	Desc : Read RGB pixel colours from a defined area
1588	void TFT_eSPI::drawCircle(int32_t x0, int32_t y0, int32_t r, uint32_t color)	Desc : Draw a circle outline
1640	void TFT_eSPI::drawCircleHelper(int32_t x0, int32_t y0, int32_t r, uint8_t cornername, uint32_t color)	Desc : Support function for drawRoundRect()
1682	void TFT_eSPI::fillCircle(int32_t x0, int32_t y0, int32_t r, uint32_t color)	Desc : draw a filled circle
1723	void TFT_eSPI::fillCircleHelper(int32_t x0, int32_t y0, int32_t r, uint8_t cornername, int32_t delta, uint32_t color)	Desc : Support function for fillRoundRect()
1755	void TFT_eSPI::drawEllipse(int16_t x0, int16_t y0, int32_t rx, int32_t ry, uint16_t color)	Desc : Draw a ellipse outline
1807	void TFT_eSPI::fillEllipse(int16_t x0, int16_t y0, int32_t rx, int32_t ry, uint16_t color)	Desc : draw a filled ellipse
1852	void TFT_eSPI::fillScreen(uint32_t color)	Desc : Clear the screen to defined colour
1863	void TFT_eSPI::drawRect(int32_t x, int32_t y, int32_t w, int32_t h, uint32_t color)	Desc : Draw a rectangle outline
1884	void TFT_eSPI::drawRoundRect(int32_t x, int32_t y, int32_t w, int32_t h, int32_t r, uint32_t color)	Desc : Draw a rounded corner rectangle outline
1910	void TFT_eSPI::fillRoundRect(int32_t x, int32_t y, int32_t w, int32_t h, int32_t r, uint32_t color)	Desc : Draw a rounded corner filled rectangle
1932	void TFT_eSPI::drawTriangle(int32_t x0, int32_t y0, int32_t x1, int32_t y1, int32_t x2, int32_t y2, uint32_t color)	Desc : Draw a triangle outline using 3 arbitrary points
1951	void TFT_eSPI::fillTriangle(int32_t x0, int32_t y0, int32_t x1, int32_t y1, int32_t x2, int32_t y2, uint32_t color)	Desc : Draw a filled triangle using 3 arbitrary points
2031	void TFT_eSPI::drawBitmap(int16_t x, int16_t y, const uint8_t *bitmap, int16_t w, int16_t h, uint16_t color)	Desc : Draw an image stored in an array on the TFT
2055	void TFT_eSPI::drawBitmap(int16_t x, int16_t y, const uint8_t *bitmap, int16_t w, int16_t h, uint16_t fgcolor, uint16_t bgcolor)	Desc : Draw an image stored in an array on the TFT
2078	void TFT_eSPI::drawXBitmap(int16_t x, int16_t y, const uint8_t *bitmap, int16_t w, int16_t h, uint16_t color)	Desc : Draw an image stored in an XBM array onto the TFT
2102	void TFT_eSPI::drawXBitmap(int16_t x, int16_t y, const uint8_t *bitmap, int16_t w, int16_t h, uint16_t color, uint16_t bgcolor)	Desc : Draw an XBM image with foreground and background colors
2126	void TFT_eSPI::setCursor(int16_t x, int16_t y)	Desc : Set the text cursor x,y position
2137	void TFT_eSPI::setCursor(int16_t x, int16_t y, uint8_t font)	Desc : Set the text cursor x,y position and font
2149	int16_t TFT_eSPI::getCursorX(void)	Desc : Get the text cursor x position
2158	int16_t TFT_eSPI::getCursorY(void)	Desc : Get the text cursor y position
2168	void TFT_eSPI::setTextSize(uint8_t s)	Desc : Set the text size multiplier
2179	void TFT_eSPI::setTextColor(uint16_t c)	Desc : Set the font foreground colour (background is transparent)
2191	void TFT_eSPI::setTextColor(uint16_t c, uint16_t b)	Desc : Set the font foreground and background colour
2202	void TFT_eSPI::setPivot(int16_t x, int16_t y)	Desc : Set the pivot point on the TFT
2213	int16_t TFT_eSPI::getPivotX(void)	Desc : Get the x pivot position
2223	int16_t TFT_eSPI::getPivotY(void)	Desc : Get the y pivot position

vvvv---search line in original file TFT_eSPI.cpp (Last update 20/03/20) for more Details			Content analyse made by Jörg Fuchs 23.06.2020
LINE		Funktions . Definitions	Description
2233	void TFT_eSPI::	setBitmapColor(uint16_t c, uint16_t b)	Desc : Set the foreground foreground and background colour
2245	void TFT_eSPI::	setTextWrap(bool wrapX, bool wrapY)	Desc : Define if text should wrap at end of line
