



# Bit/Pixel Mapping

## Applikationshinweis

# INHALTSVERZEICHNIS

<b>Bildkonvertierung für LCD-Tastenschalter</b> .....	<b>3</b>
<b>Bild Mapping</b> .....	<b>4</b>
Skalierung auf Seitenverhältnis.....	4
Anpassung an die neue Auflösung.....	5
Konvertierung in Schwarz-Weiß.....	5
Spiegelung und Drehung.....	5
Aufteilung.....	5
<b>Java-Implementierungsbeispiel</b> .....	<b>8</b>
<b>Hinweise</b> .....	<b>11</b>
Copyright Hinweis.....	11
Technische Hinweise.....	11
Gewährleistungsausschluss.....	11
<b>Hinweis zur deutschen Version</b> .....	<b>12</b>
<b>Änderungshistorie</b> .....	<b>13</b>

# BILDKONVERTIERUNG FÜR LCD-TASTENSCHALTER

Dieses Dokument zeigt die Schritte, die erforderlich sind, um ein Bild an einen LCD-Tastenschalter von [E<sup>3</sup>] zu senden. Der in diesem Dokument beschriebene Prozess ermöglicht es dem Benutzer, ein beliebiges Bild in das geeignete Datenformat für die Anzeige auf der [E<sup>3</sup>]-LCD-Taste zu konvertieren. Außerdem wird gezeigt, wie diese Schritte mit Hilfe einer Java-Anwendung automatisiert werden können.

Zur Veranschaulichung des Beispiels wird ein Bild von Leonardo DaVincis Mona Lisa verwendet, das auf einer SA6432 Taste angezeigt werden soll.



Die Umwandlung eines Farbbildes in das [E<sup>3</sup>]-Datenformat umfasst mehrere Schritte:



**Skalierung des Bildes auf das richtige Seitenverhältnis**



**Anpassung der Auflösung an den Schaltertyp**



**Konvertierung in Schwarz-Weiß**



**Spiegelung des Bildes**



**Drehen**



**Splitting (falls erforderlich)**



# BILD MAPPING

## Skalierung auf Seitenverhältnis



Dieses Bild der Mona Lisa hat eine Auflösung von 396 x 499 Pixeln, was einem Seitenverhältnis von 4 zu 5 entspricht. Die SA6432 hat aufgrund ihrer Auflösung von 64 x 32 Pixeln ein Seitenverhältnis von 2:1.

Um das Bild an das richtige Seitenverhältnis anzupassen, haben wir das Bild auf 396 x 198 Pixel beschnitten.



# Anpassung an die neue Auflösung

Dieses Bild hat das richtige Seitenverhältnis und muss in eine Auflösung von 64 x 32 Pixeln umgewandelt werden.



## Konvertierung in Schwarz-Weiß

Die LCD-Anzeige des SA6432-Schalters ist eine Schwarz-Weiß-Anzeige. Daher müssen wir das Bild in eine Schwarz-Weiß-Bitmap konvertieren. Dies kann in einem Schritt oder mit einem Graustufen-Zwischenbild geschehen, wie hier gezeigt.



## Spiegelung und Drehung

Um dem Bitmap-Format der LCD-Tastenschalter zu entsprechen, muss das Bild nun gespiegelt und um 90° gegen den Uhrzeigersinn gedreht werden.



## Aufteilung

Das SA6432-Bitmap-Format erfordert auch, dass die obere und untere Hälfte des Bildes vertauscht werden.



Das Bild kann nun mit dem Befehl 0x40 an die SA6432-Taste übertragen werden. Dazu werden vier Pixel, beginnend an der linken oberen Position, zu einem Nibble zusammengefasst und als die niederwertigsten vier Bits übertragen.

