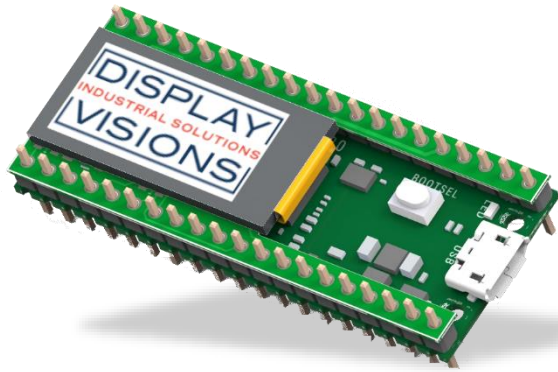
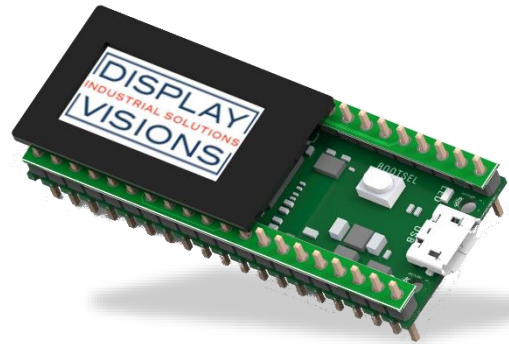


EA RaPicoTFT009

INCL. IPS TFT DISPLAY EA TFT009-81AIXX



EA RaPicoTFT009*



EA RaPicoTFT009TC
with PCAP*

FEATURES

- SHIELD FOR RASPBERRY PI PICO (RP2040)
- IPS DISPLAY WITH 160x80 PIXEL, 0.96"
- INCL: EA TFT009-81AINN WITH 4 WIRE SPI INTERFACE INCL. CONTROLLER ST7735S
- INCL: EA TFT009-81AITC WITH PCAP TOUCH PANEL
- HIGH CONTRAST
- WIDE VIEWING ANGLE DUE TO AACS (ALL ANGLE COLOR STABILITY) DISPLAY
- T_{OP} -20..+70 °C
- OPTIONAL WITH PCAP VIA I²C INTERFACE

ORDERING CODES

- PCB SHIELD WITH IPS TFT-DISPLAY EA TFT009-81AINN
- PCB SHIELD WITH CAPACITIVE TOUCH EA TFT009-81AITC

EA RaPicoTFT009
EA RaPicoTFT009TC

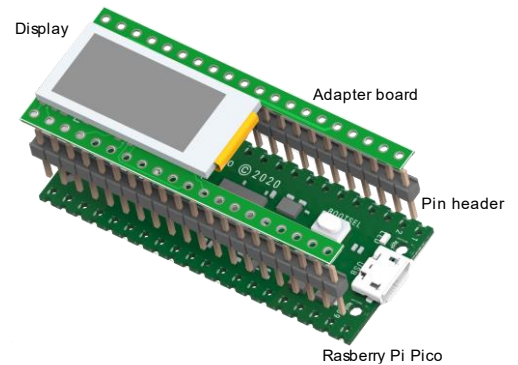
* 2x Pin-header included; Raspberry Pi Pico not included

GENERAL

The EA RaPicoTFT009 is a 0.9" TFT display with adapter board suitable for the Raspberry Pi Pico (not included).

The Raspberry Pi Pico has an Arm Cortex M0+ processor, with a clock speed up to 133 MHz, 264 kB SRAM and 2 MB Flash.

One of the two SPI interfaces is used for communication with the display, as well as one of the 26 GPIO pins (GP13). The version with touch panel (EA RaPicoTFT009TC) additionally needs an I²C connection and another GPIO pin as interrupt input (GP8).