2256	void TFT_eSPI::	setTextDatum(uint8_t d)	Desc : Set the text position reference datum
2266	void TFT_eSPI::	setTextPadding(uint16_t x_width)	Desc : Define padding width (aids erasing old text and numbers)
2275	uint16_t TFT_eSPI::	getTextPadding(void)	Desc : Define padding width (aids erasing old text and numbers)
2284	uint8_t TFT_eSPI::r	getRotation(void)	Desc : Return the rotation value (as used by setRotation())
2293	uint8_t TFT_eSPI::r	getTextDatum(void)	Desc : Return the text datum value (as used by setTextDatum())
2304	int16_t TFT_eSPI::	width(void)	Desc : Return the pixel width of display (per current rotation)
2314	int16_t TFT_eSPI::	height(void)	Desc : Return the pixel height of display (per current rotation)
2324	int16_t TFT_eSPI::	textWidth(const String& string)	Desc : Return the width in pixels of a string in a given font
2332	int16_t TFT_eSPI::	textWidth(const String& string, uint8_t font)	Desc : Return the width in pixels of a string in a given font
2340	int16_t TFT_eSPI::	textWidth(const char *string)	Desc : Return the width in pixels of a string in a given font
2345	int16_t TFT_eSPI::	textWidth(const char *string, uint8_t font)	Desc : Return the width in pixels of a string in a given font
2419	uint16_t TFT_eSPI::	fontsLoaded(void)	Desc : return an encoded 16 bit value showing the fonts loaded
2429	int16_t TFT_eSPI::	fontHeight(int16_t font)	Desc : return the height of a font (yAdvance for free fonts)
2445	int16_t TFT_eSPI::	fontHeight(void)	Desc : return the height of a font (yAdvance for free fonts)
2454	void TFT_eSPI::	drawChar(int32_t x, int32_t y, uint16_t c, uint32_t color, uint32_t bg, uint8_t size)	Desc : draw a single character in the GLCD or GFXFF font
2598	void TFT_eSPI::	setAddrWindow(int32_t x0, int32_t y0, int32_t w, int32_t h)	Desc : define an area to receive a stream of pixels
2613	void TFT_eSPI::	setWindow(int32_t x0, int32_t y0, int32_t x1, int32_t y1)	Desc : define an area to receive a stream of pixels
2657	void TFT_eSPI::	readAddrWindow(int32_t xs, int32_t ys, int32_t w, int32_t h)	Desc : define an area to read a stream of pixels
2695	void TFT_eSPI::	drawPixel(int32_t x, int32_t y, uint32_t color)	Desc : push a single pixel at an arbitrary position
2739	void TFT_eSPI::	pushColor(uint16_t color)	Desc : push a single pixel
2753	void TFT_eSPI::	pushColor(uint16_t color, uint32_t len)	Desc : push a single colour to "len" pixels
2766	void TFT_eSPI::	startWrite(void)	Desc : begin transaction with CS low, MUST later call endWrite
2776	void TFT_eSPI::	endWrite(void)	Desc : end transaction with CS high
2787	void TFT_eSPI::	writeColor(uint16_t color, uint32_t len)	Desc : raw write of "len" pixels avoiding transaction check
2798	void TFT_eSPI::	pushColors(uint8_t *data, uint32_t len)	Desc : push an array of pixels for 16 bit raw image drawing
2812	void TFT_eSPI::	pushColors(uint16_t *data, uint32_t len, bool swap)	Desc : push an array of pixels, for image drawing
2830	void TFT_eSPI::	drawLine(int32_t x0, int32_t y0, int32_t x1, int32_t y1, uint32_t color)	Desc : draw a line between 2 arbitrary points
2892	void TFT_eSPI::	drawFastVLine(int32_t x, int32_t y, int32_t h, uint32_t color)	Desc : draw a vertical line
2917	void TFT_eSPI::	drawFastHLine(int32_t x, int32_t y, int32_t w, uint32_t color)	Desc : draw a horizontal line
2942	void TFT_eSPI::	fillRect(int32_t x, int32_t y, int32_t w, int32_t h, uint32_t color)	Desc : draw a filled rectangle
2969	uint16_t TFT_eSPI::	color565(uint8_t r, uint8_t g, uint8_t b)	Desc : convert three 8 bit RGB levels to a 16 bit colour value
2979	uint8_t TFT_eSPI::r	color16to8(uint16_t c)	Desc : convert 16 bit colour to an 8 bit 332 RGB colour value
2989	uint16_t TFT_eSPI::	color8to16(uint8_t color)	Desc : convert 8 bit colour to a 16 bit 565 colour value
