**/\* 本文档将按照以下顺序记录项目源代码:**

**main.cpp->oledfont.h->OLED.h->OLED.cpp->Alert.h->Alert.cpp->WebServer.h->WeatherNow.h->WeatherNow.cpp->Tool.h->Tool.cpp-> Hash.h->Hash.cpp->Universal.h->rivest\_cipher\_4.hpp->make\_ptr.hpp->fourier\_transform.hpp->tree.hpp**

**\*/**

**// main.cpp-----------------------------------------------------------------------------------------------------------**

**// [RMSHE Infinty] 嵌入式智能可燃气体报警系统主控软件 GasSensorGen3\_Program V2023.04.15 Powered by 马山河(RMSHE)**

**// MCU: ESP8266; MODULE: ESP12F;**

**#include <Hash.h>**

**#include <PubSubClient.h>**

**#include <WeatherNow.h>**

**#include <stack>**

**#include "Alert.h"**

**#include "OLED.h"**

**#include "Tool.h"**

**#include "Universal.h"**

**#include "WebServer.h"**

**#include "fourier\_transform.hpp"**

**#include "tree.hpp"**

**ALERT alert;**

**OLED oled;**

**HTTPClient http;**

**WiFiClient client;**

**TOOL tool;**

**Ticker TimeRefresh\_ticker;**

**Ticker System\_time;**

**Ticker Desktop\_ticker;**

**Ticker CMDControlPanel\_ticker;**

**Ticker UpdateWeather;**

**Ticker WIFI\_Test;       // 用于检测WIFI是否连接;**

**WeatherNow weatherNow;  // 建立WeatherNow对象用于获取天气信息**

**PubSubClient mqtt\_client(client);     // 建立MQTT客户端对象;**

**ESP8266WebServer server(ServerPort);  // 建立网络服务器对象，该对象用于响应HTTP请求。监听端口（80）**

**//[System Mode&Status(系统模式和状态)]**

**bool Charging\_State = false;    // 充电状态(0:没在充电; 1:正在充电);**

**bool WIFI\_State = false;        // WIFI状态(0:断网; 1:联网);**

**bool Developer\_Mode = false;    // 开发者模式(0:常规运行; 1:进入开发者模式);**

**bool allowResponse = true;      // true:允许服务器对客户端进行响应;**

**bool allowDownloadMode = true;  // true:允许进入下载模式;**

**bool freezeMode = false;  //[浅度休眠模式-freeze], 冻结I/O设备, 关闭外设, ESP-12F进入Modem-sleep模式, 程序上只运行CMDControlPanel网络服务, 其他服务冻结;**

