Platform Migration of a LCD Smart Transit Sign

Computer Engineering Department
California Polytechnic State University, San Luis Obispo

Shea Clifford

March 22, 2010

Contents

Table of Figures

Project Planning

Proposal / Context & Motivation

The current incarnation of San Luis Obispo's Efficient Deployment of Advanced Public Transportation Systems (EDAPTS) project consists of several large electromechanical schedule signs, animated by microcontrollers. The sizable energy demands of these solar-powered signs led to bulky units, requiring substantial support masts installed specifically for this system. Thanks to subsequent advances in both microcontroller and display technologies, former graduate student Aniefon Ekanem developed a much less power hungry implementation. The reduced power demands and component sizes of this design bring with them a drastic reduction in the size and weight of the entire assembly, allowing mounting on existing bus stop poles at a vastly reduced per-installation cost.

This project aims to complete both the hardware and firmware migration from the existing microcontroller and display to the newer, more efficient devices. Additionally, inaccuracies in the estimated arrival time hints at a possible need to reevaluate the prediction algorithms currently in use as well investigate any sources of latency in the radio notification system. Any excess project time may focus on devising a methodology for remote firmware updates.

Design Requirements

- The system shall not exceed $1000 in components
- The system shall not exceed $1000 in development costs
- Installation of the system shall not exceed $200
- The system must work with the existing power system
- Connectors and protocols will be standardized where possible
- The system's firmware will be modifiable in place
- The system must be able to survive a minimum of 20 days of incident weather
- The system will constructed in a modular fashion where possible
- The system will comply with all applicable ADA specifications and guidelines
- The system will comply with all applicable FCC specifications and guidelines

Specifications

- Existing Power supply
  - 24 V System
  - Two 12 V horizontally oriented, 10 W solar cells
  - Four 12 V batteries
  - 1.1~1.6 AH / day generation @ 24 V
  - 34 AH storage capability @ 24 V

- Pager
  - Daviscomms TMR1P
  - 3~5 V or 8~15 supply
  - 5~15 mA drain
  - RS232 or TTL level outputs

- Existing CPU
  - Motorola 68HC11
  - -0.3~7 V supply
  - 0.4 ~ V_{DD} -0.8 V output
  - 10 µA standby consumption

- Ekanem's development CPU
  - Texas Instruments MSP430F1611
  - 4.15-8 MHz
  - 48KB+256B Flash Memory
  - 10KB RAM
  - 1.8V~3.6 V supply
  - 330 µA active consumption
  - 1.1 µA standby consumption
  - 0.2 µA off (with RAM retention)

- Current Proposed CPU
  - Xilinx Spartan 3E-500 FG320
  - 500 gates
  - 50 MHz
  - -0.5~3.75 V output driver supply
  - 800 µA active consumption

Analysis

Economic
Today's economic conditions do not lend themselves to non-critical public spending projects. Projects such as this one only become viable if they offer a level of service that increases ridership to the point where income offsets the cost of development and installation.
The other way in which the project can support itself is if it reduces operational costs. By more accurately informing passengers of arrival times, we reduce the chance of a "near-miss" rider slowing down a departure.

Environmental
Public transportation has the potential to relieve significant road congestion and reduce the city's carbon footprint. The main challenge is convincing potential riders that buses offer the same level of convince as their personal cars. To this end, increasing the reliability of scheduling and offering them up-to-the-minute information makes mass transit more appealing.

Sustainability
These installations consume very little power and what they do draw is from a renewable resource. The only direct sustainability is with the lifecycle of the components, specifically the batteries. However, the batteries in these installations will last for quite a long time with no need for disposal. If we adhere to RoHS standards in component choice and fabrication, we can minimize the toxic materials that would otherwise enter landfill at end of lifecycle. Also, this project increases the efficiency of mass transit, reducing drain on non-renewable fuel resources.

Manufacturability
These units have already proven themselves to be manufacturable on a small scale. If the new board modules are designed with reverse compatibility in mind, there need not be any significant changes to the manufacturing process. This would allow them to easily replace the boards in the existing signs.

Ethical
In the interest of accessibility, great care should be taken to make sure that these signs adhere to ADA guidelines. One of the cornerstones of a solid public transportation system is that it is accessible to all users, regardless of any special considerations they may have.

Health & Safety
Few health or safety issues arise when in the context of this project. It is, however, that the signs pose no risk to safety. They must not obstruct visibility or be a significant collision hazard.

Social
Encouraging mass transit encourages a spirit of community. It makes the city seem more accessible. It also makes riders feel that city agencies care about their daily lives and is willing to go out of its way to provide a higher standard of service.

Political

Devices such as these demonstrate the city's commitment to efficient and easy public transit. Gestures such as these improve relations between the transit agency and its users. If the people feel comfortable with the transit agency and have confidence in its reliability, ridership will increase, bringing in more funding.

Functional Overview

Hardware

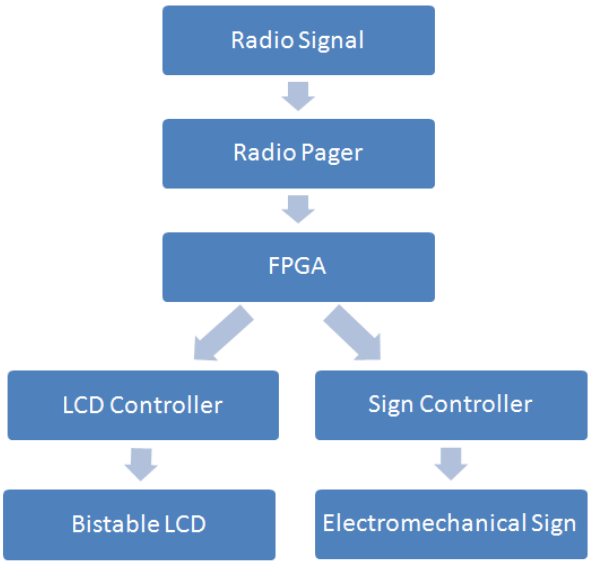

Figure 1 – Data Flow

Firmware

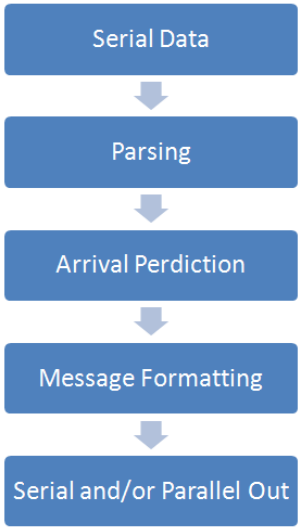

Figure 2 – Processing Flow

Estimated Cost

The bistable LCD solution proposed by Ekanem's work had a cost of $712 for materials alone. A development board for the MSP430F1611 microcontroller used in the aforementioned design cost approximately $44.00. The new intended development platform for this project, the Digilent Nexys-2 has a price point of $99. While this seems like a sharp increase in price, the old microcontroller development board is comparatively barebones to the Nexys-2 board and the final design has no need for the full development board. The Xilinx Spartan-3E FPGA (500k gate ver.) FPGA, upon which the new solution is based, is priced at $31.56/unit, not including any bulk price break. The existing MSP430F1611 sells for $14.41/unit, not including any bulk price break. This is still roughly double the cost of the Spartan chip, but the ease of maintenance and development, as well as increased expansion capability, can be reasonably expected to offset a cost difference $ 17.15/sign unit. There is no reason to expect a sharp increase in complexity, so the manufacturing cost should remain roughly equivalent.

Existing electromechanical signs can be retrofitted with Spartan-3E based control boards to standardize the system. With a $30 IC and a conservative PCB fabrication estimate of $50/board, it would probably save time and effort to just use a prefabricated Nexys-2 or similar development board (assuming it meets any ruggedization requirements. The additional cost would be minimal compared to the time and money needed in properly designing a custom, more minimal board and having it fabricated.

As can be seen from the Projected Timeline, this project will span nearly eight weeks. Predicting about 2-3 work hours per day for the duration of the project, a total of 140 hours can be expected. Due to the nature of this project, using my own salary elsewhere seems reasonable; I am the one working on the project and I am, as shown by example, willing to trade my time, at my level of education/ability for $12/hr., resulting in a hypothetical net developmental labor cost of approximately $1,680.

Figure 3 -Projected Timeline

Design Decisions

Hardware Platform

I selected the Digilent Nexys2 development board, based on the Xilinx Spartan 3e FPGA as the platform for this project. My decision was based primary on maintainability and modularity considerations. The CalPoly students use the Nexys2 board throughout the digital design and embedded systems curriculum. This allows any student working on this project will be familiar with the tools and design considerations involved in making changes or additions to the system. This cuts down on potential development time in the future. Using an FPGA based system, rather than a hardware microcontroller, allows us a very high degree of customizability. By taking care to adhere to modular design principles, the system can be quickly and easily adapted by replacing modules at different points in the datapath. This grants the flexibility to do something as drastic as changing the physical interface between the radio pager and the rest of the system without removing any hardware, allowing upgrades to the tracking and radio infrastructure to be affected in-place.

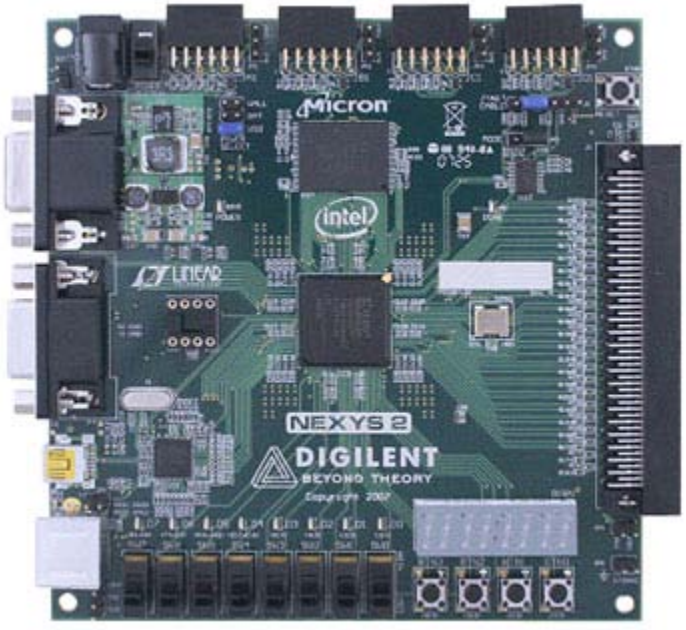

Figure 4 - Nexys2 Development Board

Firmware/Software Development Environment

I have chosen to design the system's firmware and software in Xilinx's EDK (Embedded Development Kit). This environment allows for the quick instantiation of "IP Cores," or virtualized hardware devices. It gives me the ability to make drastic hardware changes with minimal downtime. The suite has an integrated C compiler and automatically generates appropriate headers with each hardware alteration. Normally, a Digilent specific application called Adept is needed to communicate with the Nexys2 board from a PC. The USB interface on the Nexys2 board is essentially a universal programmer on-board with very basic JTAG functionalities. More advanced features like real-time debugging, direct access to the onboard flash memories, and programming from within the Xilinx EDK require the use of a Xilinx Platform USB cable. This cable attaches directly to the JTAG headers, bypassing the third-party USB interface. This approach became necessary when I realized that the combined hardware configuration and software data was too large for the FPGA itself and the external flash memory needed to be utilized. The Xilinx EDK provides a functionality by which the user's program can be placed in flash memory and the FPGA can be configured with a bootloader that properly executes the application in the flash. With the EDK, this process is accomplished with a single click.

Figure 5 - Xilinx Platform Cable (University Program Ver.)

Nexys2 Boot Process

By using the Platform Studio's flash writer application, we can load up to 16MB of program data into active RAM by a process of boot loading from flash ROM at start up.

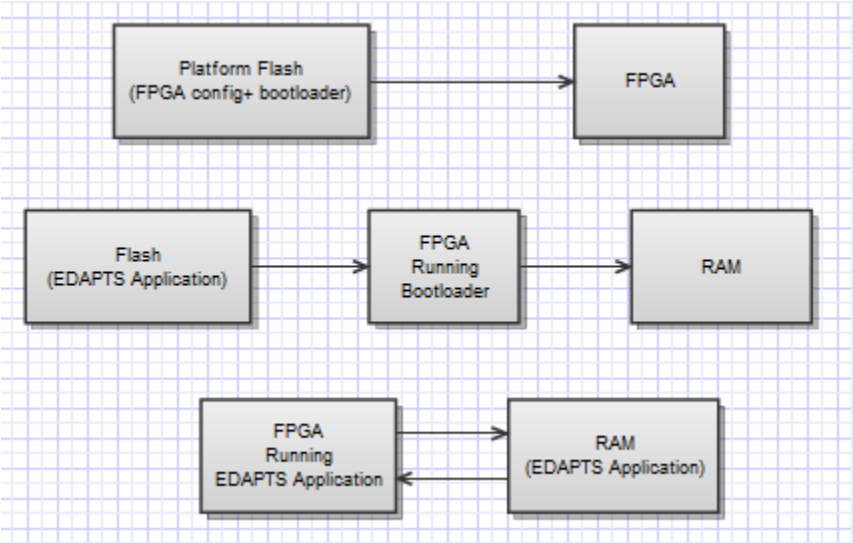

Figure 6 - Booting

Hardware/Firmware

LCD Module

Note: This section is provided for reference. Due to the issue and solution covered in Appendix A, the physical LCD module is not used in the demonstration.

The Kent Displays 1/4 VGA Cholesteric Display Module is a bistable LCD panel with a built in driver. It displays a two tone image at a resolution of 320x240 pixels. Bistability is what differentiates this from standard LCD panels. It draws power only on change, not on image hold. As such, it lacks a backlight which might make its visibility questionable at night. However, it displays a very clear, well contrasted bitmap during the daylight hours. The LCD module's onboard control communicates with external devices by means of the SPI (Serial Peripheral Interface) bus. SPI is a clocked serial protocol that allows a single master device to control several slave devices. In this implementation, the Nexys2 will be the master and the LCD module, the slave. SPI consists of four lines: a serial clock, a slave select (this would be multiple lines if there were more than one slave device), a master-to-slave data line, and a slave-to-master data line. A character is sent to the LCD by pulling slave select low and sending data over master-out/slave-in on falling clock. This is not a screen character, but rather a raw byte. Various screen operations are performed by sending a command code, followed by the appropriate number of data bytes for that command.

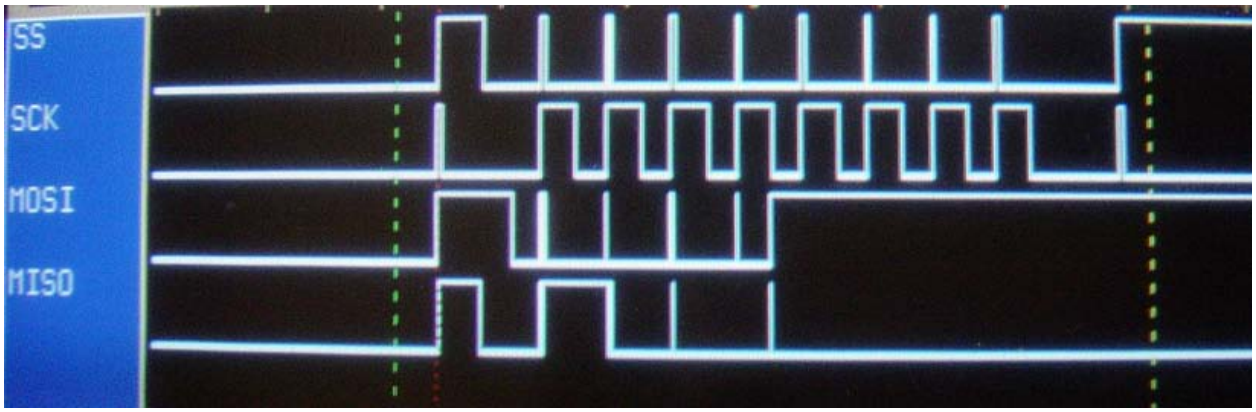

Figure 7 - Waveform of a single byte transmission, specifically, 0x1F

LCD Interface

The LCD module's connector is a 16 conductor wide ribbon cable. Since the connections on the Nexys2 that I was to be utilizing are female headers, an adaptor was needed. I found an appropriately sized ribbon cable connector on an aged hard drive. The connector was desoldered from its board and connected to leads. To avoid strain-related joint failures, I formed a shell of hot-melt glue around the connections. I placed the other ends of these leads into a pre-fabricated six pin female header to six pin female header cable and created another strain relief out of hot-melt glue. Through the use of a dual-sided six pin male header, this assembly can be plugged directly into the Nexys2 or into a breadboard such that the signals can be monitored by external equipment. Unfortunately, the power lead from the ribbon cable connector had to be run independently due to the LCD requiring different voltages for logic high and power. The pager/pager emulation device connects to the Nexys2 by means of its onboard RS232 connector.