3006	uint32_t TFT_eSPI::	color16to24(uint16_t color565)	Desc : convert 16 bit colour to a 24 bit 888 colour value
3019	uint32_t TFT_eSPI::	color24to16(uint32_t color888)	Desc : convert 24 bit colour to a 16 bit 565 colour value
3032	void TFT_eSPI::	invertDisplay(bool i)	Desc : invert the display colours i = 1 invert, i = 0 normal
3046	void TFT_eSPI::	setAttribute(uint8_t attr_id, uint8_t param) {	Desc : Sets a control parameter of an attribute
3074	uint8_t TFT_eSPI::r	getAttribute(uint8_t attr_id) {	Desc : Get value of an attribute (control parameter)
3095	uint16_t TFT_eSPI::	decodeUTF8(uint8_t c)	Desc : Serial UTF-8 decoder with fall-back to extended ASCII
3144	uint16_t TFT_eSPI::	decodeUTF8(uint8_t *buf, uint16_t *index, uint16_t remaining)	Desc : Line buffer UTF-8 decoder with fall-back to extended ASCII
3175	uint16_t TFT_eSPI::	alphaBlend(uint8_t alpha, uint16_t fg, uint16_t bg)	Desc : Blend 16bit foreground and background
3201	uint16_t TFT_eSPI::	alphaBlend(uint8_t alpha, uint16_t fg, uint16_t bg, uint8_t dither)	Desc : Blend 16bit foreground and background with dither
3217	uint32_t TFT_eSPI::	alphaBlend24(uint8_t alpha, uint32_t fg, uint32_t bg, uint8_t dither)	Desc : Blend 24bit foreground and background with optional dither
3249	size_t TFT_eSPI::	write(uint8_t utf8)	Desc : draw characters piped through serial stream
3390	int16_t TFT_eSPI::	drawChar(uint16_t uniCode, int32_t x, int32_t y)	Desc : draw a Unicode glyph onto the screen
3396	int16_t TFT_eSPI::	drawChar(uint16_t uniCode, int32_t x, int32_t y, uint8_t font)	Desc : draw a Unicode glyph onto the screen
3663	int16_t TFT_eSPI::	drawString(const String& string, int32_t poX, int32_t poY)	Desc : draw string with padding if it is defined
3671	int16_t TFT_eSPI::	drawString(const String& string, int32_t poX, int32_t poY, uint8_t font)	Desc : draw string with padding if it is defined
3680	int16_t TFT_eSPI::	drawString(const char *string, int32_t poX, int32_t poY)	Desc : draw string with padding if it is defined
3686	int16_t TFT_eSPI::	drawString(const char *string, int32_t poX, int32_t poY, uint8_t font)	Desc : draw string with padding if it is defined
3914	int16_t TFT_eSPI::	drawCentreString(const String& string, int32_t dX, int32_t poY, uint8_t font)	Desc : draw string centred on dX
3922	int16_t TFT_eSPI::	drawCentreString(const char *string, int32_t dX, int32_t poY, uint8_t font)	Desc : draw string centred on dX
3937	int16_t TFT_eSPI::	drawRightString(const String& string, int32_t dX, int32_t poY, uint8_t font)	Desc : draw string right justified to dX
3945	int16_t TFT_eSPI::	drawRightString(const char *string, int32_t dX, int32_t poY, uint8_t font)	Desc : draw string right justified to dX
3960	int16_t TFT_eSPI::	drawNumber(long long_num, int32_t poX, int32_t poY)	Desc : draw a long integer
3968	int16_t TFT_eSPI::	drawNumber(long long_num, int32_t poX, int32_t poY, uint8_t font)	Desc : draw a long integer
3983	int16_t TFT_eSPI::	drawFloat(float floatNumber, uint8_t dp, int32_t poX, int32_t poY)	Desc : drawFloat, prints 7 non zero digits maximum
3988	int16_t TFT_eSPI::	drawFloat(float floatNumber, uint8_t dp, int32_t poX, int32_t poY, uint8_t font)	Desc : drawFloat, prints 7 non zero digits maximum
4058	void TFT_eSPI::	setFreeFont(const GFXfont *f)	Desc : Sets the GFX free font to use
4087	void TFT_eSPI::	setTextFont(uint8_t f)	Desc : Set the font for the print stream
4102	void TFT_eSPI::	setFreeFont(uint8_t font)	Desc : Sets the GFX free font to use
4112	void TFT_eSPI::	setTextFont(uint8_t f)	Desc : Set the font for the print stream
4124	SPIClass& TFT_eSPI::	getSPIInstance(void)	Desc : Get the instance of the SPI class
4134	void TFT_eSPI::	getSetup(setup_t &tft_settings)	Desc : Get the setup details for diagnostic and sketch access