**bool diskMode = false;  //[深度休眠模式-disk] 运行状态(GPIO\_Status, 系统模式和状态, 文本框信息)数据存到Flash(醒来时恢复状态), 然后ESP12F进入深度睡眠;**

**bool CMDCP\_State = false;  // CMDCP是否被打开(trrue:表示被打开);**

**class FlashFileSystem {**

**private:**

**FSInfo Flash\_info;**

**Dir FileDirectory;**

**stack<String> WorkingDirectoryStack;  // 工作路径的栈(用来储存历史工作路径, 一遍返回上一个路径);**

**String WorkingDirectory = "/";        // 当前工作路径;**

**// 递归删除文件夹;**

**void deleteFolder(String path) {**

**// 打开待删除目录;**

**Dir FileDirectory = LittleFS.openDir(path);**

**// 遍历目录;**

**while (FileDirectory.next()) {**

**// 获取当前项文件的名称;**

**String entryPath = path + "/" + FileDirectory.fileName();**

**// 检查当前项是否为目录;**

**if (LittleFS.exists(entryPath)) {**

**deleteFolder(entryPath);  // 如果是目录则循环调用自身函数递归删除该目录;**

**} else {**

**LittleFS.remove(entryPath);  // 如果不是目录则删除该文件;**

**}**

**}**

**LittleFS.rmdir(path);  // 在删除该目录下的所有文件和目录后删除这个父目录;**

**}**

**public:**

**// 获取Flash信息;**

**String getFlash\_info() {**

**LittleFS.begin();           // 启动LittleFS**

**LittleFS.info(Flash\_info);  // 获取闪存文件系统信息**

**/\***

**// 可用空间总和（单位：字节）**

**Serial.print("totalBytes: ");**

**Serial.print(Flash\_info.totalBytes);**

**Serial.println(" Bytes");**

**// 已用空间（单位：字节）**

**Serial.print("usedBytes: ");**

**Serial.print(Flash\_info.usedBytes);**

**Serial.println(" Bytes");**

**// 使用占比;**

**Serial.print("Proportion: ");**

**Serial.print(Proportion);**

**Serial.println(" %");**

**// 最大文件名字符限制（含路径和'\0'）**

**Serial.print("maxPathLength: ");**

**Serial.println(Flash\_info.maxPathLength);**

**// 最多允许打开文件数量**

**Serial.print("maxOpenFiles: ");**

**Serial.println(Flash\_info.maxOpenFiles);**

**// 存储块大小**

**Serial.print("blockSize: ");**

**Serial.println(Flash\_info.blockSize);**

**// 存储页大小**

**Serial.print("pageSize: ");**

**Serial.println(Flash\_info.pageSize);**

**\*/**

**// 计算空间使用占比;**

**unsigned char Proportion =**

**static\_cast<unsigned char>(round((static\_cast<float>(Flash\_info.usedBytes) / static\_cast<float>(Flash\_info.totalBytes)) \* 100));**

**// 计算使用占比的文本进度条;**

**String ProportionBar = "[          ] ";**

**for (unsigned char i = 1; i < static\_cast<unsigned char>(0.1 \* Proportion); ++i) ProportionBar[i] = '=';**

**/\*显示到OLED屏幕上\*/**

**return "Flash Info (Bytes)\nTotal:" + String(Flash\_info.totalBytes) + "\nUsed: " + String(Flash\_info.usedBytes) + "\n" + ProportionBar +**

**String(Proportion) + "%\nMaxPathLength:" + String(Flash\_info.maxPathLength) + "\nMaxOpenFiles:" + String(Flash\_info.maxOpenFiles) +**