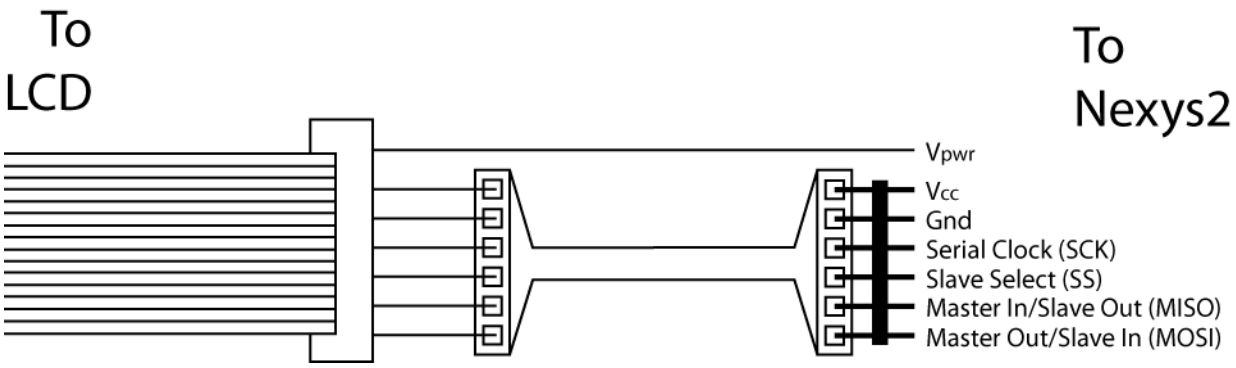

Figure 8 - LCD Connections

FPGA Configuration

Using the Xilinx EDK, I have chosen to instantiate a MicroBlaze softcore processor. This processor is a basic microprocessor with quite a range of capabilities. Within the configuration of the cores, I have also added a SPI interface, a UART, a flash memory controller, a timer/counter, and an interrupt handler. This configuration allows for a program held in the sizable external flash memory to be executed by the microprocessor, using the screen and pager as I/O devices, all triggered by either an incoming pager packet or a countdown timer. Using an interrupt driven system allows for power conservation as the system can be put into sleep mode, pending an event.

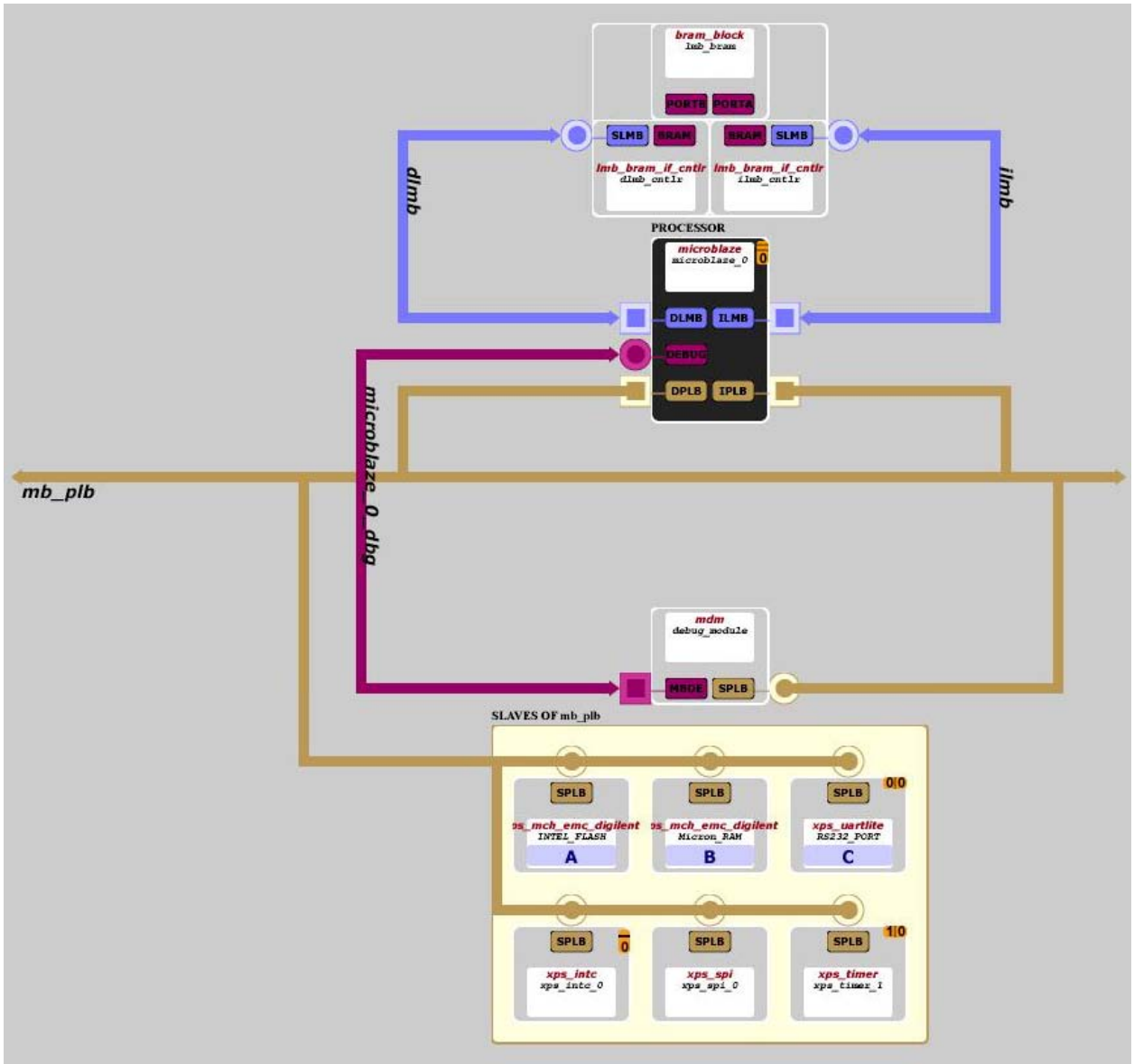

Figure 9 - IP Core layout

Software

The firmware component of the project has a fairly simple flow. A pair of counters is maintained by an interrupt driven function. When a sufficient number of ticks are reached, the screen is either redrawn to display multiple routes to riders or the ETAs of all routes are decremented per one minute passing in real time. If a radio message is waiting in the FIFO to be processed, it is parsed by the comm routines and the corresponding data structures are updated as needed.

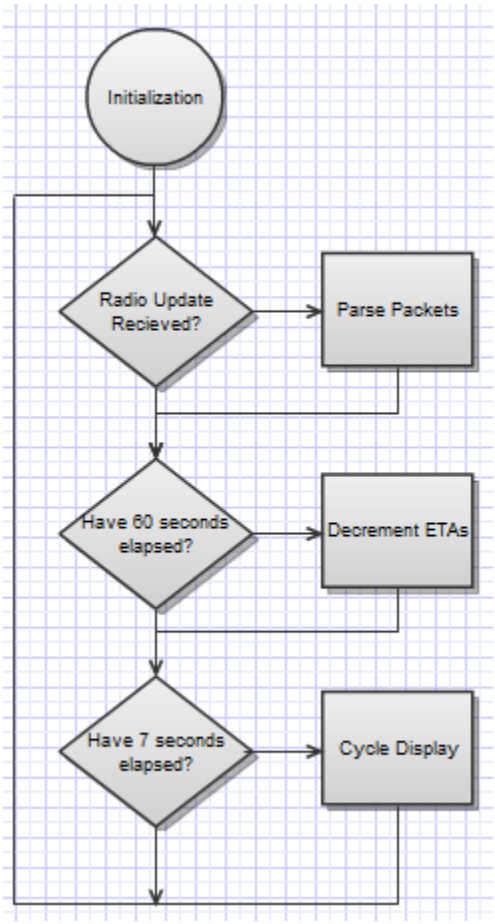

Figure 10 - Program Flow

Problems Encountered

#1

Problem:	Initial verification project does not work at all.
Cause :	Unknown, suspect bad configuration of the IP cores
Solution:	Use the Digilent-supplied Base System Builder libraries to auto-generate the appropriate configuration and constraints

#2

Problem:	Contrast of screen suddenly became very dark and is hard to see.
Cause :	Unknown, probably a hardware failure
Solution:	None. Simulation of screen used in its stead. See Appendix A.

#3

Problem:	Resource usage error when synthesizing system.
Cause :	Hardware description and software combined too large to fit on FPGA
Solution:	Opted to store the program in the flash memory chip on the Nexys2

#4

Problem:	Digilent Adept unable to access flash memory
Cause :	Functionality simply not supported
Solution:	Purchased Xilinx Platform USB Cable

#5

Problem:	Xilinx Platform USB Cable not recognized by any application
Cause :	Xilinx EDK and drivers behave improperly under 64-bit Windows
Solution:	Changed development environment to 32-bit Windows Vista

#6

Problem:	Xilinx EDK unable to access flash memory.
Cause :	Digilent's provided EMC module does not show up to EDK as a multi-channel EMC due to naming issues.
Solution:	Repack Digilent's EMC module into a new module without a Digilent-specific naming scheme.

#7

Problem:	ETAs decrement by two per minute, rather than one
Cause :	Unknown
Solution:	Unresolved at current time

Conclusions

The most important realization this project has caused me was how overwhelmingly important time management and attention to scheduling are. Allotting sufficient time for a given task keeps the entire project on track. Writing documentation as you go is also crucial in order to understand previous code when viewed at a later date. Not only are the ideas clearer at the time of inception, it provides a solid reference for the latter parts of the project.

A proper development environment was surprisingly difficult to achieve. There were problems getting the Xilinx tools to run, installing the JTAG cable drivers, IP core versioning problems, issues with slightly differing proprietary layouts. The bulk of this project was spend not on writing the application code itself, but rather configuring the cores that the FPGA was to use. This part was particularly painstaking because of the vague nature of the EDK's error messages. It is exceptionally difficult to debug software when there are no guarantees about the hardware on which it resides.

However, once the FPGA configuration was completed, the software development tools were quite nice. The ability to use an actual IDE rather than the text editor built into EDK streamlined code production. Another important note in the development process was the use of version control. Timely and regular check-ins of code allowed me to recover from errors and, in one case, the deletion of several files.

As for the EDAPTS system proper, I have become doubtful that the Nexys2 board is a good approach. While it's very customizable and expandable, it may just be too much hardware from the problem at hand. There are cheaper, simpler, microcontrollers that could accomplish similar results with lower power draw and more economical pricing. However, I cannot discount the fact that once the development environment was set up properly, implementation was a very speedy process. The APIs associated with the IP cores, though not immediately clear, provide a nice level of abstraction that makes it quite easy enact changes. For example, configuring the speed, parity, and protocol of the UART was done through a simple drop down menu, rather than loading a series of constants into specific registers. That is actually what happens, but it is at a level where the programmer does not need to concern themselves with it.

If development is to be continued by CalPoly students, I believe the Nexys2 board remains a reasonable choice. As it is used by students in the CPE169/269/329 series, it becomes quite familiar and, despite its oddities, is more familiar than picking up an entirely new architecture.

Schedules

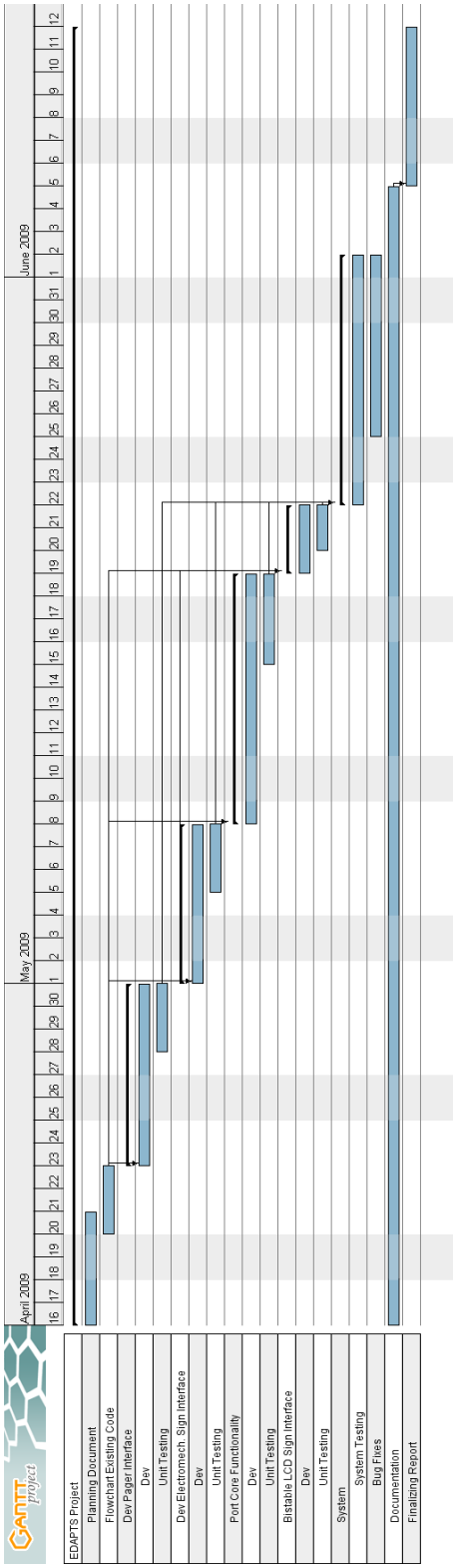

Figure 11 – Projected Timeline

Appendix A – Peripheral Simulator

Summary

I had no access to the radio pager module, as it was being used by another student. Since the output of the pager is a simple serial stream, it is easily mimicked by any sort of terminal emulator (Hyperterm, PuTTY, etc). By simply pasting or typing in data from the radio logs captured previously, the system will act as if it was receiving data over the air.

After a significant period of time trying to fix the apparent contrast problem with the LCD module and speaking with Kent Display's support, I concluded that it was most likely a hardware failure. Without a second LCD readily available, I decided that I would try a similar approach. To this end, I've commented out the SPI references in the screen drawing code and replaced them with a simple write command. Rather than loading the screen array into memory via SPI and issuing a redraw command, it will simply dump the array out though the UART.

I then created a GUI in Java using Swing that displays a 320x240 window which simulates the appearance of the hardware LCD. The serial connection is achieved through the JavaComm API. As there's only one COM port being used to connect to the board, it was necessary that the radio and screen simulators actually be part of the same application, so that they may share a single open connection. This application was made for simple utility and testing of the project, therefore it is not very fully featured. For example, it is hard-coded to port COM4, 9600 baud, 8 data bits, 1 stop bit, no parity, and a 320x240 display matrix. It does not support partial screen redraws. It assumes that every 9600 bytes received represent a single bitmap.

