132x64 Development Kit [132x64x3.0]

User's Guide

1. Introduction

This User's Guide contains information about the Kent Displays, Inc. 132x64 Development Kit and related accessories.

2. Development Kit contents

Figure 1 – Development Kit Contents

- **A.** 132x64 Display Development Kit Display Assembly
  - 132x64 Display Assembly
  - Development Kit Controller Module
  - Protective Cover Assembly
- **B.** Battery assembly with four AAA batteries
- **C.** RS232 Serial cable
- **D.** AC/DC 7.5V Power Adapter
- **E.** JTAG Tool
- **F.** DB25 cable for JTAG
- **G.** Software CD

3. Assembling the 132x64 Development Kit

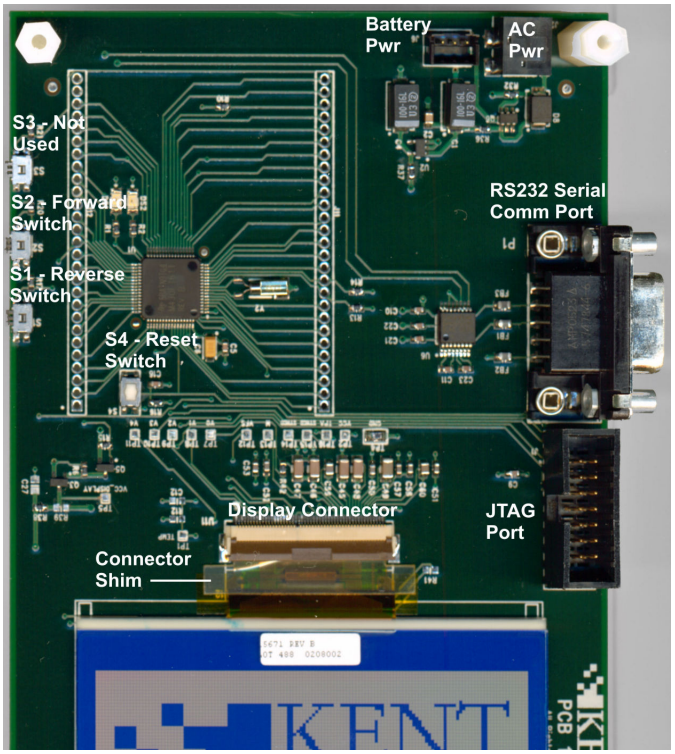
3.1 Connecting the RS232 serial cable to your PC:

Figure 2 – PC Connection

Connect the RS232 cable to a PC serial port. A 9-pin serial port connection is required; install a serial port adapter card (not supplied), if the PC motherboard has no serial port available.

Connect the other end of the RS232 cable to the 9-pin serial port connector on the 132x64 graphic display development kit controller module (see Fig. 3).

Figure 3 – Development Kit Controller Module



3.2 Connecting the JTAG Tool (Software Development Only):

The JTAG tool and parallel cable (items E and F, Fig. 1) are required for downloading firmware to the development module. Connect the parallel cable to a PC parallel port. A 25-pin parallel port connection is required; install a parallel port adapter card (not supplied), if the PC motherboard has no parallel port available. Connect the other end of the parallel cable to the JTAG tool, and connect the JTAG cable header to the JTAG port on the development board (Fig. 3).

3.3 Installing Batteries:

Place the four (4) AAA batteries into the battery holder. Connect the battery cable to the battery power port on the Development Kit Controller module (see Fig. 3). Alternately, the 7.5V A/C power adapter can be used via the A/C adapter power port on the Development Kit Controller module.

4. Stand-alone Demo Use

The 132x64 Development Kit Display Assembly can perform as a stand-alone demo by using two of the three switches at the left edge of the Development Kit Controller module. Refer to Fig. 3 for the switch locations. Pressing S1 displays the image preceding the current image. Pressing S2 advances to the next image in flash memory. S3 is not used. There is a reset switch (S4) located near S1. This switch resets the Development Kit Controller microprocessor and the Display controller.

5. Installing the Application Software

The software CD contains a SlideShow application that may be used to download custom images to the Development Kit Controller module. These images are then displayed when the system is used as a stand-alone demo as described above. For technical users developing custom firmware, BDF to C font conversion software and BMP to C image conversion software are included, along with firmware source code for the SlideShow, a Text Font Generation demo, and a Command Set demo.

Figure 4 – 132x64 Soft – InstallShield Wizard

Load the software CD into the CD drive on your PC. If the CDROM auto-run application does not start, browse to the top-level directory on your CD drive and double-click the “go_kdi.exe” file.

Select the “132x64x3.0” display from among those listed and then choose “Software – install” from the available options. This will launch the “132x64 Soft - InstallShield Wizard” window (Fig. 4) to begin the software installation process.

Follow the on-screen instructions to install the software.

6. Using the *132x64 SlideShow* Application Software

Launch the "132x64 SlideShow" application (Fig. 5) via the desktop icon or the Start Menu. The application defaults to the first available COM Port for communication with the PC. The COM port number (1 – 8) can be modified via the "Settings" pull-down menu, "Com Port" option. Problems with the COM port setting are usually caused by other software applications controlling the COM port, (e.g. PDA device software). Disable any other software that uses the COM port and restart the "132x64 SlideShow" application to resolve errors.

A default "project" file (project.txt) containing all of the images pre-loaded into the Development Kit's flash memory can be found in the 132x64 SlideShow application "Sample Images" sub-directory. This project file can be opened in the 132x64 SlideShow application via the "Open Project" command on the "File" drop down menu.

To modify the slide show, set the "Slide Number" counter (to the right of the "Current Slide" area, Fig. 5) to the number (0 – 11) of the location that will hold the new slide image bitmap file. Use the "Browse" button to locate the bitmap image file on the PC to be downloaded to the Development Kit Display. Bitmap slides must be 132 pixels wide by 64 pixels high. Any PC imaging application can be used to create the bitmap slides, but a 132x64 slide template is required, and all slides must be monochrome (1-bit). Slides 132 pixels wide by less than 64 pixels high may also be used as partial screen images. If a partial screen slide is selected, the "Starting Row" selector box is activated, allowing you to specify the vertical location of the partial image slide in the display window.

Figure 5 – 132x64 SlideShow Application Window

The slide currently displayed in the PC application window can be displayed on the Development Kit Display via the “Preview” button (below the “Current Slide” area). Using the “Preview” button has no effect on the slides stored in the Development Kit Controller’s flash memory, and the slide image sent to the display via the “Preview” button is lost once the display image is subsequently changed. Images in the “Current Slide” window can be removed from the image slide show via the “Remove Slide” button. Once a slide is removed, the next image in the slide show is moved into the vacant position, and all subsequent images are moved down in the order by one slide number. A single image can also be inserted into the slide show via the “Add Slide” button. The images held in that Slide Number location and higher are moved up to the next higher locations. Any slide moved beyond location number 11 is deleted from the slide show.

The “Index” button opens a new window to display the contents of all 12 image locations at once. The “Download” button sends all 12 images to the Development Kit Controller’s flash memory at one time. Slide shows can be saved by using the “Save as Project” command on the “File” drop-down menu. Project files contain all image bitmap data (no link to the original bitmap files is maintained). Projects can be recalled via the “Open Project” command on the “File” drop-down menu.