**"\nBlockSize:" + String(Flash\_info.blockSize) + "\nPageSize:" + String(Flash\_info.pageSize);**

**}**

**// 获取当前的工作目录;**

**String getWorkDirectory() { return WorkingDirectory; }**

**// 返回上一个工作目录;**

**void backDirectory() {**

**// 如果栈非空则弹出一个上一个工作目录将其设置为当前工作目录并返回;**

**if (!WorkingDirectoryStack.empty()) {**

**WorkingDirectory = WorkingDirectoryStack.top();**

**WorkingDirectoryStack.pop();**

**}**

**}**

**// 切换当前工作目录;**

**void changeDirectory(String path) {**

**// 工作路径压入堆栈;**

**WorkingDirectoryStack.push(path);**

**WorkingDirectory = path;**

**}**

**// 显示工作目录下之内容;**

**String listDirectoryContents() {**

**// 文件路径 = 工作路径 + 文件名;**

**String path = WorkingDirectory;**

**/\*获取工作目录下的所有文件名\*/**

**LittleFS.begin();  // 启动闪存文件系统**

**// 显示目录中文件内容以及文件大小**

**FileDirectory = LittleFS.openDir(path.c\_str());  // 建立“目录”对象**

**String FlashlistDirectory = "";**

**while (FileDirectory.next()) {  // dir.next()用于检查目录中是否还有“下一个文件”**

**FlashlistDirectory += path + FileDirectory.fileName() + "\n";**

**}**

**return FlashlistDirectory;**

**}**

**// 读文件(工作目录下的文件名, 或直接指定文件路径[指定文件路径后工作目录下的文件名就无效了]);**

**String readFile(String fileName, String filePath = "") {**

**String path;**

**// 如果直接指定文件路径就优先使用文件路径(filePath是用来方便给系统内部调用的);**

**if (filePath == "")**

**path = WorkingDirectory + fileName;  // 文件路径 = 工作路径 + 文件名(CMD调用);**

**else**

**path = filePath;  // 优先使用(系统API调用);**

**LittleFS.begin();  // 启动LittleFS;**

**File dataFile;**

**String File\_Info = "";**

**// 确认闪存中是否有文件**

**if (LittleFS.exists(path.c\_str())) {**

**File dataFile = LittleFS.open(path.c\_str(), "r");  // 建立File对象用于从LittleFS中读取文件;**

**// 读取文件内容并且通过串口监视器和OLED输出文件信息**

**while (dataFile.available()) {**

**File\_Info += (char)dataFile.read();**

**}**

**dataFile.close();  // 完成文件读取后关闭文件**

**} else {**

**File\_Info = "[FLASH FILE NOT FOUND]: " + path;**

**}**

**return File\_Info;**

**}**

**// 文件追加内容, 如果文件不存在则创建后追加(内容, 工作目录下的文件名, 或直接指定文件路径[指定文件路径后工作目录下的文件名就无效了]);**

**void fileAppend(String text, String fileName, String filePath = "") {**

**String path;**

**// 如果直接指定文件路径就优先使用文件路径(filePath是用来方便给系统内部调用的);**

**if (filePath == "")**

**path = WorkingDirectory + fileName;  // 文件路径 = 工作路径 + 文件名(CMD调用);**

**else**

**path = filePath;  // 优先使用(系统API调用);**

**LittleFS.begin();  // 启动LittleFS;**

**File dataFile;**

**// 确认闪存中是否有文件**

**if (LittleFS.exists(path)) {**

**dataFile = LittleFS.open(path.c\_str(), "a");  // 建立File对象用于向LittleFS中的file对象追加信息(添加);**

**} else {**

**dataFile = LittleFS.open(path.c\_str(), "w");  // 建立File对象用于向LittleFS中的file对象写入信息(新建&覆盖);**

**}**