SOFTWARE

A software example in C can be found at <https://www.lcd-module.com/index.php?id=809>. The following functions are available:

TFTspi_init	Initializes the SPI interface with the selected clock frequency.	
Transfer parameter	uint clk	Frequency of the SPI clock
Example	TFTspi_init(16000000)	

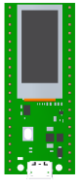
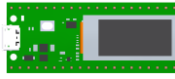
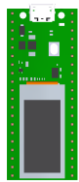
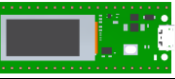
TFTinit	Initializes the selected TFT module.	
Transfer parameter	int16_t tfttyp	TFT Module 1: EA TFT009-81AI 2: EA TFT015-22AI 3: EA TFT020-23AI 4: EA TFT028-23AI 5: EA TFT035-34AI
Example	TFTinit(EA_TFT009)	


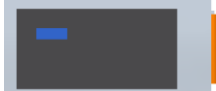
TFTreset	Performs a reset of the display and, if necessary, the touch and then initializes the selected TFT module.	
Transfer parameter	int16_t tfttyp	TFT Module 1: EA TFT009-81AI 2: EA TFT015-22AI 3: EA TFT020-23AI 4: EA TFT028-23AI 5: EA TFT035-34AI
Example	TFTreset(EA_TFT009)	


TOUCHi2c_init	Initializes the I2C interface with the selected clock frequency.	
Transfer parameter	uint clk	Frequency of the I2C clock
Example	TOUCHi2c_init(400000)	


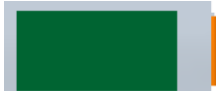
setTFTfont	Sets the font. The setting remains until it is changed again. There are 4 fonts in total. Each font can be zoomed in x- and y-direction in four steps.	
Transfer parameter	int16_t nr	Font number 1: 6x8 Pixel 2: 7x12 Pixel 3: 8x16 Pixel 4: 30x38 Pixel
	int16_t zx	Zoom factor in x direction (1-4)
	int16_t zy	Zoom factor in y direction (1-4)
Example	setTFTfont(FONT_8X16,2,2)	


setTFTcolor	Sets the foreground and background color for the font.	
Transfer parameter	uint16_t fg	Foreground color (16-Bit RGB565)
	uint16_t bg	Background color (16-Bit RGB565)
Example	setTFTcolor(RGB565(50,100,200),BLACK)	

setTFTangle	Changes the orientation of the display. The origin (0,0) is always top left.		
Transfer parameter	int16_t angle	0	
		90	
		180	
		270	
Example	setTFTangle(90)		

TFTfill	Fills a rectangular area with a color.	
Transfer parameter	uint16_t color	Color (16-Bit RGB565)
	int16_t xs	Start point x-coordinate
	int16_t ys	Start point y- coordinate
	int16_t xe	End point x- coordinate
	int16_t ye	End point y- coordinate
Example	TFTfill(RED,20,20,50,30)	
	TFTfill(RGB565(50,100,200), 20,20,50,30)	

TFTtext	Places a string. The font must have been set before with the setTFTfont function. The color of the font and its background is set by the setTFTcolor function.	
Transfer parameter	int16_t xs	Textposition x-coordinate
	int16_t ys	Textposition y- coordinate
	char *ptext	Text
Example	setTFTcolor(BLUE,YELLOW) setTFTfont(FONT_6X8,1,1) TFTtext(20,20,“Hello World“)	

TFTclear	Deletes the entire contents of the display with the selected color.	
Transfer parameter	uint16_t color	color (16-Bit RGB565)
Example	TFTclear(YELLOW)	
	TFTclear(RGB565(0,100,50))	

TFTbitmap	Places an uncompressed Windows Bitmap (*.bmp).	
Transfer parameter	int16_t xs	Bitmap position x-coordinate
	int16_t ys	Bitmap position y- coordinate
	const unsigned char *pbitmap	Bitmap data
Example	TFTbitmap(0,0,ImageData)	

waitUntilTouch	Waits until a touch event occurs	
Transfer parameter	int16_t event	Touch event 1: Touch up 2: Touch down 4: Touch drag
Example	waitUntilTouch(TOUCH_DOWN)	

Global variables for touch

gTouchOK	Indicates whether a new touch point is available. Change the value to 0 when the new touch point has been read. 0: No new touch event 1: New touch event
gTouchEvent	Indicates the current type of the touch event. 1: Touch up 2: Touch down 4: Touch drag
gX	Current touch x-coordinate
gY	Current touch y-coordinate