7. Code Composer Essentials

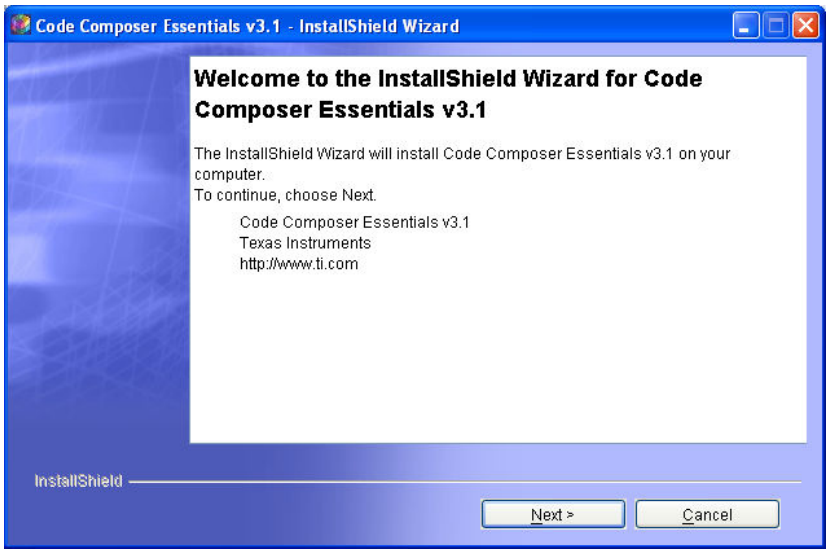
The provided firmware samples have been prepared for use with the TI Code Composer Essentials (CCE) integrated development environment. A free, limited code space version may be installed from the software CD. Installation of CCE is required only of users wishing to use this development environment to compile and download the firmware samples to the Development Kit Controller.

Installation

Load the Kent Displays’ software CD into the CD drive on your PC. If the CDROM auto-run application does not start, browse to the top-level directory on your CD drive and double-click the “go_kdi.exe” file.

Select “CCEssentials” and then choose to install “Version 3.1” from among the available options.

Figure 6 – TI CCE Installation Screen

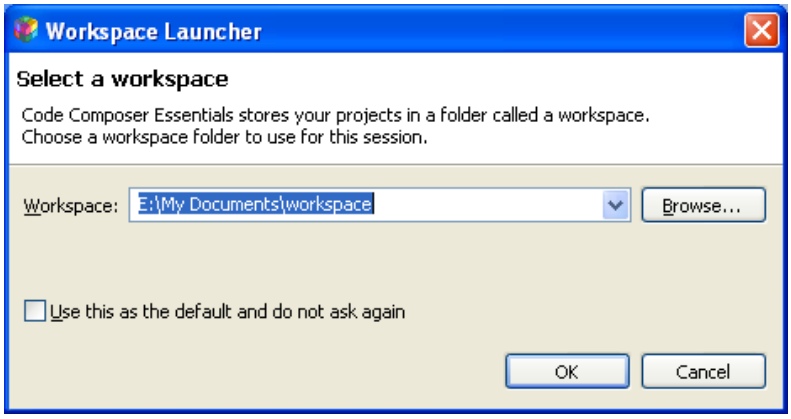


The installation process asks you to accept the TI CCE license agreement, select the location for software installation, and choose between a typical and a custom setup. A default location and a typical setup are recommended.

Launching CCE

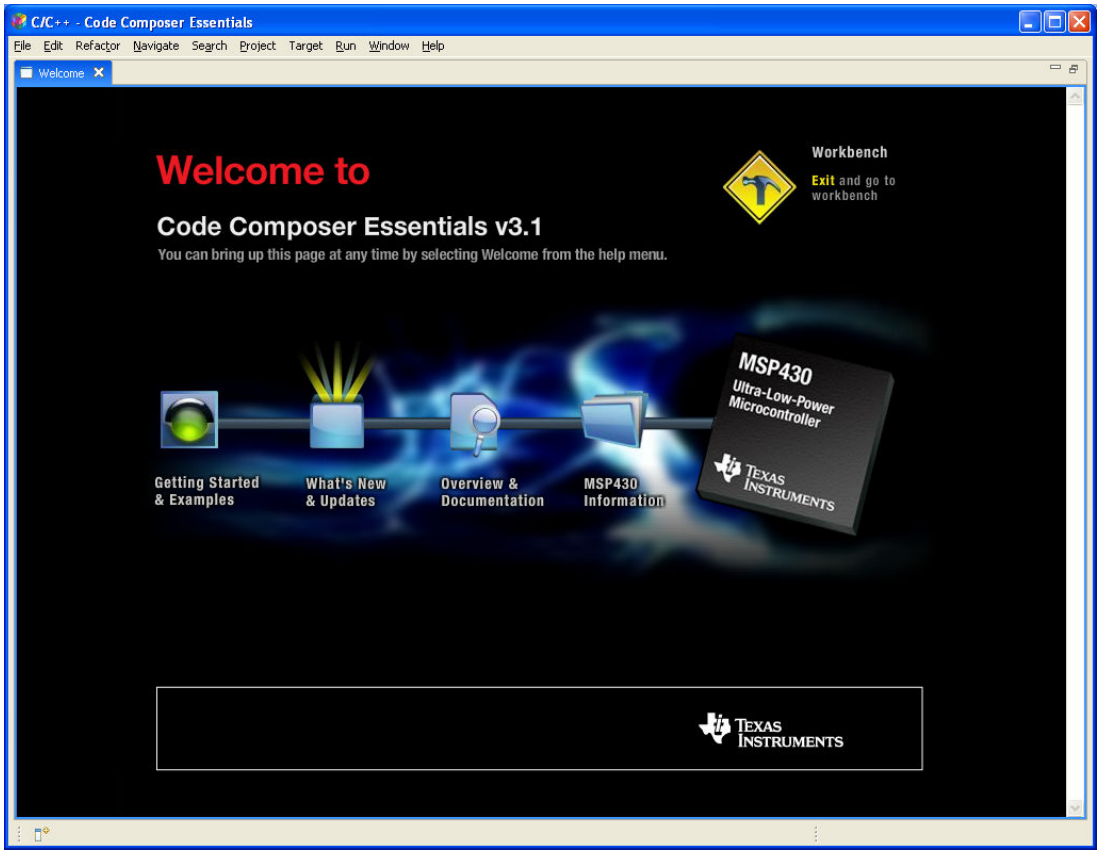
Launch the “Code Composer Essentials” application via the desktop icon or the Start Menu. The application prompts you to select a workspace via a pop-up “Workspace Launcher” window (Fig. 7). Accept the default location and click OK.

Figure 7 – CCE Workspace Launcher window



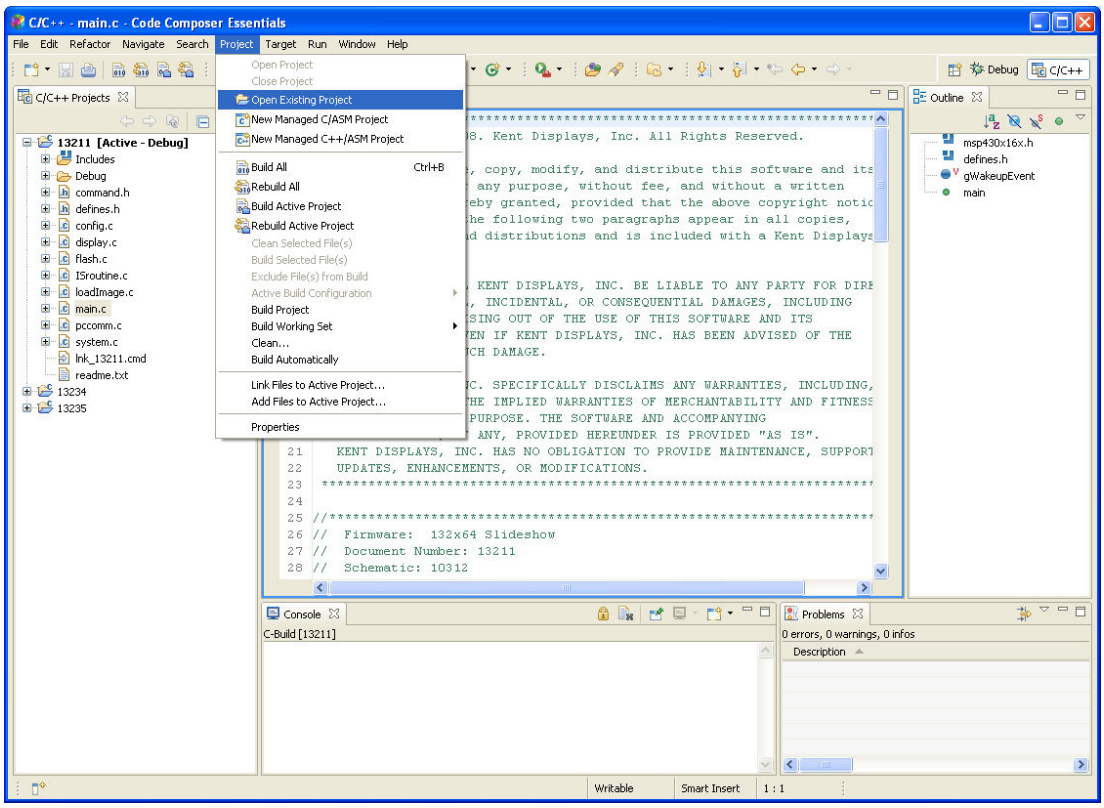
Once the workspace is selected, the CCE Welcome Screen appears (Fig. 8). The options on the welcome screen may be used for additional help in becoming familiar with the CCE compiler. Click the ‘X’ on the “Welcome” tab to close the welcome screen and proceed. This screen can be recalled at any time by clicking on “Welcome” in the “Help” drop-down menu.