Das folgende .e3t-Skript zeigt das entsprechende Datenformat für das Bild:

```

Transmit Bitmap: C:\MCP\Bitmap\MonaLisa.bmp
0x40                * Set Display address and write display data
;
* Data Bytes to follow 512 each containing just one nibble
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x03 ;* -      ## #####
0x0C 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x00 ;* -      #####
0x08 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x00 ;* -      #####
0x08 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x00 ;* -      #####
0x0C 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x00 ;* -      #####
0x08 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x00 ;* -      #####
0x08 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x00 ;* -      #####
0x08 0x0F 0x0F 0x07 0x00 0x00 0x00 0x00 ;* -      #####
0x00 0x0F 0x0F 0x07 0x00 0x00 0x00 0x00 ;* -      #####
0x00 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x00 ;* -      #####
0x0A 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x00 ;* -      # #####
0x0B 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x08 ;* -      ### #####
0x0B 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x0F ;* -      #####
0x0B 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x0C ;* -      #####
0x0B 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x0C ;* -      #####
0x0B 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x0E ;* -      #####
0x0B 0x0F 0x0F 0x0F 0x00 0x00 0x08 0x0E ;* -      # #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x08 0x0F ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x08 0x0F ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x0C 0x0F ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x08 0x0F 0x0F ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x0F 0x0F 0x0F ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x0F 0x0F 0x0F ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x08 0x0F 0x0F 0x0F ;* -      #####
0x0F 0x0F 0x0F 0x07 0x0C 0x0F 0x0F 0x0F ;* -      #####
0x0F 0x0F 0x0F 0x07 0x0E 0x0F 0x0F 0x0F ;* -      #####
0x0C 0x0F 0x0F 0x07 0x0E 0x0F 0x07 0x07 ;* -      ##### ##
0x08 0x0F 0x0F 0x07 0x0E 0x0F 0x06 ;* -      ##### ##
0x00 0x08 0x0F 0x07 0x0E 0x0F 0x00 0x06 ;* -      ##### ##
0x00 0x00 0x0F 0x07 0x0E 0x03 0x00 0x06 ;* -      ##### ##
0x00 0x00 0x0E 0x03 0x0E 0x01 0x00 0x02 ;* -      ##### #
0x00 0x04 0x0C 0x03 0x0E 0x01 0x00 0x02 ;* -      ##### #
0x08 0x0F 0x0C 0x01 0x0E 0x01 0x00 0x0E ;* -      ##### ##
0x00 0x0F 0x0C 0x0F 0x0E 0x01 0x00 0x00 ;* -      ##### ##
0x00 0x00 0x0C 0x0F 0x0C 0x01 0x00 0x06 ;* -      ##### ##
0x00 0x00 0x0F 0x0F 0x08 0x03 0x00 0x06 ;* -      ##### ##
0x00 0x08 0x0F 0x0F 0x00 0x03 0x00 0x06 ;* -      ##### ##
0x08 0x0F 0x0F 0x0F 0x00 0x0E 0x03 0x06 ;* -      ##### ##
0x0F 0x0F 0x0F 0x0F 0x00 0x08 0x0F 0x0F ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x0E 0x0F ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x08 0x0F ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x00 ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x08 ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x00 ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x00 ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x00 ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x00 ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x08 ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x0C ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x0E ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x0C ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x0C ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x00 ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x00 ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x0C ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x0E ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x00 0x0F ;* -      #####
0x0F 0x0F 0x0F 0x0F 0x00 0x00 0x08 0x0F ;* -      #####
* - End of Bitmap

```

**Hinweis:** Das vom MCP-Programm generierte .e3t-Skript zeigt die tatsächliche Bild-Bitmap, ohne die obere und untere Hälfte umzukehren.

Nach der Übertragung an die SA6432-Taste sieht das Bild wie folgt aus:





# JAVA-IMPLEMENTIERUNGSBEISPIEL

Dieses Kapitel enthält einen Auszug aus dem Java-Programm [E<sup>3</sup>] PanelControl, das ein Bild nimmt und den Bildbefehl (0x40) für die Übertragung an eine SA6432-Taste in einer .e3t-Skriptdatei erstellt.

```
/**
 * This method write the command for display of an image in a .e3t Script file.
 *
 * @param   pixelArray Image in form of a one dimensional integer array
           Each pixel, starting with the top left pixel, is inserted
           into the array resulting in an array of height x width
           resolution.
 */
private void writeImage(int pixelArray[]){

    //Pixel per column of the LCD display - in this case for resolution 64x32
    int height = 32;
    //Pixel per row of the LCD display - in this case for resolution 64x32
    int width  = 64;

    //BitSet[] Array containing a bit list of each row of the rotated image
    BitSet[] pixelMap = new BitSet[width];

    int index = 0;

    //Create PixelMap (The images is rotated 90° counter-clockwise and mirrored).
    for(int i = width - 1 ; i >= 0 ; i--){

        pixelMap[width - i - 1] = new BitSet();

        for(int j = 0 ; j < height ; j++){

            if(pixelArray[i + width * j] == 1){
                pixelMap[width - i - 1].set(j, true);
            } else {
                pixelMap[width - i - 1].set(j, false);
            }

            index++;
        }
    }
}
```



```

//Output of the command for selecting the first pixel of the display in the
//.e3t Script file. \the targetWriter is an output channel for that file.
targetWriter.println("0x40 0x00 0x00 0x00");

//this string is a buffer for assembling a nibble first in binary, then in
//Hex format.
String tmpBinary = "";
//this string array contains as many strings as the rotated image has columns.
//In the case of a 64x32 resolution there are 8 column strings.
String[] byteArray = new String[height / 4];

//This loop assembles one nibble per column for each row of the rotated image.
//Once all nibbles of a row are assembled they handed off to another method,
//which handles the specific requirements for each resolution.
for(int i = 0 ; i < pixelMap.length ; i++){

    index = 0;
    for(int j = 0 ; j < height ; j++){
        //Assembling one nibble. 1 means pixel ON; 0 means pixel OFF.
        if(pixelMap[i].get(j)){
            tmpBinary = tmpBinary + "1";
        } else {
            tmpBinary = tmpBinary + "0";
        }

        //Once four pixels are read the nibble representation is changed
        //from binary to HEX format. For example, a „0110“ would generate
        //the string „0x06“.
        if(tmpBinary.length() > 3){
            byteArray[index] = "0x0" +
Integer.toHexString(Integer.parseInt(tmpBinary, 2));
            index++;
            tmpBinary = "";
        }
    }

    //After reading the complete row the method for writing the read row
    //is executed.
        writeBytesInColumn(byteArray);
}

//Finally, the END command to terminate the transmission is written into
//the .e3t script file.
targetWriter.println("0x43");
}