**dataFile.print(text);  // 向dataFile写入字符串信息**

**dataFile.close();      // 完成文件写入后关闭文件}**

**}**

**// 文件覆盖内容(内容, 工作目录下的文件名, 或直接指定文件路径[指定文件路径后工作目录下的文件名就无效了]);**

**void fileCover(String text, String fileName, String filePath = "") {**

**String path;**

**// 如果直接指定文件路径就优先使用文件路径(filePath是用来方便给系统内部调用的);**

**if (filePath == "")**

**path = WorkingDirectory + fileName;  // 文件路径 = 工作路径 + 文件名(CMD调用);**

**else**

**path = filePath;  // 优先使用(系统API调用);**

**LittleFS.begin();  // 启动LittleFS;**

**File dataFile;**

**// 确认闪存中是否有文件**

**dataFile = LittleFS.open(path.c\_str(), "w");  // 建立File对象用于向LittleFS中的file对象写入信息(新建&覆盖);**

**dataFile.print(text);  // 向dataFile写入字符串信息**

**dataFile.close();      // 完成文件写入后关闭文件**

**}**

**// 创建或覆盖文件(工作目录下的文件名);**

**void createFile(String fileName) {**

**// 文件路径 = 工作路径 + 文件名;**

**String path = WorkingDirectory + fileName;**

**LittleFS.begin();  // 启动LittleFS;**

**File dataFile;**

**dataFile = LittleFS.open(path.c\_str(), "w");  // 建立File对象用于向LittleFS中的file对象写入信息(新建&覆盖);**

**dataFile.println("");  // 向dataFile写入空信息;**

**dataFile.close();      // 完成文件写入后关闭文件}**

**}**

**// 创建目录(工作目录下的文件夹名);**

**void makeDirector(String dirName) {**

**// 文件路径 = 工作路径 + 文件名;**

**String path = WorkingDirectory + dirName;**

**LittleFS.begin();  // 启动LittleFS;**

**LittleFS.mkdir(path.c\_str());  // 创建目录;**

**}**

**// 删除文件(工作目录下的文件名, 或直接指定文件路径[指定文件路径后工作目录下的文件名就无效了]);**

**bool removeFile(String fileName, String filePath = "") {**

**String path;**

**// 如果直接指定文件路径就优先使用文件路径(filePath是用来方便给系统内部调用的);**

**if (filePath == "")**

**path = WorkingDirectory + fileName;  // 文件路径 = 工作路径 + 文件名(CMD调用);**

**else**

**path = filePath;  // 优先使用(系统API调用);**

**LittleFS.begin();  // 启动闪存文件系统**

**// 从闪存中删除文件**

**if (LittleFS.remove(path.c\_str())) {**

**return true;**

**} else {**

**return false;**

**}**

**}**

**// 删除文件夹(工作目录下的文件夹名, 或直接指定文件路径[指定文件路径后工作目录下的文件名就无效了]);**

**void removeDirector(String dirName, String dirPath = "") {**

**String path;**

**// 如果直接指定文件路径就优先使用文件路径(filePath是用来方便给系统内部调用的);**

**if (dirPath == "")**

**path = WorkingDirectory + dirName;  // 文件路径 = 工作路径 + 文件名(CMD调用);**

**else**

**path = dirPath;  // 优先使用(系统API调用);**

**LittleFS.begin();  // 启动闪存文件系统**

**deleteFolder(path);  // 递归删除文件夹;**

**}**

**// 复制文件(源文件路径, 目标文件路径, 是否移动文件[true:复制完成后删除源文件]);**

**void copyFile(String sourceFilePath, String targetFilePath, bool moveMode = false) {**

**LittleFS.begin();  // 启动闪存文件系统**

**File source = LittleFS.open(sourceFilePath, "r");  // 打开源文件(读);**

**File target = LittleFS.open(targetFilePath, "w");  // 打开(创建)目标文件(写);**

**/\***

**这段代码将从源文件source读取的数据写入目标文件target。