Figure 8 – CCE Welcome Screen



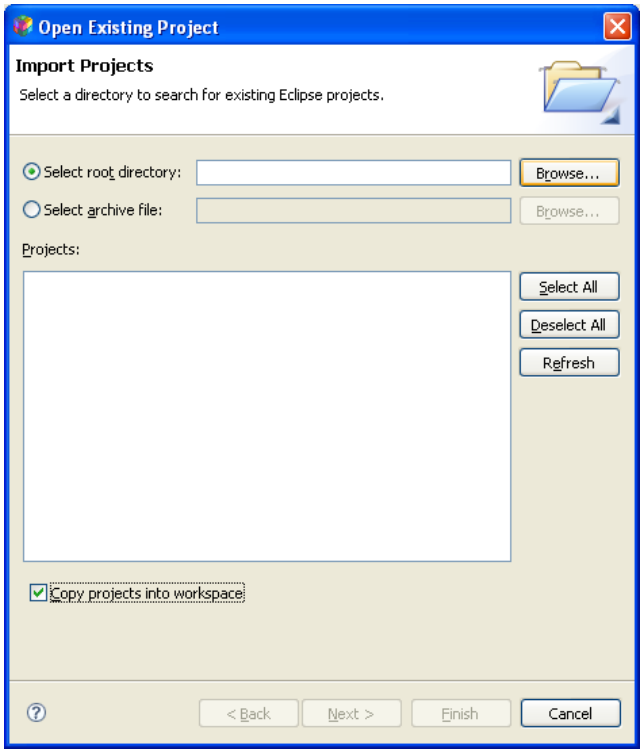
Opening a Project

Figure 9 – CCE Application Window



Click the “Project” drop-down menu button near the top of the CCE Application window (Fig. 9) and click on the “Open Existing Project” option. (Note that multiple projects have already been opened in the image in Fig. 9.) This pops-up the “Open Existing Project” window (Fig. 10).

Figure 10 – CCE Open Existing Project Window



Click the “Browse…” button associated with “Select root directory” and navigate to one of the following firmware sample directories (given relative to the destination folder of the install):

- *\Developers Kit\SlideShow FW*
- *\Developers Kit\FontDemo FW*
- *\Developers Kit\CommandDemo FW*

It is recommended to check the “Copy projects into workspace” box before clicking “Finish”. This will copy all of the sample code into a subdirectory of the workspace selected when launching CCE (Fig. 7). The working version of the code is the copy in the workspace, so all user modification of the sample code within the IDE occurs on the copies, leaving the originals in the root directory selected above unaffected.

Click on the “Finish” button at the bottom of the “Open Existing Projects” window to open the corresponding firmware project.

Building a Project

Multiple projects may be open within the IDE at one time. These projects are listed on the “C/C++ Projects” tab (Fig. 9). One of the open projects is designated as the active project, which is indicated by [Active] appearing after the project name. The active project may be selected from among the open projects by right clicking on the project name and selecting “Set as Active Project” from the menu. Ensure that the project to be built is selected as the active project.

Click the “Project” drop-down menu button near the top of the CCE Application window (Fig. 9) and click on the “Build Active Project” option.

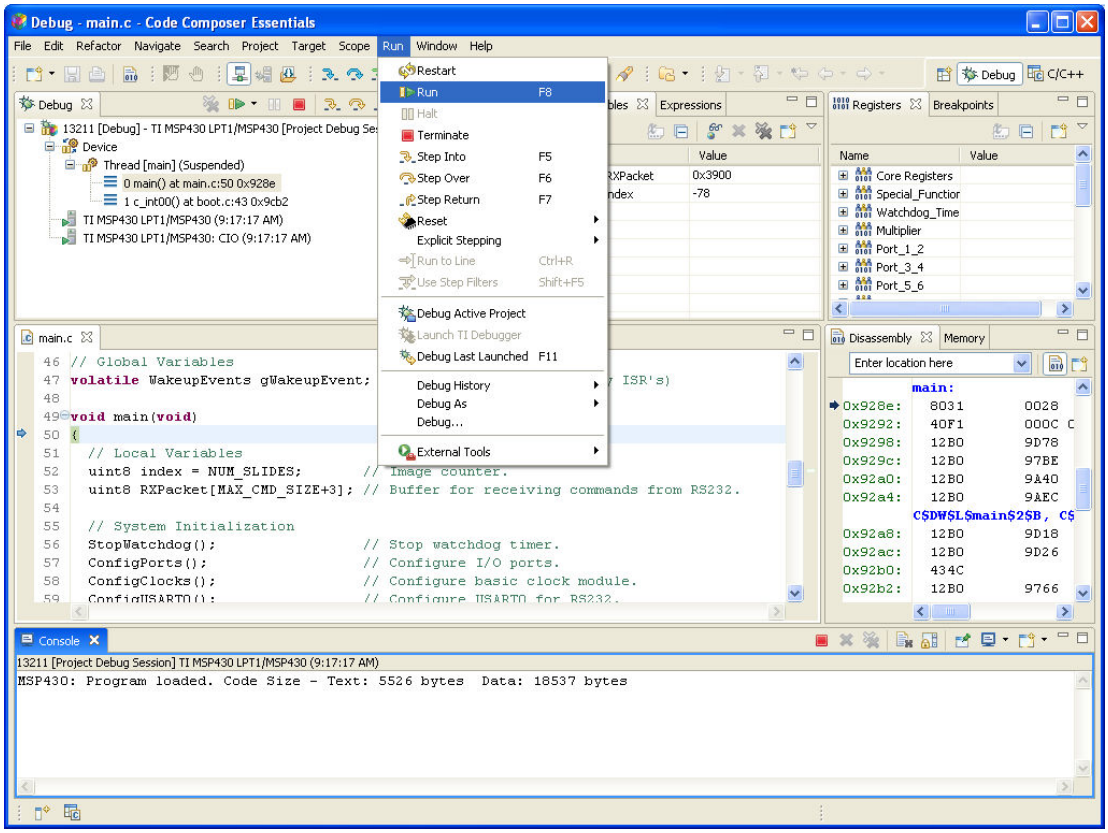
Downloading and Running a Project

Connect the JTAG tool as described in Sect. 3.2.

Ensure that the project to download has been opened, built, and set as the active project as described above.

Click on the “Run” drop-down menu button near the top of the CCE Application window (Fig. 9) and select “Debug Active Project” to download the program to the Development Kit board. This will also open the “Debug” perspective as shown in Fig. 11.

Figure 11 – CCE Debug Perspective



Click on the “Run” drop-down menu again, and select “Run” to execute the program on the Development Kit board, or remove the JTAG cable from the Development Kit board and use the switches on the board to run the code in a stand-alone mode.

The program and debugger may be stopped by clicking on the “Run” drop-down menu again and selecting “Terminate”.

Return to the original “C/C++” perspective used for managing projects and editing code by clicking on the “Window” drop-down menu and selecting “Open Perspective->C/C++”.

Note that the SlideShow firmware can be downloaded at any time to restore the Development Kit firmware to the original SlideShow demo code. However, the stored images will be cleared such that the 132x64 Slide Show application on the PC must be used to restore the images.

8. Using the 132x64 SlideShow Sample Firmware

This firmware plays a slide show of up to 12 images stored in flash. The images are downloaded to the development board using the corresponding PC application (Sect. 6).

