



Image Convertor V2.0

Overview

Application **Image Converter** allows very easy and fast conversion your image to code array for LCD graphic display (Graphic Module) connect to microprocessor. Image Converter support three compress conversion type (None, Zero and Full) and support four target language type (ASM, Pascal, Basic and C). It runs under Windows 9x / Me, Windows NT / 2000 / XP.

Main Features

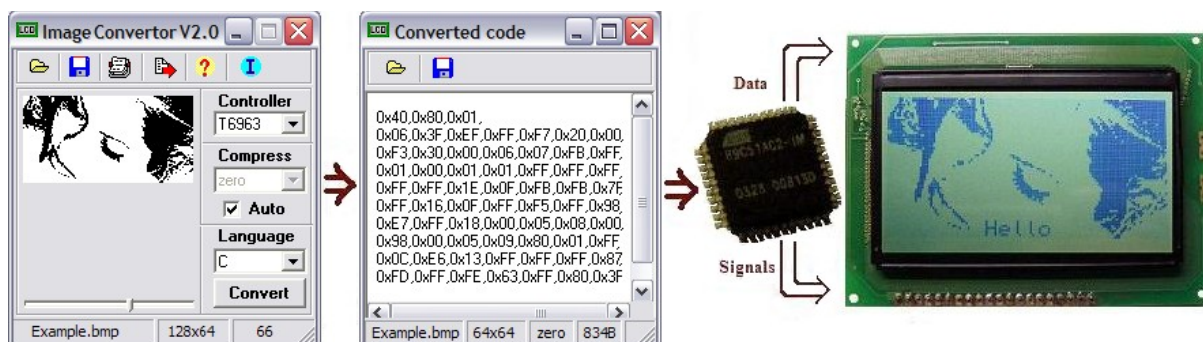
1. Fast and Easy image conversion (*.BMP) in format 1x1 až 320x240 pixels
2. Three types of controllers (KS107, ST7920, T6963)
3. Three images compress conversion type (None, Zero and Full)
4. Four target language type (ASM, Pascal, Basic and C)
5. Select boundary black/white for colour image
6. Supports library in ASM51 and C language for work with Image and Text (32 to 128 ASCII)

Available for LCD module with controllers KS107, ST7920, T6963

EL 6464, PG 12856, PG 12864, GDM 12864, EL 12864, EL 24064, EL 16032, L 16032, PG 128128, PG 24064, PG 240128

Using Application

1. Open Image *.BMP file
2. Select boundary black/white for colour image
3. Select Controllers, Compress and Language Type
4. Make Convert (press button convert)
5. Copy created code array to your include file in implementation language
6. Call code array with special library routine DISP_ICON



Obr. 1 – Conversion scheme

Creating compress code

For example we can use small image which created black and white square. Each square have 8x8 pixels size. This image we can convert to three compress type (None, Zero, Full). In all compress type in created code array means first three bytes height (in pixels), width (in pixels) and compress type (None = 0x00, Zero = 0x01 and Full = 0x02).

None (0x00)

All data are storage in converted code array without any arrange. Your routine must only get this data and write to LCD module. Conversion is advantageous for not regulary image.

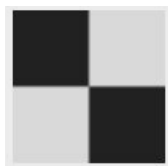
Zero (0x01)

Data are storage in simply compress. When is occurence in array code number zero then next number past this zero it means count. Conversion is advantageous for image which predominance white place.

Full (0x02)

Every first number are data and second number past first it means count. Conversion is advantageous for regulary image.

Example converted image



Picture attribute
16 x 16 pixels

Converted in *none* compress type

```
0x10,0x10,0x00,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF
```

Converted in *zero* compress type

```
0x10,0x10,0x01,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x00,0x10,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF
```

Converted in *full* compress type

```
0x10,0x10,0x02,
0xFF,0x08,0x00,0x10,0xFF,0x08
```

Example implementation code for C language

```

#include <stdio.h>
#include <reg51.h>
#include "ks107.h"           // Support routines
#include "Image.h"          // Contents convert image code

void main(void)
{
    PAUSE(5);
    DISP_INIT();           // Display initialization
    for(;;)
    {
        DISP_ICON(Head,0,0); // 1. Display Image (Image, Pos. Row,Col)
        PAUSE(20);
        DISP_TEXT("Hello",5,0,56,45); // 2. Display Text (Text,Length,Size,Pos. Row,Col)
        PAUSE(20);
        DISP_TEXT("Hello",5,1,48,35); // 3. Display Text (Text,Length,Size,Pos. Row,Col)
        PAUSE(20);
    }
}

```



1.

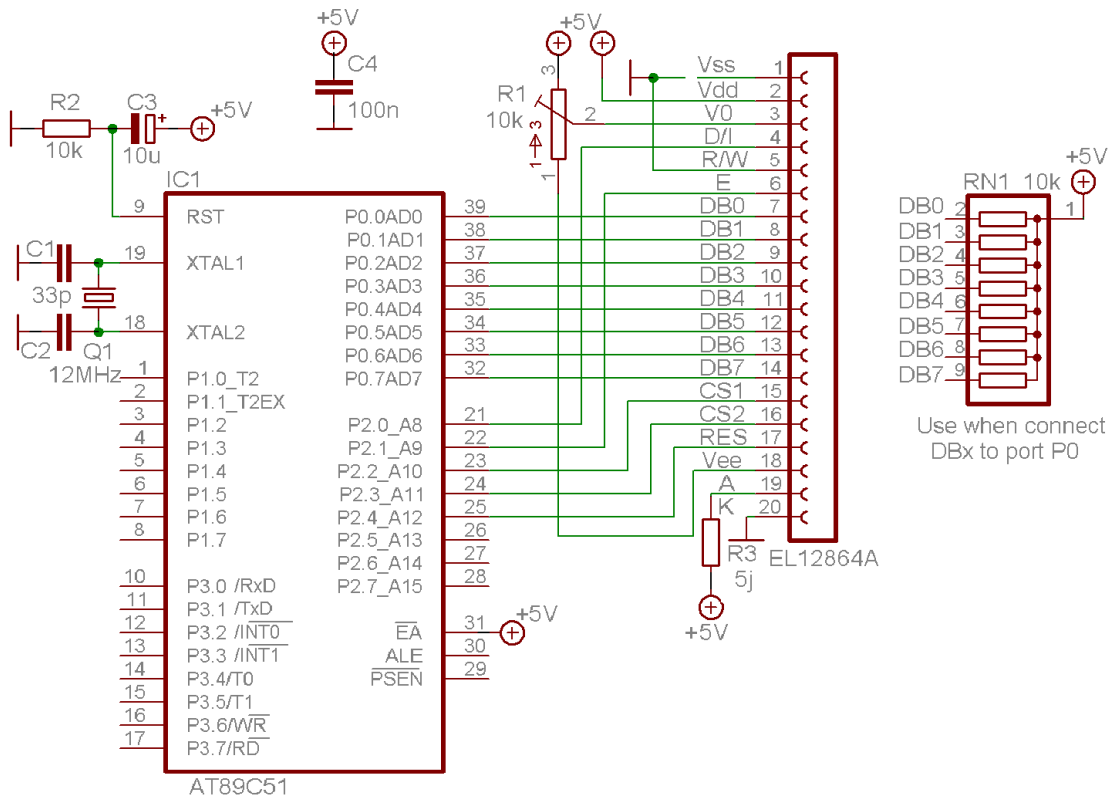


2.



3.

Example implementation scheme for LCD EL12864A and procesor AT89C51



News and actual information: <http://imfsoft.com/software/produkty/image-convertor.asp>