Following defines can be used:

TFT Modules	
EA_TFT009	1
EA_TFT015	2
EA_TFT020	3
EA_TFT028	4
EA_TFT035	5
Fonts	
FONT_6X8	1
FONT_7X12	2
FONT_8X16	3
FONT_30X38	4
Color	
RGB565(r,g,b)	(((r)&0xF8) << 8) (((g)&0xFC) << 3) ((b)>>3))
BLACK	0x0000
WHITE	0xFFFF
RED	RGB565(255,0,0)
GREEN	RGB565(0,255,0)
BLUE	RGB565(0,0,255)
CYAN	RGB565(0,255,255)
MAGENTA	RGB565(255,0,255)
YELLOW	RGB565(255,255,0)
Touch	
TOUCH_UP	1
TOUCH_DOWN	2
TOUCH_DRAG	4

Following fonts are included:

6x8 (FONT_6X8)

+ Lower Upper	\$0 (0)	\$1 (1)	\$2 (2)	\$3 (3)	\$4 (4)	\$5 (5)	\$6 (6)	\$7 (7)	\$8 (8)	\$9 (9)	\$A (10)	\$B (11)	\$C (12)	\$D (13)	\$E (14)	\$F (15)
\$20 (dez: 32)	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
\$30 (dez: 48)	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
\$40 (dez: 64)	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
\$50 (dez: 80)	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
\$60 (dez: 96)	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
\$70 (dez: 112)	p	q	r	s	t	u	v	w	x	y	z	{		}	~	Δ
\$80 (dez: 128)	Ç	ü	é	â	ä	à	ç	ê	ë	è	ï	î	ï	ñ	ñ	
\$90 (dez: 144)	É	æ	Æ	ô	ö	ò	û	ü	ö	ü	€	£	¥	β	f	
\$A0 (dez: 160)	á	í	ó	ú	ñ	ñ	á	ó	í	ú	½	¼	i	«	»	
\$B0 (dez: 176)	⋮	⋮	⋮													
\$C0 (dez: 192)	L	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥
\$D0 (dez: 208)	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥
\$E0 (dez: 224)	α	β	Γ	π	Σ	σ	μ	τ	ϖ	θ	ϖ	δ	φ	φ	€	Π
\$F0 (dez: 240)	≡	±	≥	≤	∫	J	÷	≈	°	•	.	√	n	2	3	-

7x12 (FONT_7X12)

+ Lower Upper	\$0 (0)	\$1 (1)	\$2 (2)	\$3 (3)	\$4 (4)	\$5 (5)	\$6 (6)	\$7 (7)	\$8 (8)	\$9 (9)	\$A (10)	\$B (11)	\$C (12)	\$D (13)	\$E (14)	\$F (15)
\$20 (dez: 32)	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
\$30 (dez: 48)	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
\$40 (dez: 64)	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
\$50 (dez: 80)	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
\$60 (dez: 96)	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
\$70 (dez: 112)	p	q	r	s	t	u	v	w	x	y	z	{		}	~	Δ
\$80 (dez: 128)	Ç	ü	é	â	ä	à	ç	ê	ë	è	ï	î	ï	ñ	ñ	
\$90 (dez: 144)	É	æ	Æ	ô	ö	ò	û	ü	ö	ü	€	£	¥	β	f	
\$A0 (dez: 160)	á	í	ó	ú	ñ	ñ	á	ó	í	ú	½	¼	i	«	»	
\$B0 (dez: 176)	⋮	⋮	⋮													
\$C0 (dez: 192)	L	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥
\$D0 (dez: 208)	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥
\$E0 (dez: 224)	α	β	Γ	π	Σ	σ	μ	τ	ϖ	θ	ϖ	δ	φ	φ	€	Π
\$F0 (dez: 240)	≡	±	≥	≤	∫	J	÷	≈	°	•	.	√	n	2	3	-

8x16 (FONT_8X16)