The switches on the development board function as follows (Fig. 3):
SW1 - Displays the previous slide.
SW2 - Displays the next slide.
SW3 – NOT USED

Follow the procedure in Sect. 7 for downloading and running the "SlideShow FW" program (project #13211) on the Development Kit.

Note that the SlideShow firmware can be downloaded at any time to restore the Development Kit firmware to the original SlideShow demo code. However, the stored images are cleared such that the 132x64 Slide Show application on the PC must be used to restore the images.

9. Using the BDF to C Application Software

There are many fonts in the public domain that are distributed in the form of Glyph Bitmap Distribution Format (BDF) files. A BDF file generally encodes a single size of a single font. The BDF to C application reads in the BDF file for a desired font and generates C code that encodes the individual character bitmaps. This C code may then be added to the user's own application to write text in the desired font to a display from Kent Displays, Inc. This document explains how to use the BDF to C application.

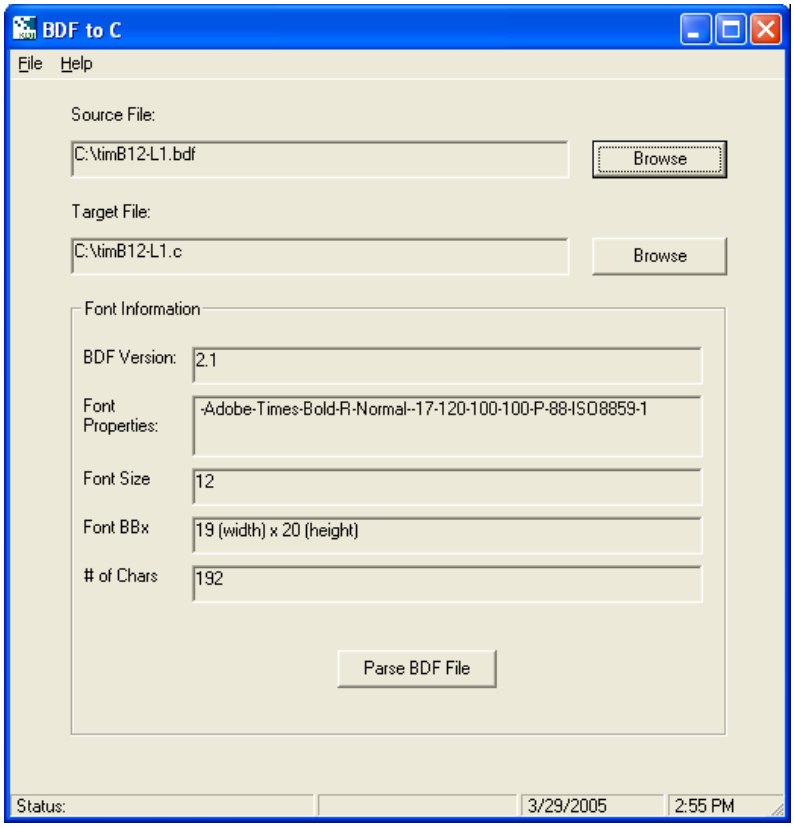
The Application GUI consists of three buttons (Fig. 12):

Browse (Source File): This button is used to locate and open the BDF file of a desired font. When a valid BDF file is opened, the Font Information area is filled in with the font properties extracted from the BDF file and the target file name is set by default to the source file name with a .c extension. The source file may also be opened using the **File->Open** menu option. The BDF files for a number of fonts are installed with the BDF to C application and are available under the application directory.

Browse (Target File): This button is used to change the target file name for the generated C code from the default.

Parse BDF File: This button is used to generate the target file from the specified BDF file.

Figure 12 – BDF to C Application Window



Using the Output

The output file (using source file timB12-L1.bdf as an example) will have the following format:

```
#include "defines.h"

const uint8 FBBy = 20;

static const uint8 timB12L1_0x00_BMP[] = {
  0xAA, 0x80,
  0x00, 0x00,
  0x80, 0x80,
  0x00, 0x00,
  0x80, 0x80,
  0x00, 0x00,
  0x80, 0x80,
  0x00, 0x00,
  0x80, 0x80,
  0x00, 0x00,
  0xAA, 0x80
};
```

```
static const uint8 timB12L1_0x20_BMP[]={
  0x00
};
⋮
const CharInfo FontBMP[] = {
  {12, 9, 11, 1, 0, timB12L1_0x00_BMP},
  ⋮
  {8, 8, 15, 0, -4, timB12L1_0xFF_BMP}
};
```

There are three main parts to the generated file. The FBBy constant gives the vertical offset in pixels for single-spaced text. The uint8 data arrays contain bitmap data for the ISO8859-1 characters encoded in the array names. Finally, the FontBMP array contains additional data about the characters. The ISO8859-1 character code is used as the index into the FontBMP array. Each element of the array is a CharInfo structure which contains the information required to correctly place and reproduce the character using the bitmap data.

The CharInfo structure is defined in defines.h as follows:

```
typedef struct {
   int8 dwidth;     // position of next char. relative to the current
   uint8 width;     // width of character in pixels

   uint8 height;    // height of character in pixels
   int8 offsetX;    // horizontal offset to left edge of bounding box
   int8 offsetY;    // vertical offset to bottom edge of bounding box
   const uint8 *CharBMP; // pointer to character bitmap defined above
} CharInfo;
```

Note that defines.h also typdef's int8 and uint8 to 8-bit signed and unsigned integers, respectively.

Refer to the following section for an example of how to use the BDF to C output code on the Development Kit.

References

1. Glyph Bitmap Distribution Format (BDF) Specification, Version 2.2, March 22, 1993, Adobe Systems Inc.
   (Available at http://partners.adobe.com/public/developer/en/font/5005.BDF_Spec.pdf)

2. GUI Development: Embedding Graphics, Part I, Niall Murphy, Embedded Systems Programming, Vol. 12, No. 7, July 1999.
   (Available at http://www.embedded.com/1999/9907/9907feat1.htm)

10. Using the 132x64 Text Generation Sample Firmware

This firmware demonstrates how to generate text on the display.

Follow the procedure in Sect. 7 for downloading and running the “FontDemo FW” program (project #13234) on the Development Kit.

Text in a different font may be displayed as follows:

1) Use the BDF to C application (Sect. 9) to generate a font source file for the desired font.
2) Replace the CharBMP.c file in the project with this new font file. Be sure to replace the version in the workspace directory if the “Copy projects into workspace” box was checked when opening the project.
3) Build, download, and run the project as described in Sect. 7.

11. Using the 132x64 BMP to C Application Software

This application takes a standard bitmap file (.bmp format) and generates C code that encodes the bitmap image. This C code may then be included in the user’s own application to put the bitmap image on the display.