**

**它通过使用 source.available() 方法来检查源文件是否还有未读取的数据，并使用 target.write(source.read()) 方法将读取的数据写入目标文件。**

**这段代码的前提是：源文件source和目标文件target已经打开。**

**\*/**

**if (source && target) {**

**while (source.available()) {**

**target.write(source.read());**

**}**

**}**

**source.close();**

**target.close();**

**if (moveMode) LittleFS.remove(sourceFilePath.c\_str());  // 如果设为移动模式则删除源文件;**

**}**

**// 复制目录(源目录路径, 目标目录路径, 是否移动目录[true:复制完成后删除源目录]);**

**void copyDir(String sourceDirPath, String targetDirPath, bool moveMode = false) {**

**LittleFS.begin();  // 启动闪存文件系统**

**if (!LittleFS.exists(sourceDirPath)) return;                         // 判断源路径是否是目录;**

**if (!LittleFS.exists(targetDirPath)) LittleFS.mkdir(targetDirPath);  // 判断目标路径是否是目录(否则创建一个目录);**

**Dir sourceDir = LittleFS.openDir(sourceDirPath);  // 打开源目录;**

**// 读取源目录下的所有文件和子目录;**

**while (sourceDir.next()) {**

**// 获得源文件路径和目标文件路径;**

**String sourceFilePath = sourceDirPath + "/" + sourceDir.fileName();**

**String targetFilePath = targetDirPath + "/" + sourceDir.fileName();**

**// 判断父目录下的内容是目录还是文件;**

**if (sourceDir.isDirectory()) {**

**copyDir(sourceFilePath, targetFilePath, moveMode);  // 如果是目录则循环调用自身函数递归复制所有子目录下的文件;**

**} else {**

**copyFile(sourceFilePath, targetFilePath, moveMode);  // 如果是文件则直接复制文件;**

**}**

**}**

**if (moveMode) deleteFolder(sourceDirPath);  // 如果设为移动模式则递归删除源目录;**

**}**

**/\***

**这段代码是一个查找文件的函数。该函数接收两个参数：dirPath (目录路径) 和 fileName (待查找文件名称)。该函数执行以下步骤：**

**启动闪存文件系统**

**打开目录dirPath**

**循环读取目录中的所有文件/目录：**

**a. 如果读取的是目录，递归调用该函数，并将其结果加入foundFile字符串。**

**b. 如果读取的是文件：**

**i. 以"."分割该文件的文件名，以得到其扩展名。**

**ii. 比较待查找文件名和该文件的文件名：**

**如果待查找文件名为"."，说明查找所有文件，不进行筛选。**

**如果待查找文件名为"\*.txt"，说明查找所有扩展名为"txt"的文件，按扩展名筛选。**

**如果待查找文件名为"a.txt"，说明查找文件名为"a.txt"的文件，按文件名筛选。**

**返回找到的文件路径列表(存储在foundFile字符串中)**

**注：**

**LittleFS.begin()是闪存文件系统的初始化函数。**

**LittleFS.openDir(dirPath)是打开目录的函数。**

**dir.next()是读取下一个文件/目录的函数，如果还有下一个文件/目录，则返回true，否则返回**

**\*/**

**// 查找指定目录下的文件(dirPath(目录路径), fileName(待查找文件名称));**

**String findFiles(String dirPath, String fileName) {**

**String foundFile = "";**

**// 按"."分割fileName待查找文件名字符串;**

**vector<String> targetName = oled.strsplit(fileName, ".");**

**LittleFS.begin();  // 启动闪存文件系统**

**Dir dir = LittleFS.openDir(dirPath);  // 打开目录;**

**while (dir.next()) {**

**if (dir.isDirectory()) {**

**foundFile += findFiles(dirPath + dir.fileName() + "/", fileName);**

**} else {**

**String foundName = dir.fileName();                               // 获取目录中文件的文件名;**

**vector<String> foundName\_Split = oled.strsplit(foundName, ".");  // 按"."