+ Lower Upper	\$0 (0)	\$1 (1)	\$2 (2)	\$3 (3)	\$4 (4)	\$5 (5)	\$6 (6)	\$7 (7)	\$8 (8)	\$9 (9)	\$A (10)	\$B (11)	\$C (12)	\$D (13)	\$E (14)	\$F (15)
\$20 (dez: 32)	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
\$30 (dez: 48)	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
\$40 (dez: 64)	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
\$50 (dez: 80)	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
\$60 (dez: 96)	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
\$70 (dez: 112)	p	q	r	s	t	u	v	w	x	y	z	{		}	~	Δ
\$80 (dez: 128)	Ç	ü	é	â	ä	à	ç	ê	ë	è	ï	î	ï	ñ	ñ	
\$90 (dez: 144)	É	æ	Æ	ô	ö	ò	û	ü	ö	ü	€	£	¥	β	f	
\$A0 (dez: 160)	á	í	ó	ú	ñ	ñ	á	ó	í	ú	½	¼	i	«	»	
\$B0 (dez: 176)	⋮	⋮	⋮													
\$C0 (dez: 192)	L	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥
\$D0 (dez: 208)	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥
\$E0 (dez: 224)	α	β	Γ	π	Σ	σ	μ	τ	ϖ	θ	ϖ	δ	φ	φ	€	Π
\$F0 (dez: 240)	≡	±	≥	≤	∫	J	÷	≈	°	•	.	√	n	2	3	-

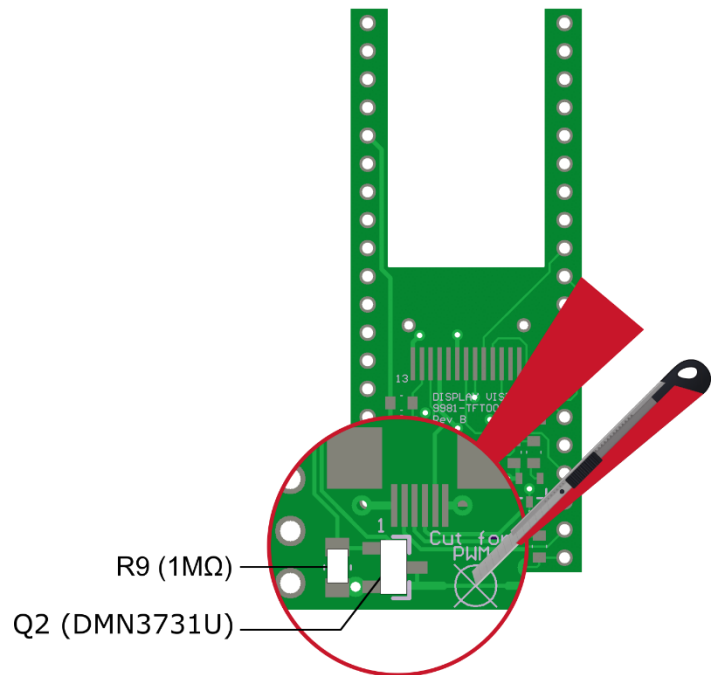
30x38 (FONT_30X38)