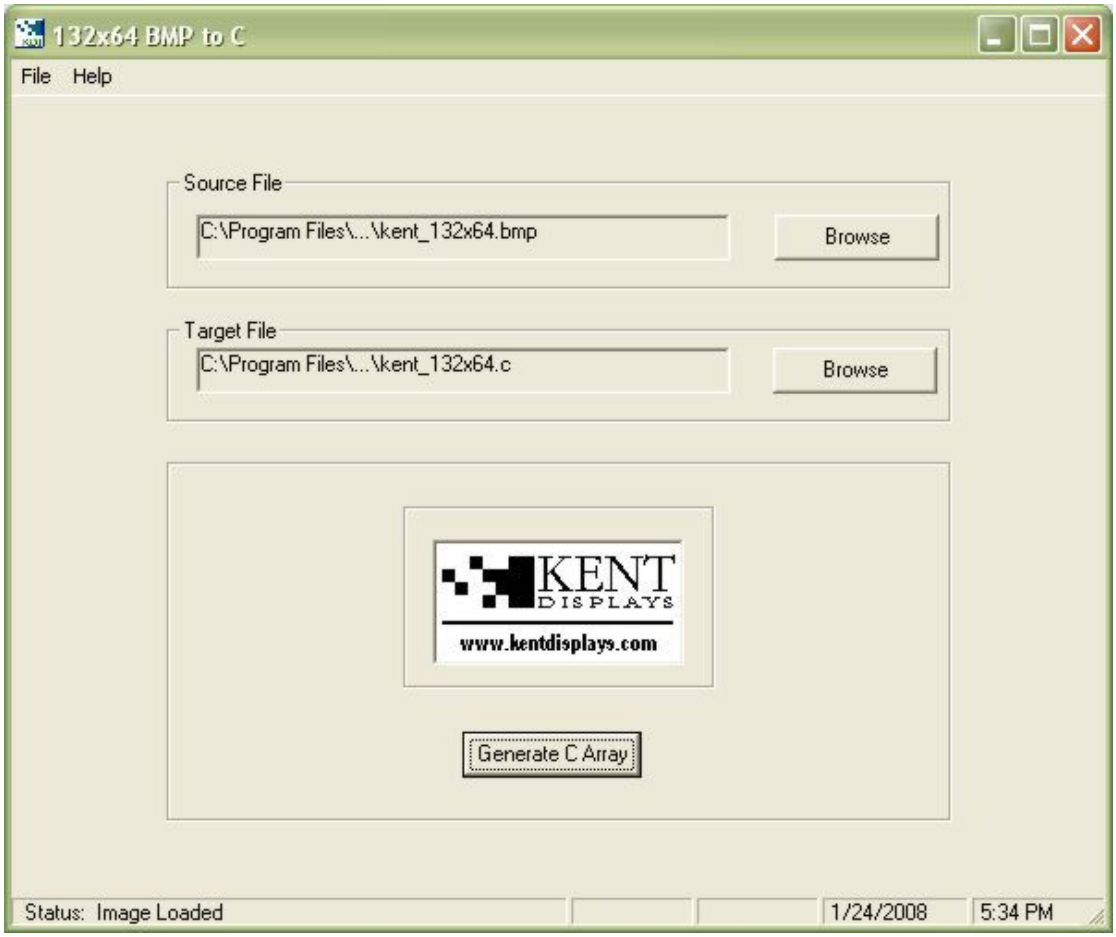
The GUI consists of three buttons (Fig. 13):

Browse (Source File): This button is used to locate and open the bitmap file for which C code is to be generated. The bitmap must be monochrome, with width equal to 132 pixels and height less than or equal to 64 pixels. A successfully opened bitmap image will be displayed in the picture box. When the bitmap file is opened, the target file name is set by default to the source file name with a .c extension. The source file may also be specified using the **File->Open** menu option.

Browse (Target File): This button is used to change the target filename for the generated C code from the default.

Generate C Array: This button is used to generate the target file from the specified source bitmap file.

Figure 13 – 132x64 BMP to C Application Window



Using the Output

The output file for a full 132x64 bitmap image named testimage.bmp will have the following format:

```
// Note:  uint8 must be typedef'd to an 8-bit unsigned integer in
defines.h.
#include "defines.h"

// ****************************************
// testimage
// ****************************************
const uint8 testimage[1056] = {
   0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF ...
}
```

A 132x64 bitmap image contains 1,056 bytes of data. The generated array is in the correct format to be directly transferred to a region of the 132x64 display memory by the user's application in order to display the bitmap. Partial screen images consist of all 132 columns and some number less than 64 rows. Because each byte of image data encodes 8 vertical pixels, the array size for partial screen images will be from 1 to 8 multiples of 132.

Refer to the note in the following section for details on how to view images generated by this application on the Development Kit display.

12. Using the 132x64 Command Demo Firmware

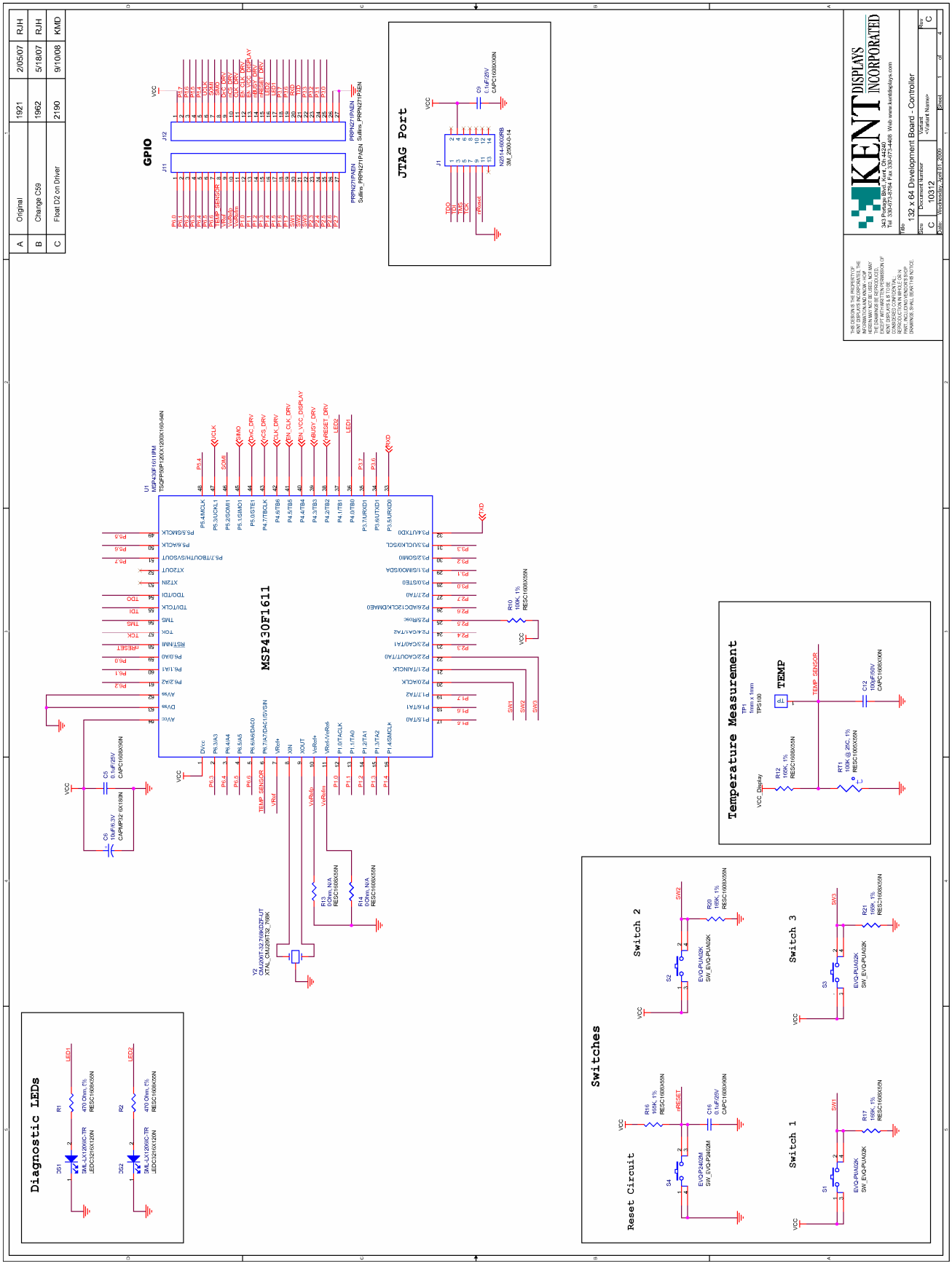
This firmware demonstrates the command set for the display. The user may follow along in the ExecuteDemo() function (in demo.c) in order to see the commands behind each demonstration.

The development board switches (SW1, SW2, and SW3) function as follows (Fig. 3):
SW1 - Executes the next demo.
SW2 - Same as SW1.
SW3 - Same as SW1.