Application Code

BitmapArea.java

```
package srproj.peripheralsim;

import java.awt.Color;
import java.awt.Graphics;

import javax.swing.JComponent;


//Draws a given matrix as a raster of 1x1 white rectangles
public class BitmapArea extends JComponent {
  public byte[] screen;
  public int cursor;

  public BitmapArea(){
    super();
    screen = new byte[9600];
    cursor = 0;
  }
  public void paint(Graphics g){
    g.setColor(Color.BLUE);
    g.fillRect (0, 0, 320, 240);
    g.setColor(Color.WHITE);
    for(int i=0; i<240; i++){
      for(int j=0; j<40; j++){
        for(int k=0; k<8; k++){
          if(((screen[i*40+j] & (0x1 << (7-k)))!=0)){
            g.fillRect(j*8+k, i, 1, 1);
          }
        }
      }
    }
  }
}
```

PeripheralSimulator.java

```
package srproj.peripheralsim;

import java.awt.*;
import java.awt.event.*;
import java.io.*;
import java.util.Date;
import java.util.TooManyListenersException;

import javax.comm.*;
import javax.swing.*;

public class PeripheralSimulator {

  //When true, serial data draws to bitmap.  When false, to console.
  private static final boolean ENABLE_DISPLAY = true;

  private static OutputStream os;
  private static InputStream is;
  private static BitmapArea bmp;
  private static JTextField msg;

  public static void main(String[] args){
    CommPortIdentifier   portId = null;
    SerialPort port = null;


    try {
      portId = CommPortIdentifier.getPortIdentifier("COM4");


      System.out.println("Opening port "
          + portId.getName());

      if (portId.isCurrentlyOwned())
      {
        System.out.println("Detected "
            + portId.getName()
            + " in use by "
            + portId.getCurrentOwner());
        System.exit(1);
      }

      port = (SerialPort)portId.open("PeripheralSimulator", 3000);
      port.setSerialPortParams(115200, SerialPort.DATABITS_8, SerialPort.STOPBITS_1,
                               SerialPort.PARITY_NONE);
      port.addEventListener(new RxListener());
      port.notifyOnDataAvailable(true);
      port.enableReceiveTimeout(30);

      os = port.getOutputStream();
       is = port.getInputStream();

       //Display Screen
```

```
      JFrame f = new JFrame("ChLCD Simulator");
      f.setSize(326, 265);
      f.setResizable(false);
      Container fcontent = f.getContentPane();
      bmp = new BitmapArea();
      f.add(bmp);
      fcontent.setBackground(Color.gray);
      f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
      f.setVisible(true);

      //Display Radio
      JFrame r = new JFrame("Radio Simulator");
      r.setSize(500, 62);
      r.setLocation(0, 270);
      r.setResizable(false);
      r.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
      Container rcontent = r.getContentPane();
      rcontent.setLayout(new FlowLayout());
      JButton send = new JButton("Send");
      send.addActionListener(new TxListener());
      msg = new JTextField(35);
      r.add(msg);
      r.add(send);
      r.setVisible(true);

      while(true){
        Thread.currentThread();
        Thread.sleep(500);
      }

    } catch (NoSuchPortException e) {
      System.out.println("Port does not exist");
    } catch (PortInUseException e) {
      System.out.println("Port in use by another application ("+portId.getCurrentOwner()+")");
    } catch (UnsupportedCommOperationException e) {
      System.out.println("Unsupported comm operation");
    } catch (TooManyListenersException e) {
      System.out.println("Too many listeners attatched");
    } catch (IOException e) {
      System.out.println("Could not get streams");
    } catch (InterruptedException e) {
      e.printStackTrace();
    }
    finally{
      port.close();
    }
  }

  static class RxListener implements SerialPortEventListener{
    @Override
    public void serialEvent(SerialPortEvent arg0) {
      int toRead = 1;
      int read = 0;

      try {
        toRead = is.available();
```

```
        if(ENABLE_DISPLAY){
          char next;
          while(toRead > 0){

            read = is.read(bmp.screen, bmp.cursor,
                                                bmp.screen.length-bmp.cursor);
            bmp.cursor += read;

            toRead = is.available();

          }
          if(bmp.cursor == bmp.screen.length){
            bmp.cursor = 0;
          }
          bmp.repaint();
        }
        else{
          while(is.available()>0){
            System.out.printf("%c", is.read());
          }
        }
      } catch (IOException e) {
        System.out.println("IOException");
      }
    }
  }

  static class TxListener implements ActionListener{

    @Override
    public void actionPerformed(ActionEvent arg0) {
      try {
        os.write(msg.getText().getBytes());
        msg.setText("");
      } catch (IOException e) {
        System.out.println("IOException");
      }
    }

  }
}
```