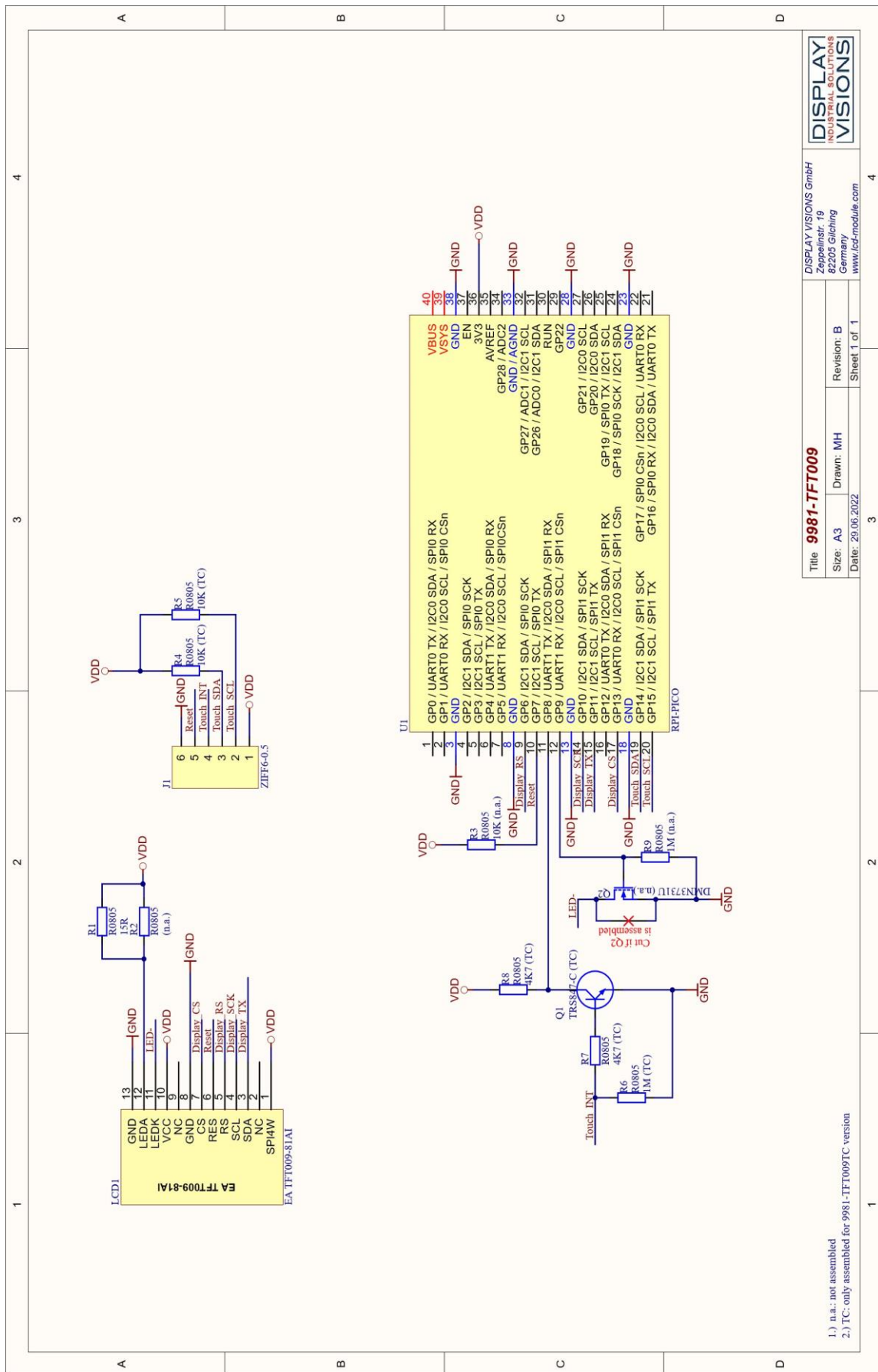
\$30 (48)	\$31 (49)	\$32 (50)	\$33 (51)	\$34 (52)	\$35 (53)	\$36 (54)	\$37 (55)	\$38 (56)	\$39 (57)	\$3A (58)
0	1	2	3	4	5	6	7	8	9	:

HARDWARE

The EA RaPicoTFT009 is supplied with the display soldered in. The components marked with “n.a.” (not assembled) in the following circuit diagram are not assembled. The components marked with “TC” are only assembled in the version EA RaPicoTFT009TC with capacitive touch panel. Suitable pin headers for the Raspberry Pi are enclosed.

By default, the backlight is always on. To make this dimmable or switchable, the components R9 (1M Ω) and Q2 (DMN3731U) must be fitted manually. It is mandatory to cut the trace marked with “*Cut for PWM*”. With this option the power-down / energy-saving modes of the Pi Pico can be used. GP9 then acts as a switch, high = backlight on, low = off. The backlight can be dimmed via PWM.