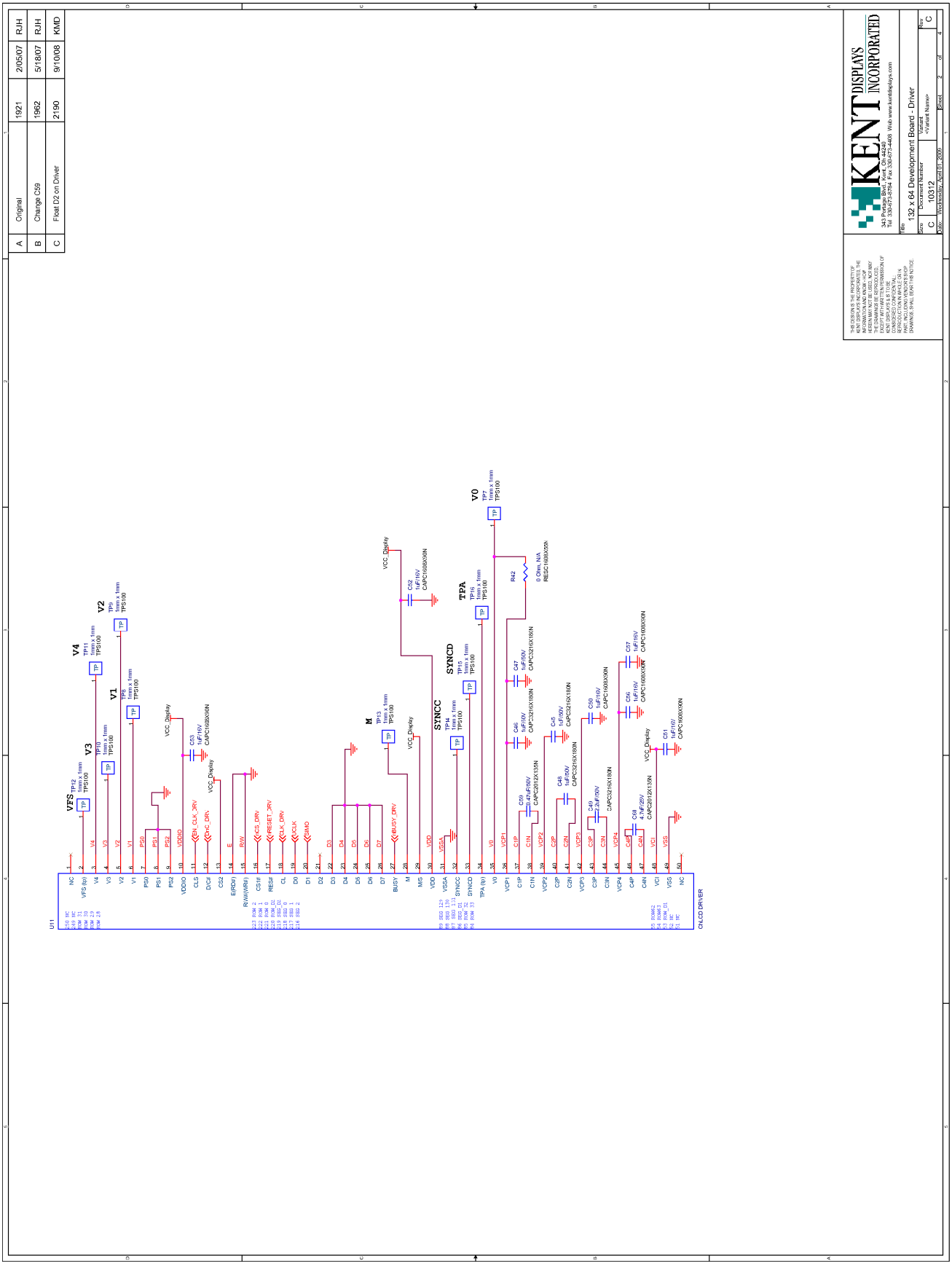
Follow the procedure in Sect. 7 for downloading and running the “CommandDemo FW” program (project #13235) on the Development Kit.

Note:
The first command demo displays a fullscreen image. The data for this image is stored in the 'Kent_132x64' data array in the file images.c. Users may replace the data in this array with data generated by the 132x64 BMP to C Application (Sect. 11) in order to display a bitmap of their choosing.

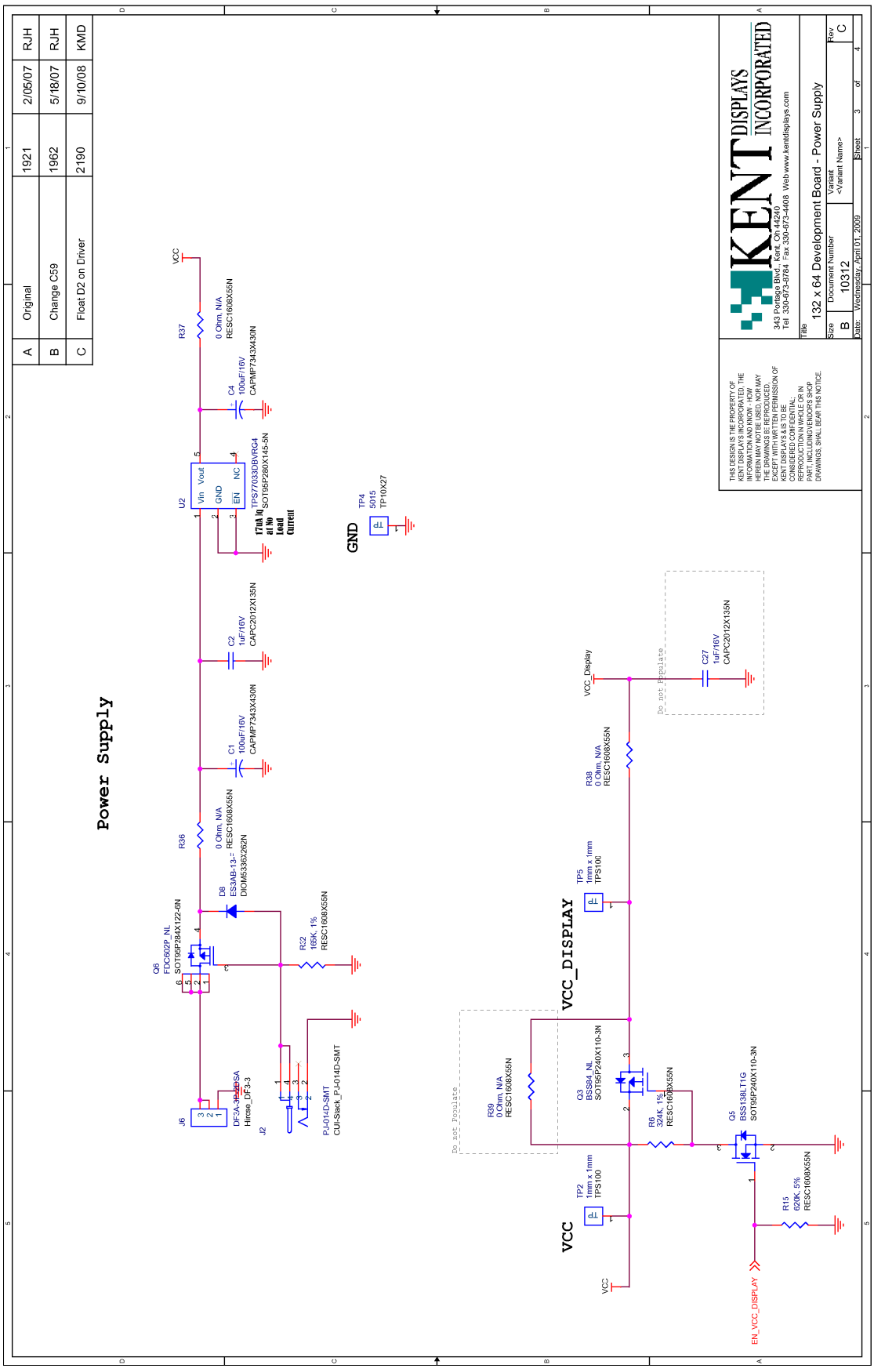
Appendix A – Development Board Schematic Drawings