Appendix B – Firmware Code

Bitmaps.h

```
#ifndef BITMAPS_H_
#define BITMAPS_H_

  unsigned char BMP_BG[] = {
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x7F, 0xFF,   0xFF, 0xF0, 0x00, 0x0F,   0xFF, 0xFF, 0xFF, 0xFF,   0x80, 0x00, 0xE0, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xFF,   0xFF, 0xFE, 0x00, 0x0F,
 0xFF, 0xFF, 0xFF, 0xFF,   0x80, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,
 0xF0, 0x3F, 0xFF, 0xFF,   0xFF, 0xFE, 0x00, 0x0F,   0xFC, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x7F, 0xFF,   0xFF, 0xFF, 0x80, 0x0F,   0xFF, 0xFF, 0xFF, 0xFF,   0x80, 0x00, 0xE0, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,   0xF0, 0x3F, 0xFF, 0xFF,   0xFF, 0xFE, 0x00, 0x1F,
 0xFE, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xFF,   0xFF, 0xFF, 0xC0, 0x0F,
 0xFF, 0xFF, 0xFF, 0xFF,   0x80, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,
 0xF0, 0x3F, 0xFF, 0xFF,   0xFF, 0xFE, 0x00, 0x1F,   0xFE, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x7F, 0xFF,   0xFF, 0xFF, 0xE0, 0x0F,   0xFF, 0xFF, 0xFF, 0xFF,   0x80, 0x00, 0xE0, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,   0xF0, 0x3F, 0xFF, 0xFF,   0xFF, 0xFE, 0x00, 0x1F,
 0xFE, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xFF,   0xFF, 0xFF, 0xF0, 0x0F,
 0xFF, 0xFF, 0xFF, 0xFF,   0x80, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,
 0xF0, 0x3F, 0xFF, 0xFF,   0xFF, 0xFE, 0x00, 0x3F,   0xFF, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x7F, 0xFF,   0xFF, 0xFF, 0xF8, 0x0F,   0xFF, 0xFF, 0xFF, 0xFF,   0x80, 0x00, 0xE0, 0x00,
 0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,   0xF0, 0x3F, 0xFF, 0xFF,   0xFF, 0xFE, 0x00, 0x3F,
 0xFF, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xFF,   0xFF, 0xFF, 0xF8, 0x0F,
 0xFF, 0xFF, 0xFF, 0xFF,   0x80, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,
 0xF0, 0x3F, 0xFF, 0xFF,   0xFF, 0xFE, 0x00, 0x3F,   0xFF, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x7F, 0xC0,   0x00, 0x3F, 0xF8, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
```

```
0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,   0xF0, 0x3F, 0xFF, 0xFF,   0xFF, 0xFE, 0x00, 0x7F,
0xFF, 0x80, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xC0,   0x00, 0x1F, 0xFC, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,
0x00, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x00, 0x7F,   0xFF, 0x80, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xC0,   0x00, 0x0F, 0xFC, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x00, 0xFF,
0xFF, 0x80, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xC0,   0x00, 0x07, 0xFC, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,
0x00, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x00, 0xFF,   0xFF, 0xC0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xC0,   0x00, 0x07, 0xFC, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x00, 0xFF,
0x7F, 0xC0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xC0,   0x00, 0x07, 0xFC, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,
0x00, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x01, 0xFF,   0x7F, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xC0,   0x00, 0x07, 0xFC, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x01, 0xFF,
0x7F, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xC0,   0x00, 0x07, 0xFC, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,
0x00, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x03, 0xFE,   0x3F, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xC0,   0x00, 0x0F, 0xFC, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x03, 0xFE,
0x3F, 0xF0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xC0,   0x00, 0x1F, 0xF8, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,
0x00, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x03, 0xFC,   0x1F, 0xF0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xC0,   0x00, 0x7F, 0xF8, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x07, 0xFC,
0x1F, 0xF0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xFF,   0xFF, 0xFF, 0xF8, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,
0xC0, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x07, 0xFC,   0x1F, 0xF8, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xFF,   0xFF, 0xFF, 0xF0, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,   0xC0, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x0F, 0xF8,
0x0F, 0xF8, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xFF,   0xFF, 0xFF, 0xE0, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,
0xC0, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x0F, 0xF8,   0x0F, 0xFC, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xFF,   0xFF, 0xFF, 0xC0, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,   0xC0, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x0F, 0xF8,
0x0F, 0xFC, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xFF,   0xFF, 0xFF, 0x80, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,
0xC0, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x1F, 0xF0,   0x07, 0xFC, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xFF,   0xFF, 0xFE, 0x00, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,   0xC0, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x1F, 0xF0,
0x07, 0xFE, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xFF,   0xFF, 0xF0, 0x00, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,
0xC0, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x3F, 0xF0,   0x03, 0xFE, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xFF,   0xFF, 0xF0, 0x00, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,   0xC0, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x3F, 0xE0,
0x03, 0xFE, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xC0,   0x7F, 0xF8, 0x00, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,
0x00, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x3F, 0xE0,   0x03, 0xFF, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xC0,   0x1F, 0xFC, 0x00, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x7F, 0xC0,
0x01, 0xFF, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xC0,   0x0F, 0xFE, 0x00, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,
0x00, 0x00, 0x00, 0xFF,   0x80, 0x00, 0x7F, 0xFF,   0xFF, 0xFF, 0x80, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xC0,   0x07, 0xFF, 0x00, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0xFF,   0x80, 0x00, 0xFF, 0xFF,
0xFF, 0xFF, 0x80, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xC0,   0x03, 0xFF, 0x80, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,
0x00, 0x00, 0x00, 0xFF,   0x80, 0x00, 0xFF, 0xFF,   0xFF, 0xFF, 0x80, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xC0,   0x01, 0xFF, 0xC0, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0xFF,   0x80, 0x00, 0xFF, 0xFF,
0xFF, 0xFF, 0xC0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xC0,   0x01, 0xFF, 0xC0, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,
0x00, 0x00, 0x00, 0xFF,   0x80, 0x01, 0xFF, 0xFF,   0xFF, 0xFF, 0xC0, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xC0,   0x00, 0xFF, 0xE0, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0xFF,   0x80, 0x01, 0xFF, 0xFF,
0xFF, 0xFF, 0xC0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xC0,   0x00, 0x7F, 0xE0, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,
0x00, 0x00, 0x00, 0xFF,   0x80, 0x01, 0xFF, 0xFF,   0xFF, 0xFF, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xC0,   0x00, 0x3F, 0xF0, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0xFF,   0x80, 0x03, 0xFF, 0xFF,
0xFF, 0xFF, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xC0,   0x00, 0x3F, 0xF8, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,
0x00, 0x00, 0x00, 0xFF,   0x80, 0x03, 0xFE, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xC0,   0x00, 0x1F, 0xF8, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0xFF,   0x80, 0x07, 0xFE, 0x00,
0x00, 0x3F, 0xF0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xC0,   0x00, 0x1F, 0xFC, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,
0xF8, 0x00, 0x00, 0xFF,   0x80, 0x07, 0xFC, 0x00,   0x00, 0x1F, 0xF0, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xC0,   0x00, 0x0F, 0xFE, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,   0xF8, 0x00, 0x00, 0xFF,   0x80, 0x07, 0xFC, 0x00,
```

```
0x00, 0x1F, 0xF8, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xC0,   0x00, 0x07, 0xFE, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,
0xF8, 0x00, 0x00, 0xFF,   0x80, 0x0F, 0xFC, 0x00,   0x00, 0x1F, 0xF8, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xC0,   0x00, 0x07, 0xFF, 0x00,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,   0xF8, 0x00, 0x00, 0xFF,   0x80, 0x0F, 0xF8, 0x00,
0x00, 0x0F, 0xF8, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xC0,   0x00, 0x03, 0xFF, 0x00,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,
0xF8, 0x00, 0x00, 0xFF,   0x80, 0x1F, 0xF8, 0x00,   0x00, 0x0F, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x7F, 0xC0,   0x00, 0x01, 0xFF, 0x80,   0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,   0xF8, 0x00, 0x00, 0xFF,   0x80, 0x1F, 0xF0, 0x00,
0x00, 0x07, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xC0,   0x00, 0x01, 0xFF, 0xC0,
0x00, 0x3F, 0xE0, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,
0xF8, 0x00, 0x00, 0xFF,   0x80, 0x1F, 0xF0, 0x00,   0x00, 0x07, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xFF,   0xF8, 0x00, 0x00, 0xFF,   0x80, 0x3F, 0xF0, 0x00,
0x00, 0x07, 0xFE, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF,   0xFF, 0xFF, 0xFF, 0xFF,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
```

```
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
```

```
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
```

```
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
```

```
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x01,
0xE0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x20, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x01,   0xE0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x60, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x01,
0xE0, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x01, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x01, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x01, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x01, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3C, 0xFC, 0x3E, 0x01,
0xE0, 0xF1, 0xF0, 0x1E,   0x03, 0xC7, 0xFC, 0x07,   0xE0, 0x03, 0xF8, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3D, 0xFE, 0xFF, 0x81,   0xE0, 0xF7, 0xFC, 0x1E,   0x03, 0xC7, 0xFC, 0x1F,
0xF8, 0x0F, 0xFE, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0x81,
0xE0, 0xFF, 0xFC, 0x1E,   0x03, 0xC7, 0xFC, 0x3F,   0xFC, 0x0F, 0xFE, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3F, 0x1F, 0xC7, 0xC1,   0xE0, 0xFC, 0x3E, 0x1E,   0x03, 0xC1, 0xE0, 0x7C,
0x3E, 0x1E, 0x1F, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3E, 0x0F, 0x83, 0xC1,
0xE0, 0xF8, 0x1E, 0x1E,   0x03, 0xC1, 0xE0, 0x78,   0x1E, 0x1E, 0x0C, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3C, 0x0F, 0x03, 0xC1,   0xE0, 0xF0, 0x1E, 0x1E,   0x03, 0xC1, 0xE0, 0xF0,
0x0E, 0x1E, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3C, 0x0F, 0x03, 0xC1,
0xE0, 0xF0, 0x1E, 0x1E,   0x03, 0xC1, 0xE0, 0xF0,   0x0F, 0x1F, 0xC0, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3C, 0x0F, 0x03, 0xC1,   0xE0, 0xF0, 0x1E, 0x1E,   0x03, 0xC1, 0xE0, 0xFF,
0xFF, 0x0F, 0xF8, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3C, 0x0F, 0x03, 0xC1,
0xE0, 0xF0, 0x1E, 0x1E,   0x03, 0xC1, 0xE0, 0xFF,   0xFF, 0x07, 0xFE, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3C, 0x0F, 0x03, 0xC1,   0xE0, 0xF0, 0x1E, 0x1E,   0x03, 0xC1, 0xE0, 0xFF,
0xFF, 0x01, 0xFF, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3C, 0x0F, 0x03, 0xC1,
0xE0, 0xF0, 0x1E, 0x1E,   0x03, 0xC1, 0xE0, 0xF0,   0x00, 0x00, 0x3F, 0x80,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3C, 0x0F, 0x03, 0xC1,   0xE0, 0xF0, 0x1E, 0x1E,   0x03, 0xC1, 0xE0, 0xF0,
0x00, 0x0C, 0x07, 0x80,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3C, 0x0F, 0x03, 0xC1,
0xE0, 0xF0, 0x1E, 0x1E,   0x07, 0xC1, 0xE0, 0x78,   0x00, 0x3E, 0x07, 0x80,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3C, 0x0F, 0x03, 0xC1,   0xE0, 0xF0, 0x1E, 0x1F,   0x0F, 0xC1, 0xE0, 0x7C,
0x1C, 0x1F, 0x0F, 0x80,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3C, 0x0F, 0x03, 0xC1,
0xE0, 0xF0, 0x1E, 0x0F,   0xFF, 0xC1, 0xFC, 0x3F,   0xFF, 0x1F, 0xFF, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x3C, 0x0F, 0x03, 0xC1,   0xE0, 0xF0, 0x1E, 0x0F,   0xFB, 0xC0, 0xFC, 0x1F,
0xFE, 0x0F, 0xFE, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x3C, 0x0F, 0x03, 0xC1,
0xE0, 0xF0, 0x1E, 0x03,   0xE3, 0xC0, 0x7C, 0x07,   0xF0, 0x03, 0xF8, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
```

```
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00
};


unsigned char BMP_WAITING[9600] = {
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
```

```
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0xFF, 0xF0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x1F, 0xF8, 0x03,
0xFF, 0x00, 0x7F, 0xE0,   0x01, 0xFF, 0xF0, 0x00,   0x0F, 0xFE, 0x0F, 0xFF,   0xFF, 0xFF, 0xF8, 0x3F,
0xF8, 0x07, 0xFE, 0x00,   0x1F, 0xFC, 0x00, 0x07,   0xFF, 0xFE, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x1F, 0xF8, 0x03,   0xFF, 0x80, 0x7F, 0xE0,   0x01, 0xFF, 0xF0, 0x00,
0x0F, 0xFE, 0x0F, 0xFF,   0xFF, 0xFF, 0xF8, 0x3F,   0xF8, 0x07, 0xFF, 0x00,   0x1F, 0xFC, 0x00, 0x1F,
0xFF, 0xFF, 0x80, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x1F, 0xFC, 0x07,
0xFF, 0x80, 0xFF, 0xE0,   0x01, 0xFF, 0xF8, 0x00,   0x0F, 0xFE, 0x0F, 0xFF,   0xFF, 0xFF, 0xF8, 0x3F,
0xF8, 0x07, 0xFF, 0x00,   0x1F, 0xFC, 0x00, 0x7F,   0xFF, 0xFF, 0xC0, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x0F, 0xFC, 0x07,   0xFF, 0x80, 0xFF, 0xC0,   0x03, 0xFF, 0xF8, 0x00,
0x0F, 0xFE, 0x0F, 0xFF,   0xFF, 0xFF, 0xF8, 0x3F,   0xF8, 0x07, 0xFF, 0x80,   0x1F, 0xFC, 0x00, 0xFF,
0xFF, 0xFF, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x0F, 0xFC, 0x07,
0xFF, 0x80, 0xFF, 0xC0,   0x03, 0xFF, 0xF8, 0x00,   0x0F, 0xFE, 0x0F, 0xFF,   0xFF, 0xFF, 0xF8, 0x3F,
0xF8, 0x07, 0xFF, 0xC0,   0x1F, 0xFC, 0x01, 0xFF,   0xFF, 0xFF, 0xF0, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x0F, 0xFC, 0x0F,   0xFF, 0xC0, 0xFF, 0xC0,   0x03, 0xFF, 0xFC, 0x00,
0x0F, 0xFE, 0x0F, 0xFF,   0xFF, 0xFF, 0xF8, 0x3F,   0xF8, 0x07, 0xFF, 0xC0,   0x1F, 0xFC, 0x03, 0xFF,
0xFF, 0xFF, 0xF0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x0F, 0xFC, 0x0F,
0xFF, 0xC0, 0xFF, 0xC0,   0x07, 0xFF, 0xFC, 0x00,   0x0F, 0xFE, 0x0F, 0xFF,   0xFF, 0xFF, 0xF8, 0x3F,
0xF8, 0x07, 0xFF, 0xE0,   0x1F, 0xFC, 0x03, 0xFF,   0xFF, 0xFF, 0xF8, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x07, 0xFE, 0x0F,   0xFF, 0xC1, 0xFF, 0x80,   0x07, 0xFF, 0xFE, 0x00,
0x0F, 0xFE, 0x0F, 0xFF,   0xFF, 0xFF, 0xF8, 0x3F,   0xF8, 0x07, 0xFF, 0xF0,   0x1F, 0xFC, 0x07, 0xFF,
0xE0, 0xFF, 0xF8, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x07, 0xFE, 0x0F,
0xFF, 0xC1, 0xFF, 0x80,   0x0F, 0xFF, 0xFE, 0x00,   0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,
0xF8, 0x07, 0xFF, 0xF0,   0x1F, 0xFC, 0x07, 0xFF,   0x80, 0x3F, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x07, 0xFE, 0x1F,   0xFF, 0xE1, 0xFF, 0x80,   0x0F, 0xFF, 0xFE, 0x00,
0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,   0xF8, 0x07, 0xFF, 0xF8,   0x1F, 0xFC, 0x0F, 0xFF,
0x00, 0x1F, 0xF0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x07, 0xFE, 0x1F,
0xFF, 0xE1, 0xFF, 0x80,   0x0F, 0xFB, 0xFF, 0x00,   0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,
0xF8, 0x07, 0xFF, 0xFC,   0x1F, 0xFC, 0x0F, 0xFE,   0x00, 0x1C, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFE, 0x1F,   0xFF, 0xE1, 0xFF, 0x80,   0x1F, 0xF9, 0xFF, 0x00,
0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,   0xF8, 0x07, 0xFF, 0xFC,   0x1F, 0xFC, 0x0F, 0xFE,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x3F,
0xFF, 0xF1, 0xFF, 0x00,   0x1F, 0xF1, 0xFF, 0x00,   0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,
0xF8, 0x07, 0xFF, 0xFE,   0x1F, 0xFC, 0x0F, 0xFC,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x3F,   0xFF, 0xF3, 0xFF, 0x00,   0x1F, 0xF1, 0xFF, 0x80,
0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,   0xF8, 0x07, 0xFF, 0xFF,   0x1F, 0xFC, 0x1F, 0xFC,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x3F,
0xFF, 0xF3, 0xFF, 0x00,   0x3F, 0xF0, 0xFF, 0x80,   0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,
0xF8, 0x07, 0xFF, 0xFF,   0x9F, 0xFC, 0x1F, 0xFC,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x3F,   0xCF, 0xF3, 0xFF, 0x00,   0x3F, 0xF0, 0xFF, 0xC0,
0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,   0xF8, 0x07, 0xFF, 0xFF,   0x9F, 0xFC, 0x1F, 0xFC,
0x03, 0xFF, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x01, 0xFF, 0x7F,
0xCF, 0xFB, 0xFE, 0x00,   0x7F, 0xE0, 0xFF, 0xC0,   0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,
0xF8, 0x07, 0xFE, 0xFF,   0xDF, 0xFC, 0x1F, 0xFC,   0x03, 0xFF, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,
```

```
0x00, 0x00, 0x00, 0x00,   0x00, 0x01, 0xFF, 0xFF,   0xCF, 0xFF, 0xFE, 0x00,   0x7F, 0xE0, 0xFF, 0xC0,
0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,   0xF8, 0x07, 0xFE, 0xFF,   0xFF, 0xFC, 0x1F, 0xFC,
0x03, 0xFF, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x01, 0xFF, 0xFF,
0x87, 0xFF, 0xFE, 0x00,   0x7F, 0xE0, 0x7F, 0xE0,   0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,
0xF8, 0x07, 0xFE, 0x7F,   0xFF, 0xFC, 0x1F, 0xFC,   0x03, 0xFF, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x01, 0xFF, 0xFF,   0x87, 0xFF, 0xFE, 0x00,   0xFF, 0xC0, 0x7F, 0xE0,
0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,   0xF8, 0x07, 0xFE, 0x3F,   0xFF, 0xFC, 0x1F, 0xFC,
0x03, 0xFF, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xFF, 0xFF,
0x87, 0xFF, 0xFE, 0x00,   0xFF, 0xFF, 0xFF, 0xE0,   0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,
0xF8, 0x07, 0xFE, 0x3F,   0xFF, 0xFC, 0x1F, 0xFC,   0x03, 0xFF, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xFF, 0xFF,   0x87, 0xFF, 0xFC, 0x00,   0xFF, 0xFF, 0xFF, 0xF0,
0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,   0xF8, 0x07, 0xFE, 0x1F,   0xFF, 0xFC, 0x0F, 0xFC,
0x03, 0xFF, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xFF, 0xFF,
0x03, 0xFF, 0xFC, 0x01,   0xFF, 0xFF, 0xFF, 0xF0,   0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,
0xF8, 0x07, 0xFE, 0x0F,   0xFF, 0xFC, 0x0F, 0xFE,   0x03, 0xFF, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xFF, 0xFF,   0x03, 0xFF, 0xFC, 0x01,   0xFF, 0xFF, 0xFF, 0xF0,
0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,   0xF8, 0x07, 0xFE, 0x0F,   0xFF, 0xFC, 0x0F, 0xFE,
0x00, 0x07, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xFF, 0xFF,
0x03, 0xFF, 0xFC, 0x03,   0xFF, 0xFF, 0xFF, 0xF8,   0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,
0xF8, 0x07, 0xFE, 0x07,   0xFF, 0xFC, 0x0F, 0xFF,   0x00, 0x07, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xFE,   0x01, 0xFF, 0xF8, 0x03,   0xFF, 0xFF, 0xFF, 0xF8,
0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,   0xF8, 0x07, 0xFE, 0x03,   0xFF, 0xFC, 0x07, 0xFF,
0x80, 0x07, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xFE,
0x01, 0xFF, 0xF8, 0x03,   0xFF, 0xFF, 0xFF, 0xFC,   0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,
0xF8, 0x07, 0xFE, 0x03,   0xFF, 0xFC, 0x07, 0xFF,   0xE0, 0x3F, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xFE,   0x01, 0xFF, 0xF8, 0x07,   0xFF, 0xFF, 0xFF, 0xFC,
0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,   0xF8, 0x07, 0xFE, 0x01,   0xFF, 0xFC, 0x03, 0xFF,
0xFF, 0xFF, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xFE,
0x01, 0xFF, 0xF8, 0x07,   0xFF, 0x00, 0x0F, 0xFC,   0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,
0xF8, 0x07, 0xFE, 0x00,   0xFF, 0xFC, 0x03, 0xFF,   0xFF, 0xFF, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x3F, 0xFC,   0x00, 0xFF, 0xF8, 0x07,   0xFE, 0x00, 0x0F, 0xFE,
0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,   0xF8, 0x07, 0xFE, 0x00,   0xFF, 0xFC, 0x01, 0xFF,
0xFF, 0xFF, 0xFC, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x3F, 0xFC,
0x00, 0xFF, 0xF0, 0x0F,   0xFE, 0x00, 0x0F, 0xFE,   0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,
0xF8, 0x07, 0xFE, 0x00,   0x7F, 0xFC, 0x00, 0xFF,   0xFF, 0xFF, 0xF0, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x3F, 0xFC,   0x00, 0xFF, 0xF0, 0x0F,   0xFE, 0x00, 0x0F, 0xFE,
0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,   0xF8, 0x07, 0xFE, 0x00,   0x3F, 0xFC, 0x00, 0x7F,
0xFF, 0xFF, 0xE0, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x3F, 0xFC,
0x00, 0x7F, 0xF0, 0x1F,   0xFC, 0x00, 0x07, 0xFF,   0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,
0xF8, 0x07, 0xFE, 0x00,   0x3F, 0xFC, 0x00, 0x1F,   0xFF, 0xFF, 0x80, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x3F, 0xF8,   0x00, 0x7F, 0xF0, 0x1F,   0xFC, 0x00, 0x07, 0xFF,
0x0F, 0xFE, 0x00, 0x01,   0xFF, 0xC0, 0x00, 0x3F,   0xF8, 0x07, 0xFE, 0x00,   0x1F, 0xFC, 0x00, 0x07,
0xFF, 0xFE, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0xFF, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
```

```
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x3F, 0xF0,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0xFF, 0xFF,
0x80, 0x01, 0xFF, 0xFF,   0x00, 0x01, 0xFF, 0xFF,   0xFC, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0xFF, 0xFF,   0x80, 0x07, 0xFF, 0xFF,   0xC0, 0x01, 0xFF, 0xFF,
0xFF, 0x80, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0xFF, 0xFF,
0x80, 0x1F, 0xFF, 0xFF,   0xE0, 0x01, 0xFF, 0xFF,   0xFF, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0xFF, 0xFF,   0x80, 0x3F, 0xFF, 0xFF,   0xF0, 0x01, 0xFF, 0xFF,
0xFF, 0xF0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0xFF, 0xFF,
0x80, 0x7F, 0xFF, 0xFF,   0xF8, 0x01, 0xFF, 0xFF,   0xFF, 0xF0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
```

```
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0xFF, 0xFF,   0x80, 0xFF, 0xFF, 0xFF,   0xFC, 0x01, 0xFF, 0xFF,
0xFF, 0xF8, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0xFF, 0xFF,
0x80, 0xFF, 0xFF, 0xFF,   0xFC, 0x01, 0xFF, 0xFF,   0xFF, 0xF8, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0xFF, 0xFF,   0x81, 0xFF, 0xF0, 0x3F,   0xFE, 0x01, 0xFF, 0xFF,
0xFF, 0xF8, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,
0x01, 0xFF, 0xE0, 0x1F,   0xFE, 0x01, 0xFF, 0x80,   0x7F, 0xF8, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,   0x03, 0xFF, 0xC0, 0x0F,   0xFF, 0x01, 0xFF, 0x80,
0x3F, 0xFC, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,
0x03, 0xFF, 0x80, 0x07,   0xFF, 0x01, 0xFF, 0x80,   0x1F, 0xF8, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,   0x03, 0xFF, 0x80, 0x07,   0xFF, 0x01, 0xFF, 0x80,
0x3F, 0xF8, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0xFF, 0xFE,
0x07, 0xFF, 0x00, 0x03,   0xFF, 0x81, 0xFF, 0x80,   0x7F, 0xF8, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0xFF, 0xFE,   0x07, 0xFF, 0x00, 0x03,   0xFF, 0x81, 0xFF, 0xFF,
0xFF, 0xF8, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0xFF, 0xFE,
0x07, 0xFF, 0x00, 0x03,   0xFF, 0x81, 0xFF, 0xFF,   0xFF, 0xF0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0xFF, 0xFE,   0x07, 0xFF, 0x00, 0x03,   0xFF, 0x81, 0xFF, 0xFF,
0xFF, 0xF0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0xFF, 0xFE,
0x07, 0xFF, 0x00, 0x03,   0xFF, 0x81, 0xFF, 0xFF,   0xFF, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0xFF, 0xFE,   0x07, 0xFF, 0x00, 0x03,   0xFF, 0x81, 0xFF, 0xFF,
0xFF, 0xC0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0xFF, 0xFE,
0x07, 0xFF, 0x00, 0x03,   0xFF, 0x81, 0xFF, 0xFF,   0xFF, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0xFF, 0xFE,   0x07, 0xFF, 0x00, 0x03,   0xFF, 0x81, 0xFF, 0xFF,
0xF8, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,
0x07, 0xFF, 0x00, 0x03,   0xFF, 0x81, 0xFF, 0xFF,   0xFF, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,   0x07, 0xFF, 0x00, 0x03,   0xFF, 0x81, 0xFF, 0x8F,
0xFF, 0x80, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,
0x03, 0xFF, 0x80, 0x07,   0xFF, 0x01, 0xFF, 0x87,   0xFF, 0xC0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,   0x03, 0xFF, 0x80, 0x07,   0xFF, 0x01, 0xFF, 0x83,
0xFF, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,
0x03, 0xFF, 0xC0, 0x0F,   0xFF, 0x01, 0xFF, 0x81,   0xFF, 0xE0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,   0x01, 0xFF, 0xE0, 0x1F,   0xFE, 0x01, 0xFF, 0x81,
0xFF, 0xF0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,
0x01, 0xFF, 0xF0, 0x3F,   0xFE, 0x01, 0xFF, 0x80,   0xFF, 0xF0, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,   0x00, 0xFF, 0xFF, 0xFF,   0xFC, 0x01, 0xFF, 0x80,
0xFF, 0xF8, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,
0x00, 0xFF, 0xFF, 0xFF,   0xFC, 0x01, 0xFF, 0x80,   0x7F, 0xF8, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,   0x00, 0x7F, 0xFF, 0xFF,   0xF8, 0x01, 0xFF, 0x80,
0x7F, 0xFC, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,
0x00, 0x3F, 0xFF, 0xFF,   0xF0, 0x01, 0xFF, 0x80,   0x3F, 0xFC, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,   0x00, 0x1F, 0xFF, 0xFF,   0xE0, 0x01, 0xFF, 0x80,
0x1F, 0xFE, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,
0x00, 0x07, 0xFF, 0xFF,   0xC0, 0x01, 0xFF, 0x80,   0x1F, 0xFE, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x03, 0xFF, 0x80, 0x00,   0x00, 0x03, 0xFF, 0xFF,   0x00, 0x01, 0xFF, 0x80,
0x0F, 0xFF, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x3F, 0xF8,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
```

```
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
```

```
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,
0x07, 0xFF, 0x03, 0xFF,   0xFF, 0xF8, 0x00, 0x3F,   0xFF, 0xFE, 0x00, 0x00,   0x01, 0xFF, 0xF0, 0x00,
0x7F, 0xFF, 0xFF, 0xFF,   0xC3, 0xFF, 0xFF, 0xFF,   0xE0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,   0x07, 0xFF, 0x03, 0xFF,   0xFF, 0xFE, 0x00, 0x3F,
0xFF, 0xFF, 0x80, 0x00,   0x01, 0xFF, 0xF0, 0x00,   0x7F, 0xFF, 0xFF, 0xFF,   0xC3, 0xFF, 0xFF, 0xFF,
0xE0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,
0x07, 0xFF, 0x03, 0xFF,   0xFF, 0xFF, 0x80, 0x3F,   0xFF, 0xFF, 0xE0, 0x00,   0x01, 0xFF, 0xF8, 0x00,
0x7F, 0xFF, 0xFF, 0xFF,   0xC3, 0xFF, 0xFF, 0xFF,   0xE0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,   0x07, 0xFF, 0x03, 0xFF,   0xFF, 0xFF, 0xC0, 0x3F,
0xFF, 0xFF, 0xF0, 0x00,   0x03, 0xFF, 0xF8, 0x00,   0x7F, 0xFF, 0xFF, 0xFF,   0xC3, 0xFF, 0xFF, 0xFF,
0xE0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,
0x07, 0xFF, 0x03, 0xFF,   0xFF, 0xFF, 0xC0, 0x3F,   0xFF, 0xFF, 0xF8, 0x00,   0x03, 0xFF, 0xF8, 0x00,
0x7F, 0xFF, 0xFF, 0xFF,   0xC3, 0xFF, 0xFF, 0xFF,   0xE0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,   0x07, 0xFF, 0x03, 0xFF,   0xFF, 0xFF, 0xE0, 0x3F,
0xFF, 0xFF, 0xFC, 0x00,   0x03, 0xFF, 0xFC, 0x00,   0x7F, 0xFF, 0xFF, 0xFF,   0xC3, 0xFF, 0xFF, 0xFF,
0xE0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,
0x07, 0xFF, 0x03, 0xFF,   0xFF, 0xFF, 0xE0, 0x3F,   0xFF, 0xFF, 0xFE, 0x00,   0x07, 0xFF, 0xFC, 0x00,
0x7F, 0xFF, 0xFF, 0xFF,   0xC3, 0xFF, 0xFF, 0xFF,   0xE0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,   0x07, 0xFF, 0x03, 0xFF,   0xFF, 0xFF, 0xF0, 0x3F,
0xFF, 0xFF, 0xFE, 0x00,   0x07, 0xFF, 0xFE, 0x00,   0x7F, 0xFF, 0xFF, 0xFF,   0xC3, 0xFF, 0xFF, 0xFF,
0xE0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,
0x07, 0xFF, 0x03, 0xFF,   0x80, 0xFF, 0xF0, 0x3F,   0xF0, 0x3F, 0xFF, 0x00,   0x0F, 0xFF, 0xFE, 0x00,
0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0x80, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,   0x07, 0xFF, 0x03, 0xFF,   0x80, 0x7F, 0xF0, 0x3F,
0xF0, 0x0F, 0xFF, 0x00,   0x0F, 0xFF, 0xFE, 0x00,   0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0x80, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,
0x07, 0xFF, 0x03, 0xFF,   0x80, 0x7F, 0xF0, 0x3F,   0xF0, 0x07, 0xFF, 0x00,   0x0F, 0xFB, 0xFF, 0x00,
0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0x80, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,   0x07, 0xFF, 0x03, 0xFF,   0x80, 0x7F, 0xF0, 0x3F,
0xF0, 0x03, 0xFF, 0x80,   0x1F, 0xF9, 0xFF, 0x00,   0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0x80, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,
0x07, 0xFF, 0x03, 0xFF,   0x80, 0x7F, 0xF0, 0x3F,   0xF0, 0x03, 0xFF, 0x80,   0x1F, 0xF1, 0xFF, 0x00,
0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0xFF, 0xFF,   0xC0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,   0x07, 0xFF, 0x03, 0xFF,   0x80, 0x7F, 0xF0, 0x3F,
0xF0, 0x03, 0xFF, 0x80,   0x1F, 0xF1, 0xFF, 0x80,   0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0xFF, 0xFF,
0xC0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,
0x07, 0xFF, 0x03, 0xFF,   0x81, 0xFF, 0xF0, 0x3F,   0xF0, 0x03, 0xFF, 0x80,   0x3F, 0xF0, 0xFF, 0x80,
0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0xFF, 0xFF,   0xC0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,   0x07, 0xFF, 0x03, 0xFF,   0xFF, 0xFF, 0xE0, 0x3F,
0xF0, 0x03, 0xFF, 0x80,   0x3F, 0xF0, 0xFF, 0xC0,   0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0xFF, 0xFF,
0xC0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,
0x07, 0xFF, 0x03, 0xFF,   0xFF, 0xFF, 0xE0, 0x3F,   0xF0, 0x03, 0xFF, 0x80,   0x7F, 0xE0, 0xFF, 0xC0,
0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0xFF, 0xFF,   0xC0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,   0x07, 0xFF, 0x03, 0xFF,   0xFF, 0xFF, 0xE0, 0x3F,
0xF0, 0x03, 0xFF, 0x80,   0x7F, 0xE0, 0xFF, 0xC0,   0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0xFF, 0xFF,
0xC0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,
0x07, 0xFF, 0x03, 0xFF,   0xFF, 0xFF, 0xC0, 0x3F,   0xF0, 0x03, 0xFF, 0x80,   0x7F, 0xE0, 0x7F, 0xE0,
0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0xFF, 0xFF,   0xC0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,   0x07, 0xFF, 0x03, 0xFF,   0xFF, 0xFF, 0x80, 0x3F,
0xF0, 0x03, 0xFF, 0x80,   0xFF, 0xC0, 0x7F, 0xE0,   0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0xFF, 0xFF,
0xC0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x03, 0xFF, 0x80,
0x07, 0xFF, 0x03, 0xFF,   0xFF, 0xFF, 0x00, 0x3F,   0xF0, 0x03, 0xFF, 0x80,   0xFF, 0xFF, 0xFF, 0xE0,
0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0x80, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x01, 0xFF, 0x80,   0x07, 0xFF, 0x03, 0xFF,   0xFF, 0xFE, 0x00, 0x3F,
0xF0, 0x03, 0xFF, 0x80,   0xFF, 0xFF, 0xFF, 0xF0,   0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0x80, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x01, 0xFF, 0x80,
0x07, 0xFE, 0x03, 0xFF,   0xFF, 0xF0, 0x00, 0x3F,   0xF0, 0x03, 0xFF, 0x81,   0xFF, 0xFF, 0xFF, 0xF0,
0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0x80, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x01, 0xFF, 0xC0,   0x0F, 0xFE, 0x03, 0xFF,   0x80, 0x00, 0x00, 0x3F,
0xF0, 0x07, 0xFF, 0x01,   0xFF, 0xFF, 0xFF, 0xF0,   0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0x80, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x01, 0xFF, 0xC0,
0x0F, 0xFE, 0x03, 0xFF,   0x80, 0x00, 0x00, 0x3F,   0xF0, 0x0F, 0xFF, 0x03,   0xFF, 0xFF, 0xFF, 0xF8,
0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0x80, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
```

```
0x00, 0x00, 0x00, 0x00,   0x00, 0x01, 0xFF, 0xE0,   0x1F, 0xFE, 0x03, 0xFF,   0x80, 0x00, 0x00, 0x3F,
0xF0, 0x1F, 0xFF, 0x03,   0xFF, 0xFF, 0xFF, 0xF8,   0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0x80, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xFF, 0xF0,
0x3F, 0xFE, 0x03, 0xFF,   0x80, 0x00, 0x00, 0x3F,   0xFF, 0xFF, 0xFE, 0x03,   0xFF, 0xFF, 0xFF, 0xFC,
0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0xFF, 0xFF,   0xF0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xFF, 0xFF,   0xFF, 0xFC, 0x03, 0xFF,   0x80, 0x00, 0x00, 0x3F,
0xFF, 0xFF, 0xFE, 0x07,   0xFF, 0xFF, 0xFF, 0xFC,   0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0xFF, 0xFF,
0xF0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0xFF, 0xFF,
0xFF, 0xFC, 0x03, 0xFF,   0x80, 0x00, 0x00, 0x3F,   0xFF, 0xFF, 0xFC, 0x07,   0xFF, 0x00, 0x0F, 0xFC,
0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0xFF, 0xFF,   0xF0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x7F, 0xFF,   0xFF, 0xF8, 0x03, 0xFF,   0x80, 0x00, 0x00, 0x3F,
0xFF, 0xFF, 0xF8, 0x07,   0xFE, 0x00, 0x0F, 0xFE,   0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0xFF, 0xFF,
0xF0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x3F, 0xFF,
0xFF, 0xF0, 0x03, 0xFF,   0x80, 0x00, 0x00, 0x3F,   0xFF, 0xFF, 0xF0, 0x0F,   0xFE, 0x00, 0x0F, 0xFE,
0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0xFF, 0xFF,   0xF0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x1F, 0xFF,   0xFF, 0xE0, 0x03, 0xFF,   0x80, 0x00, 0x00, 0x3F,
0xFF, 0xFF, 0xE0, 0x0F,   0xFE, 0x00, 0x0F, 0xFE,   0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0xFF, 0xFF,
0xF0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x0F, 0xFF,
0xFF, 0xC0, 0x03, 0xFF,   0x80, 0x00, 0x00, 0x3F,   0xFF, 0xFF, 0x80, 0x1F,   0xFC, 0x00, 0x07, 0xFF,
0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0xFF, 0xFF,   0xF0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x03, 0xFF,   0xFF, 0x00, 0x03, 0xFF,   0x80, 0x00, 0x00, 0x3F,
0xFF, 0xFC, 0x00, 0x1F,   0xFC, 0x00, 0x07, 0xFF,   0x00, 0x0F, 0xFE, 0x00,   0x03, 0xFF, 0xFF, 0xFF,
0xF0, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x7F,
0xF8, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
```

```
   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,
   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00
};

//Character tiles 0-9
unsigned char BMP_CHAR[10][180] = {
   //Char '0'
   {0x01, 0xFF, 0xFF, 0x80,   0x01, 0xFF, 0xFF, 0x80,   0x01, 0xFF, 0xFF, 0x80,   0x01, 0xFF, 0xFF, 0x80,
   0x01, 0xFF, 0xFF, 0x80,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,
   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,
   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,
   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,
   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,
   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,
   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,
   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,
   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,   0x3E, 0x00, 0x00, 0x7C,
   0x01, 0xFF, 0xFF, 0x80,   0x01, 0xFF, 0xFF, 0x80,   0x01, 0xFF, 0xFF, 0x80,   0x01, 0xFF, 0xFF, 0x80,
   0x01, 0xFF, 0xFF, 0x80},
   //Char '1'
   {0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,
   0x00, 0x01, 0xF0, 0x00,   0x03, 0xFF, 0xF0, 0x00,   0x03, 0xFF, 0xF0, 0x00,   0x03, 0xFF, 0xF0, 0x00,
   0x03, 0xFF, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,
   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,
   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,
   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,
   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,
   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,
   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,
   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,
   0x00, 0x01, 0xF0, 0x00,   0x03, 0xFF, 0xFF, 0xF8,   0x03, 0xFF, 0xFF, 0xF8,   0x03, 0xFF, 0xFF, 0xF8,
   0x03, 0xFF, 0xFF, 0xF8},
   //Char '2'
   {0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,
   0x03, 0xFF, 0xFF, 0x00,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
   0x3C, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,
   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,
   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,
   0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,   0x00, 0x01, 0xF0, 0x00,
   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x1E, 0x00, 0x00,
   0x00, 0x1E, 0x00, 0x00,   0x00, 0x1E, 0x00, 0x00,   0x00, 0x1E, 0x00, 0x00,   0x00, 0x1E, 0x00, 0x00,
   0x03, 0xE0, 0x00, 0x00,   0x03, 0xE0, 0x00, 0x00,   0x03, 0xE0, 0x00, 0x00,   0x03, 0xE0, 0x00, 0x00,
   0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,
   0x3C, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0xF8,   0x3F, 0xFF, 0xFF, 0xF8,   0x3F, 0xFF, 0xFF, 0xF8,
   0x3F, 0xFF, 0xFF, 0xF8 },
   //Char '3'
   {0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,
   0x03, 0xFF, 0xFF, 0x00,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
   0x3C, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,
   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,
   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x1F, 0xFF, 0x00,   0x00, 0x1F, 0xFF, 0x00,
   0x00, 0x1F, 0xFF, 0x00,   0x00, 0x1F, 0xFF, 0x00,   0x00, 0x1F, 0xFF, 0x00,   0x00, 0x00, 0x00, 0xF8,
   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,
   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,
   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,
   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
   0x3C, 0x00, 0x00, 0xF8,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,
   0x03, 0xFF, 0xFF, 0x00 },
   //Char '4'
   {0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,
   0x00, 0x00, 0x0F, 0x00,   0x00, 0x01, 0xFF, 0x00,   0x00, 0x01, 0xFF, 0x00,   0x00, 0x01, 0xFF, 0x00,
   0x00, 0x01, 0xFF, 0x00,   0x00, 0x1E, 0x0F, 0x00,   0x00, 0x1E, 0x0F, 0x00,   0x00, 0x1E, 0x0F, 0x00,
   0x00, 0x1E, 0x0F, 0x00,   0x00, 0x1E, 0x0F, 0x00,   0x00, 0x1E, 0x0F, 0x00,   0x00, 0x1E, 0x0F, 0x00,
   0x00, 0x1E, 0x0F, 0x00,   0x00, 0x1E, 0x0F, 0x00,   0x03, 0xE0, 0x0F, 0x00,   0x03, 0xE0, 0x0F, 0x00,
   0x03, 0xE0, 0x0F, 0x00,   0x03, 0xE0, 0x0F, 0x00,   0x03, 0xE0, 0x0F, 0x00,   0x3C, 0x00, 0x0F, 0x00,
   0x3C, 0x00, 0x0F, 0x00,   0x3C, 0x00, 0x0F, 0x00,   0x3C, 0x00, 0x0F, 0x00,   0x3F, 0xFF, 0xFF, 0xF8,
   0x3F, 0xFF, 0xFF, 0xF8,   0x3F, 0xFF, 0xFF, 0xF8,   0x3F, 0xFF, 0xFF, 0xF8,   0x3F, 0xFF, 0xFF, 0xF8,
   0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,
   0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,
   0x00, 0x00, 0x0F, 0x00,   0x00, 0x01, 0xFF, 0xF8,   0x00, 0x01, 0xFF, 0xF8,   0x00, 0x01, 0xFF, 0xF8,
   0x00, 0x01, 0xFF, 0xF8 },
   //Char '5'
   {0x3F, 0xFF, 0xFF, 0xF8,   0x3F, 0xFF, 0xFF, 0xF8,   0x3F, 0xFF, 0xFF, 0xF8,   0x3F, 0xFF, 0xFF, 0xF8,
   0x3F, 0xFF, 0xFF, 0xF8,   0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,
   0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,
   0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,
   0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,   0x3F, 0xFF, 0xFF, 0x00,   0x3F, 0xFF, 0xFF, 0x00,
   0x3F, 0xFF, 0xFF, 0x00,   0x3F, 0xFF, 0xFF, 0x00,   0x3F, 0xFF, 0xFF, 0x00,   0x00, 0x00, 0x00, 0xF8,
   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,
```

```
    0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,
    0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,
    0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
    0x3C, 0x00, 0x00, 0xF8,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,
    0x03, 0xFF, 0xFF, 0x00 },
    //Char '6'
    {0x00, 0x1F, 0xFF, 0x00,   0x00, 0x1F, 0xFF, 0x00,   0x00, 0x1F, 0xFF, 0x00,   0x00, 0x1F, 0xFF, 0x00,
    0x00, 0x1F, 0xFF, 0x00,   0x03, 0xE0, 0x00, 0x00,   0x03, 0xE0, 0x00, 0x00,   0x03, 0xE0, 0x00, 0x00,
    0x03, 0xE0, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,
    0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,
    0x3C, 0x00, 0x00, 0x00,   0x3C, 0x00, 0x00, 0x00,   0x3C, 0x1F, 0xFF, 0x00,   0x3C, 0x1F, 0xFF, 0x00,
    0x3C, 0x1F, 0xFF, 0x00,   0x3C, 0x1F, 0xFF, 0x00,   0x3C, 0x1F, 0xFF, 0x00,   0x3F, 0xE0, 0x00, 0xF8,
    0x3F, 0xE0, 0x00, 0xF8,   0x3F, 0xE0, 0x00, 0xF8,   0x3F, 0xE0, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
    0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
    0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
    0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
    0x3C, 0x00, 0x00, 0xF8,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,
    0x03, 0xFF, 0xFF, 0x00 },
    //Char '7'
    {0x3F, 0xFF, 0xFF, 0xF8,   0x3F, 0xFF, 0xFF, 0xF8,   0x3F, 0xFF, 0xFF, 0xF8,   0x3F, 0xFF, 0xFF, 0xF8,
    0x3F, 0xFF, 0xFF, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
    0x3C, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,
    0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,
    0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,
    0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,
    0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x01, 0xF0, 0x00,   0x00, 0x1E, 0x00, 0x00,
    0x00, 0x1E, 0x00, 0x00,   0x00, 0x1E, 0x00, 0x00,   0x00, 0x1E, 0x00, 0x00,   0x00, 0x1E, 0x00, 0x00,
    0x00, 0x1E, 0x00, 0x00,   0x00, 0x1E, 0x00, 0x00,   0x00, 0x1E, 0x00, 0x00,   0x00, 0x1E, 0x00, 0x00,
    0x03, 0xE0, 0x00, 0x00,   0x03, 0xE0, 0x00, 0x00,   0x03, 0xE0, 0x00, 0x00,   0x03, 0xE0, 0x00, 0x00,
    0x03, 0xE0, 0x00, 0x00,   0x03, 0xE0, 0x00, 0x00,   0x03, 0xE0, 0x00, 0x00,   0x03, 0xE0, 0x00, 0x00,
    0x03, 0xE0, 0x00, 0x00},
    //Char '8'
    {0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,
    0x03, 0xFF, 0xFF, 0x00,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
    0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
    0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
    0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,
    0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,   0x3C, 0x00, 0x00, 0xF8,
    0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
    0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
    0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
    0x3C, 0x00, 0x00, 0xF8,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,
    0x03, 0xFF, 0xFF, 0x00 },
    //Char '9'
    {0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,   0x03, 0xFF, 0xFF, 0x00,
    0x03, 0xFF, 0xFF, 0x00,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
    0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
    0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,
    0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x00, 0xF8,   0x3C, 0x00, 0x0F, 0xF8,   0x3C, 0x00, 0x0F, 0xF8,
    0x3C, 0x00, 0x0F, 0xF8,   0x3C, 0x00, 0x0F, 0xF8,   0x3C, 0x00, 0x0F, 0xF8,   0x03, 0xFF, 0xF0, 0xF8,
    0x03, 0xFF, 0xF0, 0xF8,   0x03, 0xFF, 0xF0, 0xF8,   0x03, 0xFF, 0xF0, 0xF8,   0x00, 0x00, 0x00, 0xF8,
    0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,
    0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,   0x00, 0x00, 0x00, 0xF8,
    0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,   0x00, 0x00, 0x0F, 0x00,
    0x00, 0x00, 0x0F, 0x00,   0x03, 0xFF, 0xF0, 0x00,   0x03, 0xFF, 0xF0, 0x00,   0x03, 0xFF, 0xF0, 0x00,
    0x03, 0xFF, 0xF0, 0x00}
};

#endif
```