分割查找到的文件名字符串;**

**if (targetName[0] == "\*" && targetName[1] == "\*") {  //[fileName] = \*.\* : 查找所有文件;**

**// 不筛选;**

**foundFile += dirPath + foundName + "\n";**

**} else if (targetName[0] == "\*" && targetName[1] != "\*") {  //[fileName] = \*.txt : 查找所有扩展名为txt的文件;**

**// 按扩展名筛选文件;**

**if (foundName\_Split[1] == targetName[1]) foundFile += dirPath + foundName + "\n";**

**} else if (targetName[0] != "\*" && targetName[1] != "\*") {  //[fileName] = a.txt : 查找a.txt文件;**

**// 按文件名筛选文件;**

**if (foundName == fileName) foundFile += dirPath + foundName + "\n";**

**}**

**}**

**}**

**return foundFile;**

**}**

**// 获取文件类型**

**String getContentType(String filename) {**

**if (filename.endsWith(".htm"))**

**return "text/html";**

**else if (filename.endsWith(".html"))**

**return "text/html";**

**else if (filename.endsWith(".css"))**

**return "text/css";**

**else if (filename.endsWith(".js"))**

**return "application/javascript";**

**else if (filename.endsWith(".png"))**

**return "image/png";**

**else if (filename.endsWith(".gif"))**

**return "image/gif";**

**else if (filename.endsWith(".jpg"))**

**return "image/jpeg";**

**else if (filename.endsWith(".ico"))**

**return "image/x-icon";**

**else if (filename.endsWith(".xml"))**

**return "text/xml";**

**else if (filename.endsWith(".pdf"))**

**return "application/x-pdf";**

**else if (filename.endsWith(".zip"))**

**return "application/x-zip";**

**else if (filename.endsWith(".gz"))**

**return "application/x-gzip";**

**return "text/plain";**

**}**

**} FFileS;**

**// 时间控制类;**

**class TimeRefresh {**

**public:**

**// 系统的本地时间;**

**typedef struct SystemTime {**

**// Year Month Day Hour Minute Second**

**unsigned short year = 0;**

**unsigned char month = 0;**

**unsigned char day = 0;**

**unsigned char hour = 0;**

**unsigned char minute = 0;**

**unsigned char second = 0;**

**// 当前代码实现了对系统日期的更新;**

**void updateTime() {**

**second++;**

**if (second >= 60) {**

**second = 0;**

**minute++;**

**}**

**if (minute >= 60) {**

**minute = 0;**

**hour++;**

**}**

**if (hour >= 24) {**

**hour = 0;**

**day++;**

**}**

**unsigned char daysInMonth = 31;**

**if (month == 2) {**

**// 当month等于2时, 使用三目运算符，以计算当前年份是否为闰年。如果是闰年，则daysInMonth的值将设置为29；否则为28。**

**daysInMonth = (year % 400 == 0 || (year % 4 == 0 && year % 100 != 0)) ? 