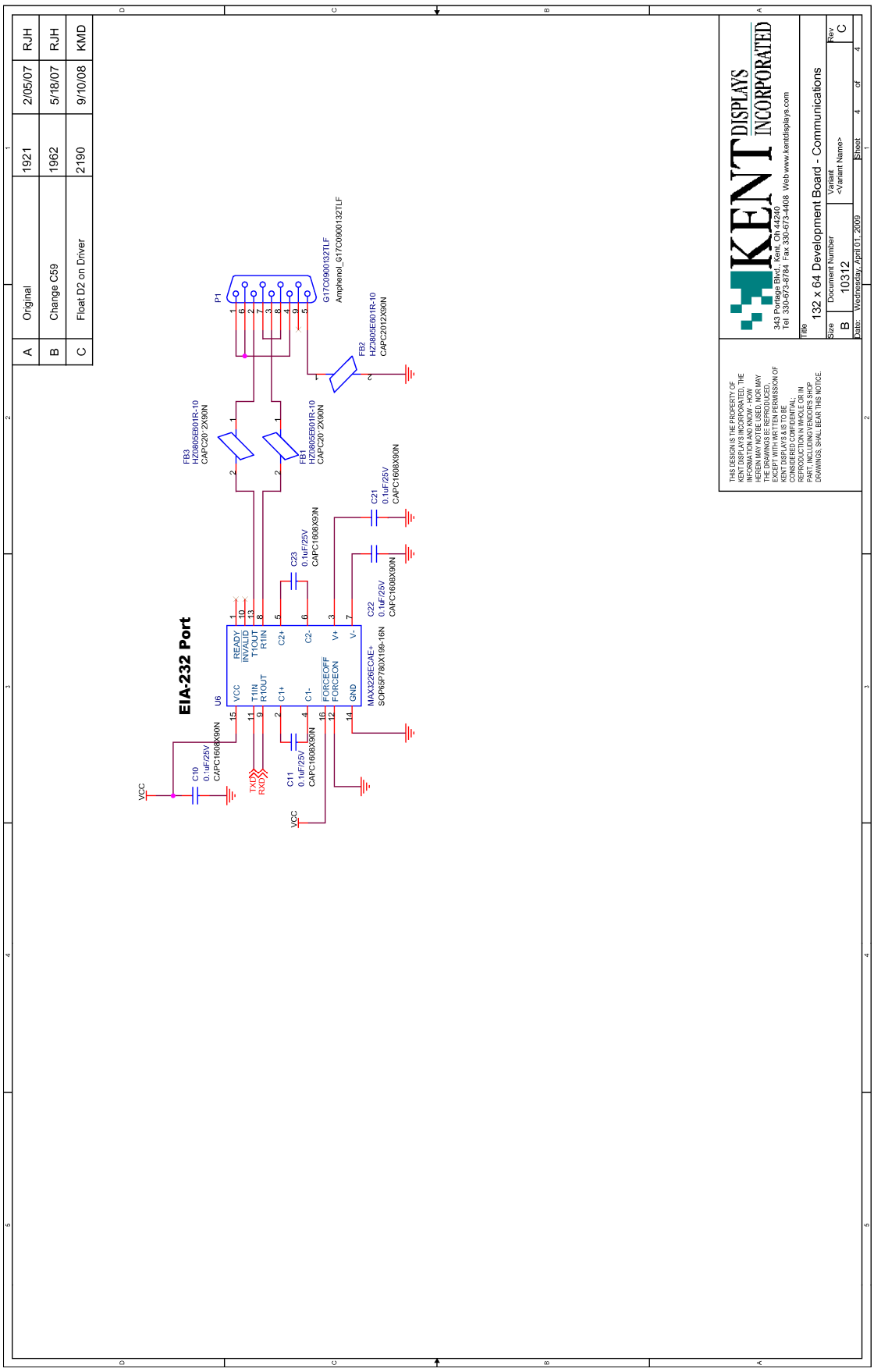
Appendix A – Development Board Schematic Drawings



Appendix A – Development Board Schematic Drawings



Appendix A – Development Board Schematic Drawings



Appendix B – Development Board Bill of Materials

Item	Qty.	Reference	Description	Manufacturer p/n
1.	2	C1, C4	Capacitor, Electrolytic, Tantalum, 100uF, 16V, 10%, 7343, SMD	Sprague, 593D107X9016E2TE3
2.	1	C2	Capacitor, Ceramic, Y5V, 1uF, 16V, 20%, 0805, SMD	Kemet, C0805C105M4VACTU
3.	8	C5, C9, C10, C11, C16, C21, C22, C23	Capacitor, Ceramic, Y5V, 0.1uF, 25V, +80/-20%, 0603, SMD	ROHM, MCH182FN104ZK
4.	1	C6	Capacitor, Electrolytic, Tantalum, 10uF, 6.3V, 10%, 3216, SMD	KEMET, T491A106K006AS
5.	1	C12	Capacitor, Ceramic, C0G, 100pF, 50V, 5%, 0603, SMD	Vishay, VJ0603A101JXACW1BC
6.	4	C45, C46, C47, C48	Capacitor, Ceramic, X7R, 1uF, 50V, 10%, 1206, SMD	TDK, C3216X7R1H105K
7.	1	C49	Capacitor, Ceramic, X7R, 2.2uF, 50V, 10%, 1206, SMD	Murata, GRM31CR71H225KA88L
8.	6	C50, C51, C52, C53, C56, C57	Capacitor, Ceramic, X7R, 1uF, 16V, 10%, 0603, SMD	Taiyo Yuden, EMK107BJ105KA
9.	1	C59	Capacitor, Ceramic, X5R, 0.47uF, 50V, 10%, 0805, SMD	Taiyo Yuden, UMK212BJ474KG-T
10.	1	C60	Capacitor, Ceramic, X5R, 4.7uF, 25V, 20%, 0805, SMD	Panasonic , ECJ-2FB1E475M
11.	1	D8	Diode, Schottky, 3A, SMB, SMD	Diodes, Inc., ES3AB-13-F
12.	2	DS1, DS2	Diode, LED, RED, 1206, SMD	Lumex, SML-LX1206IC-TR
13.	3	FB1, FB2, FB3	Miscellaneous, Ferrite, Bead Ferrite, 0805, SMD	Steward, HZ0805E601R-10
14.	1	J1	Connector, Dual In-Line, 14, Male, 2.54mm, Thru-Hole	3M, N2514-6002RB
15.	1	J2	Connector, Power, 2, Female, 1.3mm, SMD	CUI Stack, PJ-014D-SMT
16.	1	J6	Connector, Single In-Line, 3, 2mm, Thru-Hole	Hirose, DF3A-3P-2DSA
17.	1	P1	Connector, D-SUB, 9, Female, DB-9, Thru-hole	Amphenol, G17C0900132TLF
18.	1	Q3	FET, P Channel, -50, -0.13, +/- 20, -2, SOT-23, SMD	Fairchild, BSS84_NL
19.	1	Q5	FET, N Channel, 50, 200mA, +/- 20, 1.5, SOT-23, SMD	ON Semiconductor, BSS138LT1G
20.	1	Q6	FET, P Channel, -20, -5.5, +/- 12, 1.5, SOT-23-6, SMD	Fairchild, FDC602P_NL
21.	2	R1, R2	Resistor, Chip, Thick Film, 470 Ohm, 5%, 0603, SMD	Vishay, CRCW0603470RJKEA
22.	1	R6	Resistor, Chip, Thick Film, 324K, 1%, 0603, SMD	Vishay, CRCW0603324KFKEA