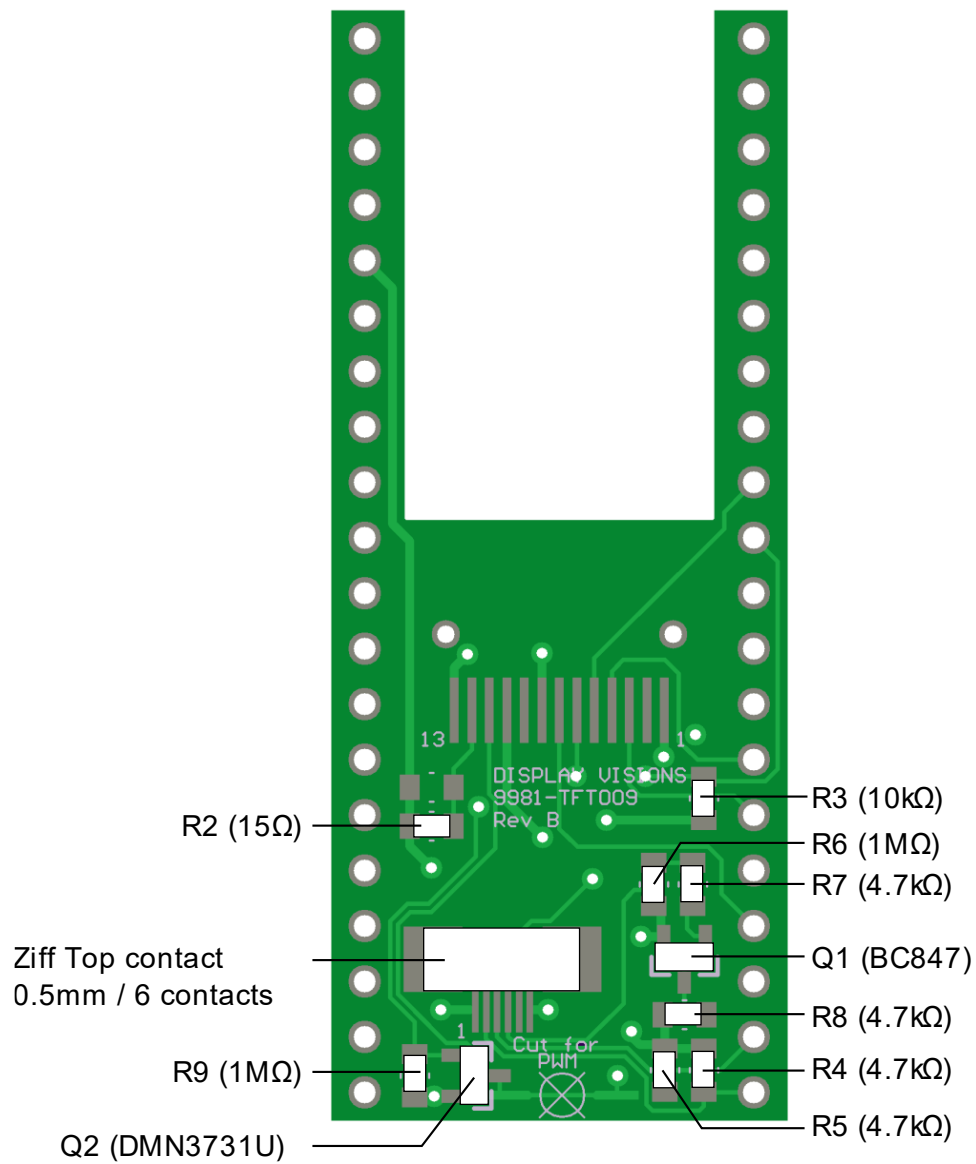
DISPLAY
INDUSTRIAL SOLUTIONS
VISIONS

Title: **9981-TFT009**
 Size: A3
 Date: 20.06.2022
 Drawn: MH
 Revision: B
 Sheet 1 of 1

DISPLAY VISIONS GmbH
 Zappelstr. 19
 82205 Garching
 Germany
 www.lcd-module.com

1.) n.a.: not assembled
 2.) TC: only assembled for 9981-TFT009TC version

PLACEMENT SCHEDULE



DIMENSION

