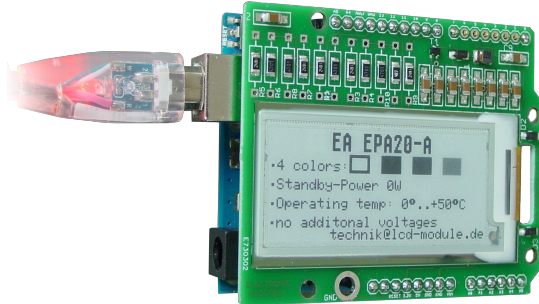


Arduino meets EA EPA20-A



This little project describes the connection between an Arduino-Board and a ELECTRONIC ASSEMBLY ePaper Display, like the EA EPA20-A. For more details, you will find the display's datasheet under <http://www.lcd-module.com/fileadmin/eng/pdf/grafik/epa20-ae.pdf> and the controller's under http://www.lcd-module.de/fileadmin/eng/pdf/zubehoer/ssd1606_1_1.pdf.

Table of content

1. Hardware.....	2
1.1. Display.....	2
1.2. Schematic.....	3
1.3. PCB.....	4
1.4. Bill of Materials EA EPA20-A.....	4
2. Software library classname: epa20.....	5
2.1. initialize.....	1
2.2. clear.....	1
2.3. update.....	1
2.4. string.....	1
2.5. rectangle.....	1
2.6. picture.....	2
3. The Library in action.....	3

1. Hardware

1.1. Display

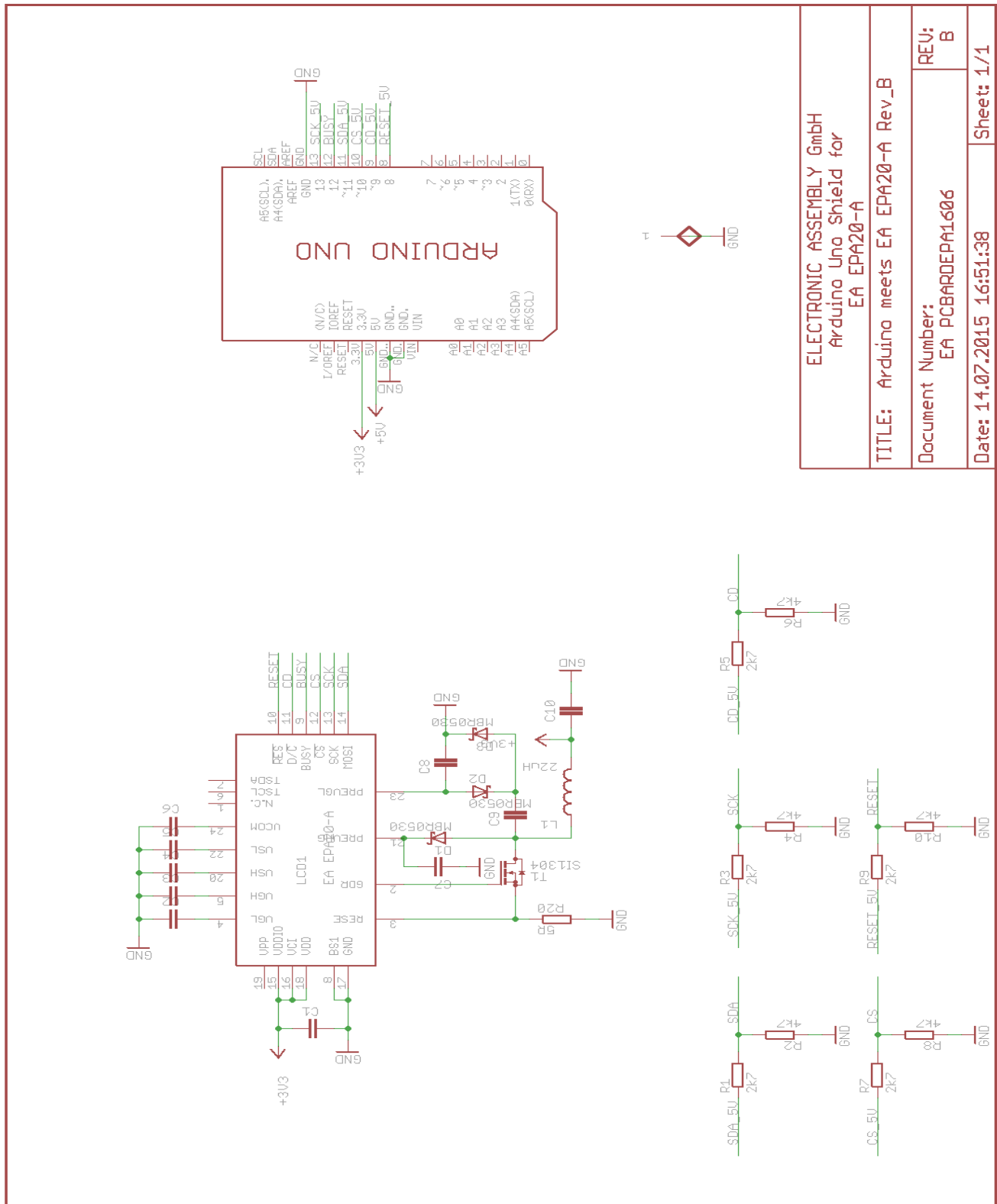
The EA EPA20-A is a standard ePaper display. This technology implies many advantages, such as wide viewing angle high contrast, i.e. paper white and ultra low power consumption.