Comm.h

```
#include "comm.h"
#include "xuartlite_l.h"
#include "xparameters.h"

#include "routes.h"
#include "signConfig.h"

#define BUFFER_SIZE 50
#define PACKET_ERROR -999

int readUntil(unsigned char *buffer, int *cursor, int size, unsigned char x);
unsigned int nextByte();
int hasNext();

unsigned char buffer[BUFFER_SIZE];

unsigned int nextByte(){
   return XUartLite_RecvByte(XPAR_UARTLITE_0_BASEADDR);
}

int hasNext(){
   return !XUartLite_mIsReceiveEmpty  (XPAR_UARTLITE_0_BASEADDR);
}

int processesPacket(){

   int dest;
   int cmd;
   int seq;
   int agency;
   int rt;
   int trip;
   int stp;
   int time;
   int chksum;

   int bytes = 0;
   int cursor = 0;
   int status = 0;
   unsigned char last;

   for(cursor = 0; cursor < BUFFER_SIZE; cursor++){
      buffer[cursor] = '\0';
   }
   cursor = 0;

   while((last = nextByte()) != '!');   //Go to start of packet
   while((last = nextByte()) != '!'){   //Until end of packet
      buffer[bytes++] = last;
   }

   cursor++;
   if(buffer[cursor] == '4'){
//      xil_printf("Page type 4\n");
```

```
   }
   status = readUntil(buffer, &cursor, bytes, '+');
   dest = status;
//      xil_printf("Dest: %d\n", dest);

   cmd = (buffer[cursor]-'0')*10+buffer[cursor+1]-'0';
   cursor+=2;
//      xil_printf("Cmd: %d\n", cmd);

   status = readUntil(buffer, &cursor, bytes, ';');
   seq = status;
//      xil_printf("Seq: %d\n", seq);

   status = readUntil(buffer, &cursor, bytes, '&');
   agency = status;
//      xil_printf("Agency: %d\n", agency);

   status = readUntil(buffer, &cursor, bytes, '%');
   rt = status;
//      xil_printf("Route: %d\n", rt);

   status = readUntil(buffer, &cursor, bytes, '#');
   trip = status%10;
//      xil_printf("Trip: %d\n", trip);

   status = readUntil(buffer, &cursor, bytes, '/');
   stp = status;
//      xil_printf("Stop: %d\n", stp);

   status = readUntil(buffer, &cursor, bytes, '~');
   time = status;
//      xil_printf("Time: %d\n", time);

   if(!(status == PACKET_ERROR)){
//      xil_printf("Packet Intact\n");
      int i,j;

      const unsigned char *thisTrip;
      //Time of the stop mentioned in update
      int stop_time_packet;
      //Time of the stop the sign resides at
      int stop_time_this;
      //Number of stops in this trip
      int numStops;
      //Total time of entire trip cycle
      int routeOverhead;

      for(i=0; i<num_monitored; i++){      //Fore every route
                                                  monitored
         if(rt == monitored[i][0]){      //If this route is
                                                  monitored
            //If this trip is monitored
            if(trip == monitored[i][1]){
               thisTrip = routes[rt].tripTimes[trip];
               stop_time_packet = thisTrip[stp];
               numStops = thisTrip[0];
               routeOverhead = thisTrip[numStops+1];
```

```
                //Set this stop's time
                stop_time_this = thisTrip[monitored[i][2]];

                //If the tripTime value of the packet
                                received
                //is for a stop that comes after this one
                if(stop_time_packet >= stop_time_this){
                   routes[rt].route_eta[trip] =
                                           stop_time_this+
                                           routeOverhead -
                         stop_time_packet;
                }
                //If he tripTime of the packet comes before
                //the monitored stop
                else{
                   routes[rt].route_eta[trip] =
                     stop_time_this-stop_time_packet;
                }


                if(stp == monitored[i][2]){   //If it's this
                                        stop on some route
                   routes[rt].avgLateness[trip] += time;
//Add new delay
                   //If previous value not 0
                   if (routes[rt].avgLateness[trip] != time){
                      //Average new with old
                      routes[rt].avgLateness[trip] /= 2;
                   }
                }
                //If is stop on the route, but not this one
                //and the previous lateness isn't zero
                else if (routes[rt].avgLateness[trip] != 0){
                   //Add avg lateness for trip to eta
                   routes[rt].route_eta[trip] +=
                     routes[rt].avgLateness[trip];
                }
                //If no lateness average gathered
                else{
                   //Eta is directly adujsted by packet
                   routes[rt].route_eta[trip] += time;
                }
                if(routes[rt].route_eta[trip] < 0){
                   //If the bus should have already been here
                   //ETA is 0
                   routes[rt].route_eta[trip] = 0;
                }
             }
          }
       }
       return 0;
    }
    else{
       return 1;
    }
}
```

```
/**
 * Converts a string of chars with a termination character x
 * to an integer.
 */
int readUntil(unsigned char *buffer, int *cursor, int size, unsigned char x){
   unsigned int result = 0;
   int i;
   int sign = 1;
   for(i=*cursor; i<size; (*cursor)++){
      if(buffer[*cursor] != x){
         if(buffer[*cursor] == '-'){
            sign = -1;
         }
         else{
            result = result*10+(buffer[*cursor]-'0');
         }
      }
      else{
         *cursor+=1;
         return sign*result;
      }
   }
   return PACKET_ERROR;
}
```