29 : 28;**

**} else if (month == 4 || month == 6 || month == 9 || month == 11) {**

**daysInMonth = 30;  // 对于4,6,9,11月只有30天,将daysInMonth的值设置为30;**

**}**

**// 如果天数大于当前月份的天数，代码将月份增加1，并在必要时将年份增加1;**

**if (day > daysInMonth) {**

**day = 1;**

**month++;**

**if (month > 12) {**

**month = 1;**

**year++;**

**}**

**}**

**}**

**// 解析网络时间的字符串, 并设置到系统时间;**

**void setSystemTime(String netWorkTimeStr) {**

**year = static\_cast<unsigned short>(netWorkTimeStr.substring(0, 4).toInt());**

**month = static\_cast<unsigned char>(netWorkTimeStr.substring(4, 6).toInt());**

**day = static\_cast<unsigned char>(netWorkTimeStr.substring(6, 8).toInt());**

**hour = static\_cast<unsigned char>(netWorkTimeStr.substring(8, 10).toInt());**

**minute = static\_cast<unsigned char>(netWorkTimeStr.substring(10, 12).toInt());**

**second = static\_cast<unsigned char>(netWorkTimeStr.substring(12, 14).toInt());**

**}**

**} SystemTime;**

**SystemTime sysTime;**

**String netWorkTimeStr = "";  // 储存从网络获取的时间(20230202135512);**

**bool allow = false;  // 允许时间刷新;**

**// 用于更新时间;**

**void begin() {**

**// 不触发警报的条件下每隔10min同步一次网络时间(多线程);**

**TimeRefresh\_ticker.attach(600, [this](void) -> void {**

**// 如果系统进入freezeMode(浅度睡眠)停止定时调用函数;**

**if (freezeMode == true) TimeRefresh\_ticker.detach();**

**if (digitalRead(SENOUT) == HIGH) allow = true;  // 授予获取网络时间许可证;**

**});**

**// 每隔1s更新一次系统本地日期和时间;**

**System\_time.attach(1, [this](void) -> void { sysTime.updateTime(); });**

**}**

**// 从授时网站获得时间**

**void getNetWorkTime() {**

**uint8 httpCode = http.GET();**

**if (httpCode > 0) {**

**if (httpCode == HTTP\_CODE\_OK) {**

**netWorkTimeStr = http.getString();  // 获取JSON字符串**

**// 解析JSON数据**

**StaticJsonDocument<200> doc;           // 创建一个StaticJsonDocument对象**

**deserializeJson(doc, netWorkTimeStr);  // 使用deserializeJson()函数来解析Json数据**

**netWorkTimeStr = doc["sysTime1"].as<String>();  // 读取JSON数据;**

**sysTime.setSystemTime(netWorkTimeStr);  // 解析网络时间的字符串, 并设置到系统时间;**

**// Serial.println(netWorkTimeStr);**

**} else {**

**Serial.printf("[HTTP GET Failed] ErrorCode: %s\n", http.errorToString(httpCode).c\_str());**

**}**

**} else {**

**Serial.printf("[HTTP GET Failed] ErrorCode: %s\n", http.errorToString(httpCode).c\_str());**

**}**

**http.end();**

**}**

**// 格式化时间(在个位数前添加0; 例如: 1 -> 01);**

**String format(unsigned char timeInt) {**

**if (timeInt < 10) {**

**return "0" + String(timeInt);**

**} else {**

**return String(timeInt);**

**}**

**}**

**// 读取时间(mode: true读取netWorkTimeStr\_Format的时间[时:分]; false读取netWorkTimeStr的时间[年月日时分秒]);**

**String timeRead(bool mode = true) {**

**if (mode == true) {**

**return format(sysTime.hour) + ":" + format(sysTime.minute);**

**} else {**

**return String(sysTime.year) + format(sysTime.month) + format(sysTime.day) + format(sysTime.hour) + format(sysTime.minute) +**