```

```

/**
 * This method writes the pixels of a column of an image into the .e3t script file.
 *
 * @param bytesInColumn This string array contains the column bytes in HEX format.
 */
private void writeBytesInColumn(String[] bytesInColumn){
    switch(board.getKeyType()){
        case LCDKey.SA6432:
            //First, the last four bytes of the row are sent
            //(since the lower part of the image must be sent
            //first).
            for(int i = 4 ; i < 8 ; i++){
                targetWriter.print(bytesInColumn[i] + " ");
            }
            //Next follow the first four bytes.
            for(int i = 0 ; i < 4 ; i++){
                targetWriter.print(bytesInColumn[i] + " ");
            }
            break;
    }
}
}

```

# HINWEISE

## Copyright Hinweis

© 2009-2024 Copyright [E<sup>3</sup>] Engstler Elektronik Entwicklung GmbH. Alle Rechte vorbehalten.

[E<sup>3</sup>], The Third Evolution™ und Legacy Mode™ sind Warenzeichen von [E<sup>3</sup>]. The Keys to Intelligence™ ist eine Marke von I/O Universal Technologies, Inc. und wird mit Genehmigung verwendet. Alle anderen Marken sind Eigentum der jeweiligen Inhaber.

Kein Teil dieser Veröffentlichung darf ohne die ausdrückliche schriftliche Genehmigung der [E<sup>3</sup>] Engstler Elektronik Entwicklung GmbH kopiert, fotokopiert, reproduziert, übersetzt oder auf ein elektronisches Medium oder eine maschinenlesbare Form reduziert werden.

## Technische Hinweise

Dieses Datenblatt wendet sich an technisch qualifiziertes und auf dem Gebiet der Elektronik ausgebildetes Personal.

Die Kenntnis der Elektronik und die technisch korrekte Umsetzung des Inhalts dieses Datenblatts sind Voraussetzung für die problemlose Installation, Inbetriebnahme und den sicheren Betrieb des beschriebenen Produkts. Nur qualifiziertes Personal verfügt über das erforderliche Know-how, um die in diesem Datenblatt gemachten Angaben umzusetzen.

Aus Gründen der Übersichtlichkeit wurden nicht alle Details über das Produkt oder dessen Implementierung, Installation, Betrieb oder Wartung aufgenommen. Sollten Sie zusätzliche Informationen oder weitere Unterstützung benötigen, wenden Sie sich bitte an Ihren örtlichen [E<sup>3</sup>] Distributor oder an [E<sup>3</sup>] Engstler Elektronik Entwicklung GmbH unter [techsupport@e3-keys.com](mailto:techsupport@e3-keys.com). Sie können auch unsere Website unter [www.e3-keys.com](http://www.e3-keys.com) besuchen.

## Gewährleistungsausschluss

Die [E<sup>3</sup>] ENGSTLER ELEKTRONIK ENTWICKLUNG GMBH übernimmt in Bezug auf dieses Datenblatt keine Garantie, weder ausdrücklich noch stillschweigend, und sie haftet nicht für direkte oder indirekte Schäden. In einigen Staaten ist der Ausschluss von beiläufigen Schäden oder Folgeschäden nicht zulässig, so dass diese Erklärung in solchen Fällen möglicherweise nicht gültig ist.

Dieses Datenblatt wurde mit der gebotenen Sorgfalt erstellt. Da jedoch Fehler nicht ausgeschlossen werden können, übernimmt die [E<sup>3</sup>] Engstler Elektronik Entwicklung GmbH keine Gewährleistung und keine rechtliche Verantwortung oder Haftung in irgendeiner Form für fehlerhafte Angaben in diesem Datenblatt.

# HINWEIS ZUR DEUTSCHEN VERSION

Dieses Dokument wurde mit Hilfe von [www.DeepL.com/Translator](http://www.DeepL.com/Translator) übersetzt. Für den Fall, daß es zwischen dem Originaldokument in Englisch und der deutschen Übersetzung Unterschiede gibt, gilt immer das englische Original als bindend.

# ÄNDERUNGSHISTORIE

Version	Datum	Bemerkungen
0.1	07.10.09	Erstes Entwurfsdokument (Deutsch)
0.2	12.11.09	Erstes Entwurfsdokument (Englisch)
0.3	30.06.20	Neue Formattierung
0.4	30.03.22	Aktualisierter und bearbeiteter Inhalt
1.0	15.06.22	aktualisiertes Freigabedokument
1.1	24.10.24	Neue Geschäftsadresse

**[E<sup>3</sup>] Engstler Elektronik Entwicklung GmbH**  
Auweg 27 • 63920 Grossheubach • Germany

[WWW.E3-KEYS.COM](http://WWW.E3-KEYS.COM)