Graphics.h

```
#ifndef GRAPHICS_H_
#define GRAPHICS_H_

unsigned char screen[9600];
unsigned char eta;
unsigned char route;

void dispWaiting();
void dispMain();
void repaint();

#endif /*GRAPHICS_H_*/
```

Graphics.c

```
#include <string.h>

#include "bitmaps.h"
#include "graphics.h"
#include "spi.h"

#define ETA_X 120  //Multiple of 8
#define ETA_Y 122
#define ETA_MAX_WIDTH 200 //Multiple of 8

#define ROUTE_X 0  //Multiple of 8
#define ROUTE_Y 122
#define ROUTE_MAX_WIDTH 112

#define CHAR_HEIGHT 45
#define CHAR_WIDTH 32  //Multiple of 8

unsigned int digitsInUInt(unsigned int);
unsigned int nthDigit(unsigned int, unsigned int);

unsigned int digitsInUInt(unsigned int x){
      int digits = 0;
      do{
            digits ++;
            x/= 10;
      }while(x>0);

      return digits;
}

unsigned int nthDigit(unsigned int number, unsigned int n){
      int i;
      for(i=0;i<n;i++){
            number/=10;
      }
      return number%10;
}
```

```
void dispWaiting(){
	int i;
	for(i=0; i<9600; i++){
		screen[i] = BMP_WAITING[i];
	}
}

void dispMain(){
	int i;
	//Load backgrop
	for(i=0; i<9600; i++){
		screen[i] = BMP_BG[i];
	}
	//Draw ETA
	int row, digit, byte;
	unsigned int num_digits = digitsInUInt(eta);
	int start_x = ETA_X+(ETA_MAX_WIDTH-CHAR_WIDTH*num_digits)/2;
	for(row = 0; row < CHAR_HEIGHT; row++){
		for(digit=0; digit<num_digits;digit++){
			for(byte = 0; byte < CHAR_WIDTH/8; byte++){

	screen[start_x/8+40*(ETA_Y+row)+digit*(CHAR_WIDTH/8)+byte] =
					BMP_CHAR[nthDigit(eta, num_digits-digit-1)]
					  [(CHAR_WIDTH/8)*row+byte];
			}
		}
	}

	//Draw Route
	num_digits = digitsInUInt(route);
	start_x = ROUTE_X+(ROUTE_MAX_WIDTH-CHAR_WIDTH*num_digits)/2;
	for(row = 0; row < CHAR_HEIGHT; row++){
		for(digit=0; digit<num_digits;digit++){
			for(byte = 0; byte < CHAR_WIDTH/8; byte++){

	screen[start_x/8+40*(ETA_Y+row)+digit*(CHAR_WIDTH/8)+byte] =
					BMP_CHAR[nthDigit(route, num_digits-digit-
1)][(CHAR_WIDTH/8)*row+byte];
			}
		}
	}
}

void repaint(){

	//The below statement will dump binary to terminal
	write(0, screen, 9600);

	//The below statement will dump plaintext to terminal
	//xil_printf("Route: %d - ETA: %d\n", route, eta);

	/**
	 * Uncomment below block and comment the above to use the physical
ChLCD
	 *

	//Load memory
```

```
	XIo_Out32(XPAR_SPI_INTERFACE_BASEADDR+0x60 ,0x196);
	spi_slave_select(0x00);                    // Start transaction
	XIo_Out32(XPAR_SPI_INTERFACE_BASEADDR+0x60 ,0x96);

	while(spi_status()&0x08);
	spi_data(0x00);                                 // Put Data in Tx

	while(spi_status()&0x08);
	spi_data(0x00);                                 //Starting at 0

	while(spi_status()&0x08);
	spi_data(0x00);                                 //Starting at 0

	int i;
	for (i=0;i<9600;i++){
		while(spi_status()&0x08);
		spi_data(screen[i]);
	}

	while(!(spi_status()&0x04));
	spi_slave_select(0xFF);                    // End transaction


	//Draw screen
	XIo_Out32(XPAR_SPI_INTERFACE_BASEADDR+0x60 ,0x196);
	spi_slave_select(0x00);                    // Start transaction
	XIo_Out32(XPAR_SPI_INTERFACE_BASEADDR+0x60 ,0x96);

	while(spi_status()&0x08);
	spi_data(0x19);                                 // Draw Full command

	while(spi_status()&0x08);
	spi_data(0x00);

	while(spi_status()&0x08);
	spi_data(0x00);

	while(!(spi_status()&0x04));
	spi_slave_select(0xFF);                    // End transaction

	*/
}
```

Main.c

```
#include "xparameters.h"
#include "xio.h"

#include "signConfig.h"
#include "graphics.h"
#include "timer.h"
#include "comm.h"
#include "routes.h"
#include "xuartlite_l.h"
#include "xtmrctr.h"

#define SECONDS_BETWEEN_SCREENS           3
#define SECONDS_IN_A_MINUTE               20

//extern unsigned char monitored[2][3];
//extern int num_monitored;

int main(){
      XUartLite_mDisableIntr  (XPAR_UARTLITE_0_BASEADDR);
    XStatus status;
    status = IntrSetup(&InterruptController,
                              &TimerCounterInst,
                              XPAR_XPS_TIMER_0_DEVICE_ID,
                              XPAR_XPS_INTC_0_XPS_TIMER_0_INTERRUPT_INTR,
                              0);
      eta=0;

      initRoutes();
      dispWaiting();
      repaint();

      int displayed = -1;
      int lastDisplayed = -1;
      while(1){
            if(!XUartLite_mIsReceiveEmpty  (XPAR_UARTLITE_0_BASEADDR)){
                  processesPacket();
            }

            if (TimerExpired != LastTimerExpired){
            if(TimerExpiredMin >= SECONDS_IN_A_MINUTE){
                  int i,p;
                  //For every monitored route
                  for(i=0; i<num_monitored; i++){
                        int dispRoute = monitored[i][0];
                        //For every trip in a monitored route
                        for(p=0;p<MAX_TRIPS_PER_ROUTE;p++){
                              //Tick ETAs by 1
                              if(routes[dispRoute].route_eta[p]>0){
                                    routes[dispRoute].route_eta[p]--;
                              }
                        }
                  }
                  TimerExpiredMin = 0;
            }
```

```
                  if(TimerExpired >= SECONDS_BETWEEN_SCREENS){
                    int min_eta = 0;
                    int dispRoute;
                    int eta_sum = 0;
                    int k;
                    //For each monitored route
                    eta_sum = 0;
                          while(displayed < num_monitored-1){
                          displayed++;
                          dispRoute = monitored[displayed][0];
                          //For every trip in a monitored route
                          for(k=0;k<MAX_TRIPS_PER_ROUTE;k++){
                                //Find the minimum ETA
                                if(min_eta == 0 ||

      (min_eta>routes[dispRoute].route_eta[k] &&

      routes[dispRoute].route_eta[k] != 0)){
                                      min_eta = routes[dispRoute].route_eta[k];
                                }
//xil_printf("%d, %d eta: %d\n", monitored[displayed][0], k,
routes[dispRoute].route_eta[k]);
                          }
                          eta_sum += min_eta;
//xil_printf("min_eta: %d\n", min_eta);
                          if(min_eta>0 && ((lastDisplayed != displayed) ||
                                (min_eta != eta))){
                                //Display eta
                                route = dispRoute;
                                eta = min_eta;
                                dispMain();
                                lastDisplayed = displayed;
                                break;
                          }
                          }
                          if(eta_sum == 0){
                                dispWaiting();
//xil_printf("Waiting for update\n");
                         }
                          if(displayed==num_monitored-1){
                                displayed = -1;
                    }

                          repaint();
                          TimerExpired = 0;
              }
              }
      }

      return 0;
}
```

Routes.h

```
#include "xparameters.h"
#include "xio.h"

#include "signConfig.h"
#include "graphics.h"
#include "timer.h"
#include "comm.h"
#include "routes.h"
#include "xuartlite_l.h"
#include "xtmrctr.h"

#define SECONDS_BETWEEN_SCREENS           3
#define SECONDS_IN_A_MINUTE               20

//extern unsigned char monitored[2][3];
//extern int num_monitored;

int main(){
      XUartLite_mDisableIntr  (XPAR_UARTLITE_0_BASEADDR);
    XStatus status;
    status = IntrSetup(&InterruptController,
                              &TimerCounterInst,
                              XPAR_XPS_TIMER_0_DEVICE_ID,
                              XPAR_XPS_INTC_0_XPS_TIMER_0_INTERRUPT_INTR,
                              0);
      eta=0;

      initRoutes();
      dispWaiting();
      repaint();

      int displayed = -1;
      int lastDisplayed = -1;
      while(1){
            if(!XUartLite_mIsReceiveEmpty  (XPAR_UARTLITE_0_BASEADDR)){
                  processesPacket();
            }

            if (TimerExpired != LastTimerExpired){
            if(TimerExpiredMin >= SECONDS_IN_A_MINUTE){
                  int i,p;
                  //For every monitored route
                  for(i=0; i<num_monitored; i++){
                        int dispRoute = monitored[i][0];
                        //For every trip in a monitored route
                        for(p=0;p<MAX_TRIPS_PER_ROUTE;p++){
                              //Tick ETAs by 1
                              if(routes[dispRoute].route_eta[p]>0){
                                    routes[dispRoute].route_eta[p]--;
                              }
                        }
                  }
```

```
                  TimerExpiredMin = 0;
            }
                if(TimerExpired >= SECONDS_BETWEEN_SCREENS){
                  int min_eta = 0;
                  int dispRoute;
                  int eta_sum = 0;
                  int k;
                  //For each monitored route
                  eta_sum = 0;
                        while(displayed < num_monitored-1){
                        displayed++;
                        dispRoute = monitored[displayed][0];
                        //For every trip in a monitored route
                        for(k=0;k<MAX_TRIPS_PER_ROUTE;k++){
                              //Find the minimum ETA
                              if(min_eta == 0 ||

      (min_eta>routes[dispRoute].route_eta[k] &&

      routes[dispRoute].route_eta[k] != 0)){
                                          min_eta = routes[dispRoute].route_eta[k];
                              }
//xil_printf("%d, %d eta: %d\n", monitored[displayed][0], k,
routes[dispRoute].route_eta[k]);
                        }
                        eta_sum += min_eta;
//xil_printf("min_eta: %d\n", min_eta);
                        if(min_eta>0 && ((lastDisplayed != displayed) ||
                              (min_eta != eta))){
                              //Display eta
                              route = dispRoute;
                              eta = min_eta;
                              dispMain();
                              lastDisplayed = displayed;
                              break;
                        }
                        }
                        if(eta_sum == 0){
                              dispWaiting();
//xil_printf("Waiting for update\n");
                       }
                        if(displayed==num_monitored-1){
                              displayed = -1;
                  }

                        repaint();
                        TimerExpired = 0;
            }
            }
      }

      return 0;
}
```