Because of the technology there is no backlight available.

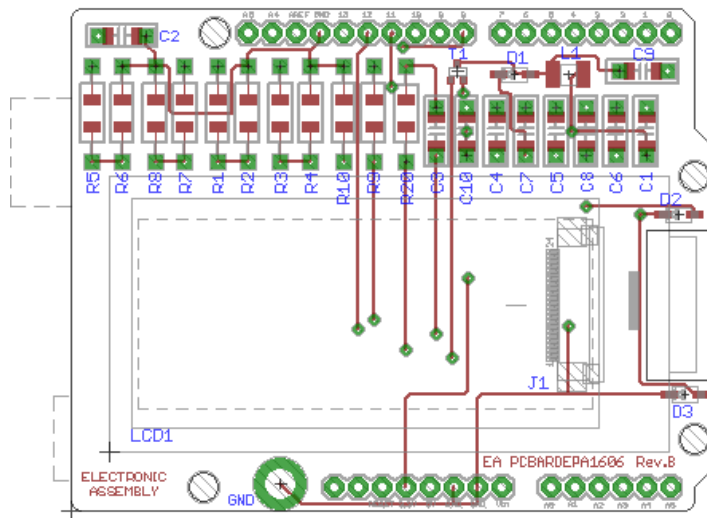
Arduino meets EA EPA20-A

with SSD1606 controller

1.2. Schematic



1.3. PCB



1.4. Bill of Materials EA EPA20-A

Designator	Value	Mouser p/n	Reichelt p/n
C1, C2, C3, C4, C5, C6, C7, C8, C9, C10	1 μ F	80-ARR07D105KGS	X7R-G1206 1,0/50
L1	22 μ H	710-744032220	LQH3C 22 μ
R1, R3, R5, R7, R9	2.7 K Ω	273-2.7K-RC	SMD 1/4W 2,7K
R2, R4, R6, R8, R10	4.7 K Ω	273-4.7K-RC	SMD 1/4W 4,7K
R20	5 Ω	71-CPF15R0000BEE14	SMD 1/4W 5,1
D1, D2, D3	MBR0530	511-STPS0530Z	BAT 42W
T1	SI1304	78-SI1308EDL-T1-GE3	
J1	EA WF050-24T	710-68712414022	
LCD1	EA EPA20-A		EA EPA20-A

2. Software library classname: epa20

The library provides all necessary functions for using a EA EPA20-A graphic display with SSD1606 controller.

Importing the library is very easy: start Arduino application, Sketch → Import Library → Add Library

Please add “Arduino meets EA EPA20-A.zip” Your sketch should look like this:

```
#include <Arduino.h>
#include <SPI.h>
#include <epa20.h>
#include <logo_BLV.h>
#include <font_6x8.h>
#include <font_8x8.h>
#include <font_8x16.h>
#include <font_16x32nums.h>

epa20 EPA20;

void setup()
{

}

void loop()
{

}
```

In addition you will find examples for EA EPA20-A with SSD1606 controller in the library folder “examples”.

These functions and definitions are provided within the library. Thanks to the grayscales it's only possible to access the display y-coordinates in 4 bit boundaries, that means there are 18 ‘pages’ in y direction. The coordinate 0|0 is in the lower left corner.