Item	Qty.	Reference	Description	Manufacturer p/n
23.	1	R10	Resistor, Chip, Thick Film, 100K, 1%, 0603, SMD	Vishay, CRCW0603100KFKEA
24.	6	R12, R16, R17, R20, R21, R32	Resistor, Chip, Thick Film, 165K, 1%, 0603, SMD	Vishay, CRCW0603165KFKEA
25.	6	R13, R14, R36, R37, R38, R42	Resistor, Chip, Thick Film, 0 Ohm, N/A, 0603, SMD	Vishay, CRCW06030000Z0EA
26.	1	R15	Resistor, Chip, Thick Film, 620K, 5%, 0603, SMD	Vishay, CRCW0603620KJKEA
27.	1	RT1	Resistor, Chip, NTC, 100K @ 25C, 1%, 0402, SMD	Murata, NCP15WF104F03RC
28.	3	S1, S2, S3	Miscellaneous, Switch, Switch, Momentary LightTouch Side-Push, SMD	Panasonic, EVQ-PUA02K
29.	1	S4	Miscellaneous, Switch, Switch, Momentary LightTouch, SMD	Panasonic, EVQ-P2402M
30.	1	TP4	Connector, Test Point, 1, 1x2.7mm, SMD	Keystone, 5015
31.	1	U1	IC, Microcontroller, 16bit, 48K Flash, 10K byte RAM, 64 pin QFP, SMD	Texas Instruments, MSP430F1611IPM
32.	1	U2	IC, Linear Regulator, 50mA low dropout, SOT23-5, SMD	Texas Instruments, TPS77033BVRG4
33.	1	U6	IC, Communication, RS-232 Transceiver, SSOP-16, SMD	Maxim, MAX3226ECAE+
34.	1	U11	Connector, Flat Flex, 50, Female, 0.5mm, SMD	Hirose, FH12S-50S-0.5SH(55)
35.	1	Y2	Crystals_Oscillators, Crystal, 32.768KHz, 12.5pF, SMD	Citizen, CMJ206T-32.768KDZF-UT
36.	1	PCB	PCB, 132 X 64 Development Board, Lead Free	KDI#10314

Appendix C – BDF Copyright Information

Fonts distributed with BDF to C contain the following copyright notices:

```
SUN MICROSYSTEMS, INC., AT&T, AND BIGELOW & HOLMES
MAKE NO REPRESENTATIONS ABOUT THE SUITABILITY OF
THIS SOURCE CODE FOR ANY PURPOSE. IT IS PROVIDED "AS IS"
WITHOUT EXPRESS OR IMPLIED WARRANTY OF ANY KIND.
SUN  MICROSYSTEMS, INC., AT&T AND BIGELOW  & HOLMES,
SEVERALLY AND INDIVIDUALLY, DISCLAIM ALL WARRANTIES
WITH REGARD TO THIS SOURCE CODE, INCLUDING ALL IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE. IN NO EVENT SHALL SUN MICROSYSTEMS,
INC., AT&T OR BIGELOW & HOLMES BE LIABLE FOR ANY
SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES,
OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA
OR PROFITS, WHETHER IN AN ACTION OF  CONTRACT, NEGLIGENCE
OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
WITH THE USE OR PERFORMANCE OF THIS SOURCE CODE.

************************************************************************

Copyright (c) 1987, 1994 Digital Equipment Corporation

The name "Bitstream" is a registered trademark of Bitstream, Inc.
Permission to use this trademark is hereby granted only in association
with the images described in this file.

The X Consortium, and any party obtaining a copy of these files from
the X Consortium, directly or indirectly, is granted, free of charge, a
full and unrestricted irrevocable, world-wide, paid up, royalty-free,
nonexclusive right and license to deal in this software and
documentation files (the "Software"), including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons who receive
copies from any such party to do so.  This license includes without
limitation a license to do the foregoing actions under any patents of
the party supplying this software to the X Consortium.

Digital Equipment Corporation make no representations about the
suitability of this software for any purpose.  It is provided "as
is" without express or implied warranty.
```

IMPORTANT NOTICE

Kent Displays, Inc. (KENT DISPLAYS) Development Kits are provided under the following conditions:

Development Kits sold by Kent Displays are intended for EVALUATION and/or ENGINEERING DEVELOPMENT PURPOSES ONLY. Development Kits are not intended for commercial use or resale. Development Kits may not be complete in terms of regulatory testing and approvals - including product safety considerations. Development Kits are considered prototypes and are not subject to the European Union directive on electromagnetic compatibility and therefore may not meet the requirements of the directive.

KENT DISPLAYS WARRANTS ITS MANUFACTURED GOODS TO BE FREE FROM DEFECTS IN MATERIAL AND WORKMANSHIP FOR 1 YEAR, FOLLOWING THE DATE OF SHIPMENT. IF ANY OF THE GOODS ARE FOUND BY KENT DISPLAYS TO BE DEFECTIVE, SUCH GOODS WILL, AT KENT DISPLAYS' OPTION, BE REPLACED OR REPAIRED AT KENT DISPLAYS' COST. IF KENT DISPLAYS IS UNABLE TO REPAIR OR REPLACE ANY SUCH GOODS, KENT DISPLAYS SHALL REFUND THE ORIGINAL PURCHASE PRICE IN FULL. THE PARTIES HERETO EXPRESSLY AGREE THAT BUYER'S SOLE AND EXCLUSIVE REMEDY AGAINST KENT DISPLAYS SHALL BE FOR THE REPAIR, REPLACEMENT, OR REFUND FOR DEFECTIVE GOODS AS PROVIDED HEREIN. THE FOREGOING WARRANTY IS IN LIEU OF ALL OTHER WARRANTIES, EXPRESS OF IMPLIED, INCLUDING THOSE OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. NO AFFIRMATION OF KENT DISPLAYS, BY WORDS OR ACTION, OTHER THAN AS SET FORTH IN THIS SECTION SHALL CONSTITUTE A WARRANTY. KENT DISPLAYS' WARRANTY DOES NOT APPLY TO ANY GOODS WHICH HAVE BEEN SUBJECTED TO MISUSE, MISHANDLING, MISAPPLICATION, NEGLECT, ACCIDENT, IMPROPER INSTALLATION, MODIFICATION (INCLUDING BUT NOT LIMITED TO USE OF UNAUTHORIZED PARTS OR ATTACHMENTS), OR ADJUSTMENT OR REPAIR PERFORMED BY ANYONE OTHER THAN KENT DISPLAYS OR ONE OF KENT DISPLAYS' AUTHORIZED AGENTS. ANY CLAIM BY BUYER WITH REFERENCE TO THE GOODS SOLD HEREUNDER SHALL BE DEEMED WAIVED BY THE BUYER UNLESS SUBMITTED IN WRITING TO KENT DISPLAYS WITHIN THE EARLIER OF (i) THIRTY (30) DAYS FOLLOWING THE DATE BUYER DISCOVERED OR BY REASONABLE INSPECTION SHOULD HAVE DISCOVERED, ANY CLAIMED BREACH OF THE FOREGOING WARRANTY, OR (ii) (15) FIFTEEN MONTHS FOLLOWING THE DATE OF SHIPMENT. ANY CAUSE OF ACTION FOR BREACH OF THE FOREGOING WARRANTY SHALL BE BROUGHT WITHIN (6) SIX MONTHS FROM THE DATE THE ALLEGED BREACH WAS DISCOVERED OR SHOULD HAVE BEEN DISCOVERED, WHICHEVER OCCURS FIRST

The user assumes all responsibility and liability for proper, safe handling of the Development Kit. The user also indemnifies KENT DISPLAYS from all claims arising from the handling or use of the Development Kit. The user has been made aware that the Development Kit received may not be regulatory compliant or agency certified (FCC, UL, CE, etc.). Due to the open design of the Development Kit, it is the user's responsibility to take any and all appropriate precautions with regard to electrostatic discharge (ESD). KENT DISPLAYS' LIABILITY SHALL BE LIMITED TO REPAIRING OR REPLACING PARTS FOUND BY KENT DISPLAYS TO BE DEFECTIVE, OR AT KENT DISPLAYS' OPTION, TO REFUNDING THE PURCHASE PRICE OF SUCH GOODS OR PARTS THEREOF.

Except to the extent of the indemnity set forth above, neither part shall be liable to the other for any indirect, special, incidental, or consequential damages.

KENT DISPLAYS assumes no liability for customer product design, software performance, applications assistance, or infringement of patents or services described herein.

Please carefully review the product datasheet for this development kit and the underlying display product for important information about voltages and temperatures prior to using the Development Kit. Contact KENT DISPLAYS with additional questions about this product.

Persons handling the Development Kit must have electronics training and observe proper laboratory practices.

No license is granted under any patent right or other intellectual property right of KENT DISPLAYS covering or relating to any device, process, or combination in which such KENT DISPLAYS products or services are or might be used. KENT DISPLAYS specifically reserves all of its rights in and to its proprietary information and trade secrets.

Kent Displays, Inc.
343 Portage Blvd.
Kent, OH 44240
support@kentdisplays.com