Routes.c

```
#include "routes.h"
#include "rt1.h"
#include "rt2.h"
#include "rt3.h"
#include "rt4.h"
#include "rt5.h"
#include "rt6.h"

void initRoutes(){
	int i, j;
	for(i=0; i < TOTAL_ROUTES; i++){
		for(j = 0; j < MAX_TRIPS_PER_ROUTE; j++){
			routes[i].avgLateness[j] = 0;
			routes[i].tripTimes[j] = 0;
			routes[i].route_eta[j] = 0;
		}
	}

	routes[1].tripTimes[1] = trip11Times;
	routes[1].tripTimes[1] = trip11Times;
	routes[2].tripTimes[1] = trip21Times;
	routes[2].tripTimes[2] = trip22Times;
	routes[3].tripTimes[1] = trip31Times;
	routes[3].tripTimes[2] = trip32Times;
	routes[4].tripTimes[1] = trip41Times;
	routes[4].tripTimes[2] = trip42Times;
	routes[4].tripTimes[3] = trip43Times;
	routes[5].tripTimes[1] = trip51Times;
	routes[5].tripTimes[2] = trip52Times;
	routes[6].tripTimes[1] = trip61Times;
	routes[6].tripTimes[2] = trip62Times;
	routes[6].tripTimes[3] = trip63Times;
	routes[6].tripTimes[4] = trip64Times;

}
```

Rt.1

```
/* Valid as of 5/5/08 */


#ifndef RT1_H_
#define RT1_H_
// Broad - Johnson - Highland
const unsigned char trip11Times[] = { // WEEKDAYS
40, // 40 stops total
15, // stop1   Osos @ Palm (City Hall)
16, // stop2   Palm @ Morro (WP)
18, // stop3   Nipomo @ Higuera
19, // stop4   Marsh @ Broad
22, // stop5   Broad @ Islay
25, // stop6   Broad @ Funston
27, // stop7   Broad @ Caudill
29, // stop8   Broad at the Brickyard - S
31, // stop9   Orcutt @ Laurel
34, // stop10  Orcutt @ Johnson -N
35, // stop11  Johnson @ Gregory
36, // stop12  Southwood @ Woodside
36, // stop13  Southwood @ Laurel Lane
37, // stop14  Laurel @ Laurel Lane Market
37, // stop15  Augusta @ Laurel - W
38, // stop16  Augusta @ Gerda
39, // stop17  Augusta @ Bishop -N
39, // stop18  Johnson @ Bishop -W
40, // stop19  Johnson @ Lizzie -W
42, // stop20  Johnson @ Marsh -W
44, // stop21  Monterey @ Toro
45, // stop22  Monterey @ Santa Rosa
47, // stop23  Osos @ Palm (City Hall)
48, // stop24  Palm @ Santa Rosa (WP)
51, // stop25  Santa Rosa @ Murray - N
53, // stop26  Foothill @ University Sq.
54, // stop27  Foothill @ Ferrini - W
55, // stop28  Foothill @ Cuesta
55, // stop29  Highland @ Cuesta
56, // stop30  Highland @ Jeffrey
57, // stop31  Patricia @ Highland
58, // stop32  Patricia @ Foothill
59, // stop33  Foothill @ La Entrada
0,  // stop34  Foothill @ Cuesta - E
1,  // stop35  Foothill @ Ferrini - E
1,  // stop36  Foothill @ Chorro - E
2,  // stop37  Foothill @ Casa - E
3,  // stop38  Casa @ Deseret - S
3,  // stop39  Murray @ Casa - W
4,  // stop40  Santa Rosa @ Murray - S
60, // ROUTE OVERHEAD
};

// End of rt1.c
#endif
```

Rt2.h

```
/* Valid as of 5/5/08 */


#ifndef RT2_H_
#define RT2_H_
// S Higuera - Suburban
const unsigned char trip21Times[] = {
23, // 23 stops total
25, // stop1   Osos @ Palm (City Hall)
28, // stop2   Nipomo @ Higuera
29, // stop3   Nipomo @ Marsh
30, // stop4   Pismo @ Carmel
31, // stop5   Pismo @ Archer
33, // stop6   Higuera @ South -S
35, // stop7   Higuera @ Bridge
37, // stop8   Prado Day Center
38, // stop9   Higuera @ Prado -S
39, // stop10  Higuera @ Granada
41, // stop11  Higuera @ Silver City
42, // stop12  Tank Farm @ Higuera
43, // stop13  Higuera @ Suburban -N
44, // stop14  Higuera @ Hind
45, // stop15  Higuera @ Prado -N
46, // stop16  Higuera @ Chumash
47, // stop17  Higuera @ Elks
50, // stop18  Higuera @ South -N
52, // stop19  Marsh @ Archer
54, // stop20  Marsh @ Broad
55, // stop21  Marsh @ Chorro
56, // stop22  Marsh @ Osos
57, // stop23  Santa Rosa @ Higuera
40, // ROUTE OVERHEAD
};

//  S Higuera - Suburban (Evening)
const unsigned char trip22Times[] = {
22, // 22 stops total
50, // stop1   Osos @ Palm (City Hall)
52, // stop2   Nipomo @ Higuera
53, // stop3   Nipomo @ Marsh
54, // stop4   Pismo @ Carmel
55, // stop5   Pismo @ Archer
56, // stop6   Higuera @ South -S
58, // stop7   Higuera @ Bridge
0,  // stop8   Higuera @ Granada
1,  // stop9   Higuera @ Silver City
2,  // stop10  Tank Farm @ Higuera
3,  // stop11  Higuera @ Suburban -N
4,  // stop12  Higuera @ Hind
5,  // stop13  Higuera @ Prado -N
5,  // stop14  Higuera @ Margarita (DMV) -N
6,  // stop15  Higuera @ Chumash
7,  // stop16  Higuera @ Elks
8,  // stop17  Higuera @ South -N
```

```
12, // stop18  Marsh @ Archer
14, // stop19  Marsh @ Broad
15, // stop20  Marsh @ Chorro
16, // stop21  Marsh @ Osos
17, // stop22  Santa Rosa @ Higuera
60, // ROUTE OVERHEAD
};

// End of rt2.c
#endif
```

Rt3.h

```
/* Valid as of 5/5/08 */


#ifndef RT3_H_
#define RT3_H_
// Broad - Johnson - Marigold
const unsigned char trip31Times[] = {
25, // 25 stops total
25, // stop1   Osos @ Palm (City Hall)
28, // stop2   Marsh @ Johnson -E
31, // stop3   Johnson @ Lizzie -E
31, // stop4   Johnson @ Bishop -E
32, // stop5   Johnson @ Sydney
33, // stop6   Johnson @ La Cita
34, // stop7   Laurel @ Augusta
36, // stop8   Laurel @ Southwood
36, // stop9   Laurel @ Camden
37, // stop10  Orcutt @ Laurel
39, // stop11  Orcutt @ Johnson -N
42, // stop12  Tank Farm @ Wavertree
42, // stop13  Tank Farm @ Brookpine
43, // stop14  Tank Farm @ Hollyhock
43, // stop15  Tank Farm @ Poinsettia
44, // stop16  Broad @ Marigold Center
46, // stop17  Broad @ Hopkins
48, // stop18  Broad @ Rockview
49, // stop19  Broad @ The Brickyard
49, // stop20  Broad @ Humbert
50, // stop21  Broad @ Santa Barbara
51, // stop22  Chorro @ Upham
52, // stop23  Chorro @ Islay
53, // stop24  Chorro @ Marsh
54, // stop25  Chorro @ Monterey
40, // ROUTE OVERHEAD
};

//  Broad - Johnson - Marigold (Evening)
const unsigned char trip32Times[] = {
21, // 22 stops total
18, // stop1   Osos @ Palm (City Hall)
20, // stop2   Marsh @ Johnson -E
22, // stop3   Johnson @ Lizzie -E
23, // stop4   Johnson @ Bishop -E
24, // stop5   Johnson @ Sydney
25, // stop6   Johnson @ La Cita
26, // stop7   Laurel @ Augusta
27, // stop8   Laurel @ Southwood
27, // stop9   Laurel @ Camden
28, // stop10  Orcutt @ Laurel
30, // stop11  Orcutt @ Johnson -N
32, // stop12  Tank Farm @ Wavertree
32, // stop13  Tank Farm @ Brookpine
33, // stop14  Tank Farm @ Hollyhock
33, // stop15  Tank Farm @ Poinsettia
```

```
34, // stop16  Broad @ Marigold Center
36, // stop17  Broad @ Hopkins
38, // stop18  Broad @ Rockview
39, // stop19  Broad @ The Brickyard
39, // stop20  Broad @ Humbert
40, // stop21  Broad @ Santa Barbara
60, // ROUTE OVERHEAD
};

// End of rt2.c
#endif
```

Rt4.h

```
/* Valid as of 5/5/08 */


#ifndef RT4_H_
#define RT4_H_
// Madonna - Laguna Lake - Cal Poly
const unsigned char trip41Times[] = {
34, // 34 stops total
10, // stop1   Downtown Transit Center**
13, // stop2   Santa Rosa at Marsh
14, // stop3   Santa Rosa at Buchon
15, // stop4   Santa Barbara at Church
15, // stop5   Santa Barbara at High
16, // stop6   South at Broad
16, // stop7   South at King
17, // stop8   South at Parker**
20, // stop9   Madonna Rd at Madonna Plaza
22, // stop10  Promenade**
24, // stop11  Madonna at Oceanaire
26, // stop12  LOVR at Madonna
28, // stop13  LOVR at Home Depot**
30, // stop14  LOVR at Auto Park Way
34, // stop15  LOVR at Laguna Village**
35, // stop16  LOVR at Oceanaire
36, // stop17  LOVR at Laguna Ln
37, // stop18  LOVR at Diablo
38, // stop19  LOVR at Valle Vista
40, // stop20  Foothill at Blarney
42, // stop21  Foothill at La Entrada
43, // stop22  Tassajara at Ramona
44, // stop23  Ramona at Palomar**
46, // stop24  Foothill at Chorro
49, // stop25  Foothill at Casa
52, // stop26  CalPoly Graphic Arts
0,  // stop27* CalPoly Mott Gym**
0,  // stop28  CalPoly Vista Grande
1,  // stop29  Grand at McCollum
2,  // stop30  Grand at Wilson
2,  // stop31  Mill at Grand
3,  // stop32  Mill at California
4,  // stop33  Mill at Johnson
4,  // stop34  Mill at Santa Barbara
60, // ROUTE OVERHEAD
};

// Madonna - Laguna Lake - Cal Poly
const unsigned char trip42Times[] = {
35, // 35 stops total
40, // stop1   Downtown Transit Center**
43, // stop2   Santa Rosa at Marsh
44, // stop3   Santa Rosa at Buchon
45, // stop4   Santa Barbara at Church
45, // stop5   Santa Barbara at High
46, // stop6   South at Broad
```

```
46, // stop7   South at King
47, // stop8   South at Parker**
50, // stop9   Madonna Rd at Madonna Plaza
52, // stop10  Promenade**
54, // stop11  Madonna at Oceanaire
56, // stop12  LOVR at Madonna
58, // stop13  LOVR at Home Depot**
0,  // stop14  LOVR at Auto Park Way
4,  // stop15  LOVR at Laguna Village**
5,  // stop16  LOVR at Oceanaire
6,  // stop17  LOVR at Laguna Ln
6,  // stop18  LOVR at Descanso
7,  // stop19  LOVR at Diablo
8,  // stop20  LOVR at Valle Vista
10, // stop21  Foothill at Blarney
12, // stop22  Foothill at La Entrada
13, // stop23  Tassajara at Ramona
14, // stop24  Ramona at Palomar**
16, // stop25  Foothill at Chorro
19, // stop26  Foothill at Casa
22, // stop27  CalPoly Graphic Arts
30, // stop28* CalPoly Mott Gym**
30, // stop29  CalPoly Vista Grande
31, // stop30  Grand at McCollum
32, // stop31  Grand at Wilson
32, // stop32  Mill at Grand
33, // stop33  Mill at California
34, // stop34  Mill at Johnson
34, // stop35  Mill at Santa Barbara
60, // ROUTE OVERHEAD
};

// Madonna - Laguna Lake - Cal Poly (Evening)
const unsigned char trip43Times[] = {
34, // 34 stops total
10, // stop1   Downtown Transit Center**
13, // stop2   Santa Rosa at Marsh
14, // stop3   Santa Rosa at Buchon
15, // stop4   Santa Barbara at Church
15, // stop5   Santa Barbara at High
16, // stop6   South at Broad
16, // stop7   South at King
17, // stop8   South at Parker**
20, // stop9   Madonna Rd at Madonna Plaza
22, // stop10  Promenade**
24, // stop11  Madonna at Oceanaire
26, // stop12  LOVR at Madonna
28, // stop13  LOVR at Home Depot**
30, // stop14  LOVR at Auto Park Way
34, // stop15  LOVR at Laguna Village**
35, // stop16  LOVR at Oceanaire
36, // stop17  LOVR at Laguna Ln
37, // stop18  LOVR at Diablo
38, // stop19  LOVR at Valle Vista
40, // stop20  Foothill at Blarney
42, // stop21  Foothill at La Entrada
43, // stop22  Tassajara at Ramona
```

```
44, // stop23  Ramona at Palomar**
46, // stop24  Foothill at Chorro
49, // stop25  Foothill at Casa
52, // stop26  CalPoly Graphic Arts
0,  // stop27* CalPoly Mott Gym**
0,  // stop28  CalPoly Vista Grande
1,  // stop29  Grand at McCollum
2,  // stop30  Grand at Wilson
2,  // stop31  Mill at Grand
3,  // stop32  Mill at California
4,  // stop33  Mill at Johnson
4,  // stop34  Mill at Santa Barbara
60, // ROUTE OVERHEAD
};
// End of rt4.c
#endif
```