Classname: **epa20**

```
#define XPIXEL 172
#define YPIXEL 72
```

```
#define WHITE 0xFF
#define BLACK 0x00
#define GRAY1 0x55
#define GRAY2 0xAA
```

```
void initialize (byte p_cs, byte p_si, byte p_so, byte p_clk, byte p_a0, byte p_res);
```

```
void update (void);
```

```
void clear (void);
```

```
void rectangle (byte xs, byte ys, byte xe, byte ye, byte outline, byte filling);
```

Printing and typographical errors reserved. ELECTRONIC ASSEMBLY reserves the right to change specifications without prior notice.

ELECTRONIC ASSEMBLY * Zeppelinstr.19 * D-82205 Gilching * Phone: +49-8105-778090 * www.lcd-module.de * info@lcd-module.de

```
void string      (byte column, byte page, const byte *font_adress, const char *str, byte fg_color,
                 byte bg_color);
void picture     (byte x, byte y, const byte *pic_adress, byte fg_color, byte bg_color);
void rectangle   (byte start_column, byte start_page, byte end_column, byte end_page, byte
                 pattern);
void picture     (byte column, byte page, const byte *pic_adress);
```

2.1. initialize

Name: `void initialize (byte p_cs, byte p_si, byte p_so, byte p_clk, byte p_a0, byte p_res)`

Vars: CS-Pin; MOSI-Pin; MISO-Pin, SCK-PIN; A0-Pin (data or command), Reset-Pin

Description: The function inits the SPI. You can select whether the Arduino uses a software SPI (p_si, p_so and p_clk are different and name the port) or you can select hardware SPI by setting p_si=p_clk=0. Second the function inits the EA EPA20-A

Example: Initialize EA EPA20-A using Hardware SPI:

```
EPA20.initialize(10,0,0,0,9,8);
```

SS = 10, 0,0,0= use Hardware SPI, 9 = A0, 8=Reset

Initialize EA EPA20-A using Software SPI

```
EPA20.initialize(10,11,12,13,9,8);
```

SS = 10, MOSI = 11, MISO = 12, SCK = 13 use Software SPI, 9 = A0, 8=Reset

2.2. clear

Name: `void clear (void);`

Vars: ---

Description: The function clears the entire display

Example:

```
EPA20.clear();
```

2.3. update

Name: `void update (void);`

Vars: ---

Description: The function starts to refresh display content. → Show new RAM content on the screen

Example: `EPA20.update();`

2.4. string

Name: `void string (byte x, byte y, const byte *font_adress, const char *str, byte fg_color, byte bg_color)`

Vars: x (0..172), y (0..72), ptr to font data, ptr to 0 terminated string, color foreground and background

Description: The function writes a string with selected font to a specific position. Fonts that are included in the library are: **font_6x8**, **font_8x8**, **font_8x16**, **font_16x32nums**. The font needs to be a 'FV' font. You can generate it with ELECTRONIC ASSEMBLY's FontEditor (EA USBSTICKFONT).

Example: "Hello World" at low left corner with big characters (8x16), with normal colors
`EPA20.string(0, 0, font_8x16, "Hello World", BLACK, WHITE);`

2.5. rectangle

Name: `void rectangle (byte xs, byte ys, byte xe, byte ye, byte outline, byte filling);`

Vars: low left corner (xs,ys) to upper right corner (xe,ye), outline color (2 dots width) and filling color

Description: Draws a filled rectangle on the screen.

Example: Fill whole display

```
EPA20.rectangle(0,0,171,71, BLACK, GRAY1);
```

2.6. picture

Name: void picture (byte x, byte y, const byte *pic_adress, byte fg_color, byte bg_color);

Vars: lower left corner, ptr to picture data, color for set pixels (fg_color) and background

Description: The function shows a picture. The picture needs to be 'BLV' formatted. You can generate it with ELECTRONIC ASSEMBLY's BitMapEdit, which is available in the EA LCD-Tools. You can download them from <http://www.lcd-module.com/support.html>. We have included one demo picture: ea_logo.

Example: show EA logo (118x32 dots) in the middle

```
EPA20.picture(27,20, ea_logo);
```


3. The Library in action

Please feel free to explore some pictures of our module working with the Arduino library the shield "EA PCBARDEPA1606" together with our EA EPA-20A display.