Rt5.h

```
/* Valid as of 5/5/08 */


#ifndef RT5_H_
#define RT5_H_
// Cal Poly - Laguna Lake - Madonna
const unsigned char trip51Times[] = {
41, // 41 stops total
50, // stop1   Downtown Transit Center**
51, // stop2   Mill at Santa Rosa
52, // stop3   Mill at Johnson
53, // stop4   Mill at California
54, // stop5   Mill at Grand
56, // stop6   Grand at Abbott
56, // stop7   Grand at McCollum
57, // stop8   CalPoly Vista Grande
6,  // stop9*  CalPoly UU**
6,  // stop10  CalPoly Graphic Arts
7,  // stop11  CalPoly Mustang Stadium
9,  // stop12  Foothill at Casa
11, // stop13  Foothill at University Square**
12, // stop14  Foothill at Ferrini
13, // stop15  Foothill at Cuesta
14, // stop16  Foothill at Rosita
15, // stop17  Foothill at Blarney
17, // stop18  LOVR at Valle Vista
18, // stop19  LOVR at Diablo
18, // stop20  Perfumo at Del Rio
19, // stop21  Descanso at Del Rio
20, // stop22  Descanso at LOVR**
21, // stop23  LOVR at Laguna Ln
21, // stop24  LOVR at Oceanaire
22, // stop25  LOVR at Laguna Village**
23, // stop26  LOVR at Madonna
24, // stop27  LOVR at Home Depot**
26, // stop28  LOVR at Auto Park Way
30, // stop29  Madonna at Oceanaire
32, // stop30  Promenade**
33, // stop31  Madonna Rd at Madonna Plaza
36, // stop32  South at Parker
36, // stop33  South at Exposition
37, // stop34  South at King
38, // stop35  South at Meadow Park
39, // stop36  Santa Barbara at High
40, // stop37  Santa Barbara at Church
41, // stop38  Amtrak Station**
42, // stop39  Santa Rosa at Leff
43, // stop40  Santa Rosa at Buchon
44, // stop41  Santa Rosa at Higuera
60, // ROUTE OVERHEAD
};

// Cal Poly - Laguna Lake - Madonna
const unsigned char trip52Times[] = {
```

```
41, // 41 stops total
20, // stop1   Downtown Transit Center**
21, // stop2   Mill at Santa Rosa
22, // stop3   Mill at Johnson
23, // stop4   Mill at California
24, // stop5   Mill at Grand
26, // stop6   Grand at Abbott
26, // stop7   Grand at McCollum
27, // stop8   CalPoly Vista Grande
36, // stop9*  CalPoly UU**
36, // stop10  CalPoly Graphic Arts
37, // stop11  CalPoly Mustang Stadium
39, // stop12  Foothill at Casa
41, // stop13  Foothill at University Square**
42, // stop14  Foothill at Ferrini
43, // stop15  Foothill at Cuesta
44, // stop16  Foothill at Rosita
45, // stop17  Foothill at Blarney
47, // stop18  LOVR at Valle Vista
48, // stop19  LOVR at Diablo
48, // stop20  Perfumo at Del Rio
49, // stop21  Descanso at Del Rio
50, // stop22  Descanso at LOVR**
51, // stop23  LOVR at Laguna Ln
51, // stop24  LOVR at Oceanaire
52, // stop25  LOVR at Laguna Village**
53, // stop26  LOVR at Madonna
54, // stop27  LOVR at Home Depot**
56, // stop28  LOVR at Auto Park Way
0,  // stop29  Madonna at Oceanaire
2,  // stop30  Promenade**
3,  // stop31  Madonna Rd at Madonna Plaza
6,  // stop32  South at Parker
6,  // stop33  South at Exposition
7,  // stop34  South at King
8,  // stop35  South at Meadow Park
9,  // stop36  Santa Barbara at High
10, // stop37  Santa Barbara at Church
11, // stop38  Amtrak Station**
12, // stop39  Santa Rosa at Leff
13, // stop40  Santa Rosa at Buchon
14, // stop41  Santa Rosa at Higuera
60, // ROUTE OVERHEAD
};
// End of rt5.c
#endif
```

Rt6.h

```
/* Valid as of 5/5/08 */

#ifndef RT6_H_
#define RT6_H_
// Cal Poly - Highland
const unsigned char trip61Times[] = {
14, // 14 stops total
35, // stop 1     Cal Poly Mott Gym
40, // stop 2     Cal Poly Ag Sci
41, // stop 3     Highland at Mt. Bishop
43, // stop 4     Highland at Cuesta
45, // stop 5     Highland at Jeffrey
45, // stop 6     Patricia at Highland
46, // stop 7     Patricia at Foothill
47, // stop 8     La Entrada at Del Norte
48, // stop 9     Ramona at Tassajara
49, // stop 10    Ramona at Palomar
51, // stop 11    Foothill at Chorro
53, // stop 12    Casa at Murray
54, // stop 13    Casa at Deseret
58, // stop 14    Cal Poly at Graphic Arts
30, // ROUTE OVERHEAD
};

// Cal Poly - Downtown
const unsigned char trip62Times[] = {
15, // 15 stops total
4,  // stop 1     Cal Poly Mott Gym
4,    // stop 2   Cal Poly Vista Grande
5,    // stop 3   Grand at McCollum
5,    // stop 4   Grand at Wilson
6,    // stop 5   Mill at Park
6,    // stop 6   Mill at California
7,    // stop 7   Mill at Johnson
8,    // stop 8   Mill at Santa Rosa
15,   // stop 9   Osos at Palm (City Hall)
16,   // stop 10  Mill at Santa Rosa
17, // stop 11    Mill at Johnson
18, // stop 12    Mill at California
19,   // stop 13  California at Phillips
20, // stop 14    California at Taft
25,   // stop 15  Cal Poly at Graphic Arts
30, // ROUTE OVERHEAD
};

// Cal Poly - Highland - Downtown
const unsigned char trip63Times[] = {
30,   // 30 stops total
34,   // stop 1   Cal Poly Mott Gym
34, // stop 2     Cal Poly Vista Grande
35,   // stop 3   Grand at McCollum
35, // stop 4     Grand at Wilson
36, // stop 5     Mill at Park
```

```
36,   // stop 6   Mill at California
37, // stop 7     Mill at Johnson
38, // stop 8     Mill at Santa Rosa
45, // stop 9     Osos at Palm (City Hall)
46, // stop 10    Mill at Santa Rosa
47, // stop 11    Mill at Johnson
48, // stop 12    Mill at California
49, // stop 13    California at Phillips
50, // stop 14    California at Taft
55, // stop 15    Cal Poly at Graphic Arts
5,    // stop 16  Cal Poly Mott Gym
5,    // stop 17  Cal Poly Admin
10, // stop 18    Cal Poly Ag Sci
11, // stop 19    Highland at Mt. Bishop
13, // stop 20    Highland at Cuesta
15, // stop 21    Highland at Jeffrey
15, // stop 22    Patricia at Highland
16, // stop 23    Patricia at Foothill
17, // stop 24    La Entrada at Del Norte
18, // stop 25    Ramona at Tassajara
19, // stop 26    Ramona at Palomar
21, // stop 27    Foothill at Chorro
23, // stop 28    Casa at Murray
24, // stop 29    Casa at Deseret
28, // stop 30    Cal Poly at Graphic Arts
60,   // ROUTE OVERHEAD
};

// Cal Poly - Highland - Downtown (Evening)
const unsigned char trip64Times[] = {
30,   // 30 stops total
40,   // stop 1   Cal Poly Mott Gym
40,   // stop 2   Cal Poly Vista Grande
41,   // stop 3   Grand at McCollum
41,   // stop 4   Grand at Wilson
42,   // stop 5   Mill at Park
42,   // stop 6   Mill at California
43,   // stop 7   Mill at Johnson
44,   // stop 8   Mill at Santa Rosa
51,   // stop 9   Osos at Palm (City Hall)
52,   // stop 10  Mill at Santa Rosa
53,   // stop 11  Mill at Johnson
54,   // stop 12  Mill at California
55,   // stop 13  California at Phillips
56,   // stop 14  California at Taft
1,    // stop 15  Cal Poly at Graphic Arts
10,   // stop 16  Cal Poly Mott Gym
10,   // stop 17  Cal Poly Admin
15,   // stop 18  Cal Poly Ag Sci
16,   // stop 19  Highland at Mt. Bishop
18,   // stop 20  Highland at Cuesta
20,   // stop 21  Highland at Jeffrey
20,   // stop 22  Patricia at Highland
21,   // stop 23  Patricia at Foothill
22,   // stop 24  La Entrada at Del Norte
23,   // stop 25  Ramona at Tassajara
24,   // stop 26  Ramona at Palomar
```

```
26,   // stop 27  Foothill at Chorro
28,   // stop 28  Casa at Murray
29,   // stop 29  Casa at Deseret
33,   // stop 30  Cal Poly at Graphic Arts
60,   // ROUTE OVERHEAD
};
// End of rt6.c
#endif
```

Signconfig.h

```
#ifndef SIGNCONFIG_H_
#define SIGNCONFIG_H_

/**
 * This array is sign-specific.  It indicates the
 * route and stop to monitor.
 *
 * The format of this array is a n-deep array of
 * unsigned char pairs.  The first represents route number
 * the second, the stop number.
 * For example, monitoring two stops: route 3, trip 1, stop 5 and route 4,
 * trip 2, stop 6: {{3, 1, 5}, {4, 2, 6}}
 *
 * Num_monitored is simply the number of pairs monitored.
 */

static unsigned char monitored[3][3] = {{3, 1, 5}, {6, 2, 4}, {6, 1, 5}};
static int num_monitored = 3;

#endif /*SIGNCONFIG_H_*/
```

Spi.h

```
#ifndef _SPI_
      #define _SPI_
//REGSISTER CONTROL FXNS

#define XPAR_SPI_INTERFACE_BASEADDR XPAR_SPI_0_BASEADDR

int spi_status();            //gives the output of SPI STATUS REG
void spi_data(int input);    //outputs to SPI DATA REG
void spi_slave_select(int input); //OUTPUTS TO SLAVE SELECT REG

#endif
```

Spi.c

```
#include "xparameters.h"
#include "xio.h"
#include "spi.h"

int spi_status()
{
      return(XIo_In32(XPAR_SPI_INTERFACE_BASEADDR+0x64));
}

void spi_data(int input)
{
      XIo_Out32(XPAR_SPI_INTERFACE_BASEADDR+0x68 ,input);
}

void spi_slave_select(int input)
{
      XIo_Out32(XPAR_SPI_INTERFACE_BASEADDR+0x70 ,input);
}
```

Timer.h

```
#ifndef TIMER_H_
#define TIMER_H_

#include "xtmrctr.h"
#include "xintc.h"

#define TIMER_CNTR_0               0
#define RESET_VALUE            0xFD056500

XIntc InterruptController;      /* The instance of the Interrupt Controller
*/
XTmrCtr TimerCounterInst;       /* The instance of the Timer Counter */

volatile int TimerExpired;
volatile int TimerExpiredMin;
volatile int LastTimerExpired;
volatile int TotalSentCount;

XStatus IntrSetup(XIntc* IntcInstancePtr,
                  XTmrCtr* InstancePtr,
                  Xuint16 DeviceId,
                  Xuint16 IntrId,
                  Xuint8 TmrCtrNumber);

static XStatus TmrCtrSetupIntrSystem(XIntc* IntcInstancePtr,
                                     XTmrCtr* InstancePtr,
                                     Xuint16 IntrId,
                                     Xuint8 TmrCtrNumber);
```

```
void TimerCounterHandler(void *CallBackRef, Xuint8 TmrCtrNumber);

#endif /*TIMER_H_*/
```

Timer.c

```
#include "stdio.h"
#include "xparameters.h"
#include "xtmrctr.h"
#include "xintc.h"
#include "mb_interface.h"

#include "timer.h"
#include "graphics.h"

/**
 * Set up timer counter
 */
XStatus IntrSetup(XIntc* IntcInstancePtr,
                  XTmrCtr* TmrCtrInstancePtr,
                  Xuint16 DeviceId,
                  Xuint16 IntrId,
                  Xuint8 TmrCtrNumber){
    XStatus status;
    LastTimerExpired = 0;

      //Initialize timer counter
    status = XTmrCtr_Initialize(TmrCtrInstancePtr, DeviceId);
    if (status != XST_SUCCESS)
    {
        return XST_FAILURE;
    }

      //Initialize interrupt controller
    status = XIntc_Initialize(IntcInstancePtr, XPAR_XPS_INTC_0_DEVICE_ID);
    if (status != XST_SUCCESS)
    {
       return XST_FAILURE;
    }

      //Configure timer counter's interrupt system
    status = TmrCtrSetupIntrSystem(IntcInstancePtr,
                                   TmrCtrInstancePtr,
                                   IntrId,
                                   TmrCtrNumber);
    if (status != XST_SUCCESS)
    {
        return XST_FAILURE;
    }

      //Activate interrupt controller
    status = XIntc_Start(&InterruptController, XIN_REAL_MODE);

    if (status != XST_SUCCESS)
    {
        return XST_FAILURE;
    }

      //Register timer interrupt handler
    XTmrCtr_SetHandler(TmrCtrInstancePtr, TimerCounterHandler,
```

```
                        TmrCtrInstancePtr);

      //Configure timer counter
    XTmrCtr_SetOptions(TmrCtrInstancePtr, TmrCtrNumber,
                        XTC_INT_MODE_OPTION | XTC_AUTO_RELOAD_OPTION);

      //More configutation of timer counter
    XTmrCtr_SetResetValue(TmrCtrInstancePtr, TmrCtrNumber, RESET_VALUE);

      //Enable timer counter interrupt
    XIntc_Enable(IntcInstancePtr, IntrId);

      //Enable microblaze interrupts
    microblaze_enable_interrupts();

      //Begin timer counter ticking
    XTmrCtr_Start(TmrCtrInstancePtr, TmrCtrNumber);

    return XST_SUCCESS;
}

/**
 * Timer interrupt handler
 */
void TimerCounterHandler(void *CallBackRef, Xuint8 TmrCtrNumber)
{
      LastTimerExpired = TimerExpired;
    TimerExpired++;
    TimerExpiredMin++;
}

/**
 * Setup interrupts for the timer counter
 */
static XStatus TmrCtrSetupIntrSystem(XIntc* IntcInstancePtr,
                                     XTmrCtr* TmrCtrInstancePtr,
                                     Xuint16 IntrId,
                                     Xuint8 TmrCtrNumber)
{
    XStatus status;

    //Register timer counter intrrupt handler
    status = XIntc_Connect(IntcInstancePtr, IntrId,
                           (XInterruptHandler)XTmrCtr_InterruptHandler,
                           (void *)TmrCtrInstancePtr);

    if (status != XST_SUCCESS)
    {
        return XST_FAILURE;
    }
}
```

Appendix C – Bitmap Utilities

This pair of tools is for conversion of monochrome bitmaps to C arrays and vice versa. Bitmap_util can handle any size bitmap, but Rev_Bitmap can only cope with 320x240 pixel images.

bitmapUtil.cpp

```
// bitmapUtil.cpp : Defines the entry point for the console application.

#include "stdafx.h"
#include "stdio.h"
#include "stdlib.h"
#include "string.h"

#define FILESIZE 10000
#define WIDTH 32
#define HEIGHT 45

int main(int argc, char* argv[])
{
	if(argc < 3){
		printf("Usage:\n\tbitmapUtil.exe inFile.bmp outFile.txt\n");
		return 1;
	}
	FILE* inFile = fopen(argv[1], "r");
	FILE* outFile = fopen(argv[2], "a");

	if(inFile== NULL || outFile== NULL){
		return 1;
	}
	unsigned char* buf = (unsigned char*)malloc(FILESIZE);
	size_t nRead = fread(buf, 1, FILESIZE, inFile);

	size_t imgsize = (*((short*)(buf+0x0012)))*(*((short*)(buf+0x0016)))/8;

	argv[1][strlen(argv[1])-4] = '\0';
	fprintf(outFile, "\nunsigned char %s[%d] = {\n\t", argv[1], imgsize);

	buf += *((short*)(buf+0x000a));

	int i;
	for(i=0; i < imgsize; i++){
		fprintf(outFile, "0x%02X", *(buf+i));
		if(i!=imgsize-1) fprintf(outFile, ", ");
		if((i+1)%4==0) fprintf(outFile, "  ");
		if(i!=imgsize-1 && (i+1)%16==0) {
			fprintf(outFile, "\n\t");
		}
	}
	fprintf(outFile, "\n};\n");

	return 0;
}
```

Revbitmap.cpp

```
// revBitmap.cpp : Defines the entry point for the console application.

#include "stdafx.h"
#include "eta_bkgnd2.h"

#define IMGVAR out

unsigned char bmp_header[62] = {0x42, 0x4d, 0xbe, 0x25,0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x3e, 0x00,   0x00, 0x00, 0x28, 0x00,0x00, 0x00, 0x40,
0x01, 0x00, 0x00, 0xf0, 0x00,   0x00, 0x00, 0x01, 0x00,0x01, 0x00, 0x00,
0x00, 0x00, 0x00, 0x80, 0x25,   0x00, 0x00, 0x00, 0x00,0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,   0x00, 0x00, 0x00, 0x00,0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0xff, 0xff,   0xff, 0x00};

int main(int argc, char* argv[])
{
      FILE* inFile = fopen(argv[1], "w");

      int i;
      int j;

      unsigned char imageFV[240][40];
      for(i=0;i<240;i++){
            for(j=0; j<40; j++){
                  imageFV[239-i][j] = IMGVAR[i*40+j];
            }
      }

      fwrite(bmp_header, 1, 62, inFile);
      fwrite(imageFV, 1, 9600, inFile);

      return 0;
}
```

Appendix D – Character and Screen Bitmaps

Figure 12 - Static Elements

Figure 13 - Character Tiles