**format(sysTime.second);**

**}**

**}**

**} timeRef;**

**// 获取实时天气类;**

**class Weather {**

**public:**

**void beginWeather() {**

**// 读取Weather\_Config.ini文件(保存了私钥和位置), 以<PRIVATEKEY/LOCATION>分割字符串;**

**vector<String> WeatherConfig = oled.strsplit(FFileS.readFile("", "/Weather\_Config.ini"), "<PRIVATEKEY/LOCATION>");**

**// 配置心知天气请求信息**

**weatherNow.config(WeatherConfig[0], WeatherConfig[1], "c");**

**// 不触发警报的条件下每隔15min更新一次天气信息(多线程);**

**UpdateWeather.attach(900, [this](void) -> void {**

**// 如果系统进入freezeMode(浅度睡眠)停止定时调用函数;**

**if (freezeMode == true) UpdateWeather.detach();**

**if (digitalRead(SENOUT) == HIGH) updateWeather();**

**});**

**}**

**// 更新天气信息;**

**String updateWeather() {**

**if (!weatherNow.update()) {**

**return "Weather Update Fail: " + weatherNow.getServerCode();  // 更新失败;**

**} else {**

**return "";**

**}**

**}**

**// 设置城市ID;**

**String setCityID(String cityID) {**

**// 读取Weather\_Config.ini文件(保存了私钥和位置), 以<PRIVATEKEY/LOCATION>分割字符串;**

**vector<String> WeatherConfig = oled.strsplit(FFileS.readFile("", "/Weather\_Config.ini"), "<PRIVATEKEY/LOCATION>");**

**// 保持私钥不变, 覆盖原有的配置文件;**

**FFileS.fileCover(WeatherConfig[0] + "<PRIVATEKEY/LOCATION>" + cityID, "", "/Weather\_Config.ini");**

**// 配置心知天气请求信息**

**weatherNow.config(WeatherConfig[0], cityID, "c");**

**// 更新天气信息;**

**return updateWeather();**

**}**

**} weather;**

**typedef struct ANIM\_INDEX {**

**unsigned short name;      // 动画名称**

**unsigned short Duration;  // 动画播放时长;**

**unsigned short Begin;     // 动画播放起始帧;**

**unsigned short End;       // 动画播放结束帧;**

**unsigned char IMG\_Width;  // 动画帧宽度;**

**unsigned char IMG\_Hight;  // 动画帧高度;**

**} ANIM\_INDEX;**

**// 动画控制类(动画数组的最后一张用来清空动画显示区);**

**class Animation {**

**private:**

**ANIM\_INDEX Index;  // 当前选中的动画;**

**POINT Pos = {0, 0};                    // 动画播放的坐标(左上角为原点);**

**unsigned short Duration = UINT16\_MIN;  // 动画播放时长;**

**unsigned short Begin = UINT16\_MAX;     // 动画播放起始帧;**

**unsigned short End = UINT16\_MIN;       // 动画播放结束帧;**

**// 动画控制器(索引);**

**void AnimController() {**

**// 如果没有设置(自定义)动画播放参数则使用对应动画的默认参数;**

**if (Duration == UINT16\_MIN) Duration = Index.Duration;**

**if (Begin == UINT16\_MAX) Begin = Index.Begin;**

**if (End == UINT16\_MIN) End = Index.End;**

**// 根据播放时长计算每一帧的显示在屏幕上的时间(帧长度);**

**unsigned long Sleep\_ms = static\_cast<unsigned long>(static\_cast<float>(Duration) / static\_cast<float>(End - Begin));**

**// 根据索引内的动画名称播放指定动画;**

**if (Index.name == loading\_X16\_30F.name) {**

**for (unsigned short i = Begin; i < End; ++i) {**

**oled.OLED\_DrawBMP(Pos.x, Pos.y, Index.IMG\_Width, Index.IMG\_Hight, Loading\_X16\_30F[i]);**

**delay(Sleep\_ms);**

**}**

**}**

**if (Index.name == loading\_X16\_60F.name) {**

**for (unsigned short i = Begin; i < End; ++i) {**

**oled.OLED\_DrawBMP(Pos.x, Pos.y, Index.IMG\_Width, Index.IMG\_Hight, Loading\_X16\_60F[i]);**

**delay(Sleep\_ms);**

**}**

**}**

**if (Index.name == loadingBackForthBar\_60x8\_60F.name) {**

**for (unsigned short i = Begin; i < End; ++i) {**

**oled.OLED\_DrawBMP(Pos.x, Pos.y, Index.IMG\_Width, Index.IMG\_Hight, LoadingBackForthBar\_60x8\_60F[i]);**

**delay(Sleep\_ms);**

**}**

**}**

**if (Index.name == loadingBar\_60x8\_30F.name) {**

**for (unsigned short i = Begin; i < End; ++i) {**

**oled.OLED\_DrawBMP(Pos.x, Pos.y, Index.IMG\_Width, Index.IMG\_Hight, LoadingBar\_60x8\_30F[i]);**

**delay(Sleep\_ms);**

**}**

**}**

**if (Index.name == loadingBar\_60x8\_60F.name) {**

**for (unsigned short i = Begin; i < End; ++i) {**

**oled.OLED\_DrawBMP(Pos.x, Pos.y, Index.IMG\_Width, Index.IMG\_Hight, LoadingBar\_60x8\_60F[i]);**

**delay(Sleep\_ms);**

**}**

**}**

**//......**

**}**

**public:**

**// 在这里声明动画索引;**

**// ANIM\_INDEX{Name, Duration, Begin, End, IMG\_Width, IMG\_Hight};**

**ANIM\_INDEX loading\_X16\_30F = {0, 250, 0, 30, 16, 16};**

**ANIM\_INDEX loading\_X16\_60F = {1, 500, 0, 60, 16, 16};**

**ANIM\_INDEX loadingBackForthBar\_60x8\_60F = {2, 500, 0, 60, 60, 8};**

**ANIM\_INDEX loadingBar\_60x8\_30F = {3, 300, 0, 30, 60, 8};**

**ANIM\_INDEX loadingBar\_60x8\_60F = {4, 500, 0, 60, 60, 8};**

**//......**

**// 设置动画播放参数(坐标{x, y}, 播放时长[ms], 起始帧, 结束帧);**

**void setAnimation(u8 x, u8 y, u16 duration = UINT16\_MIN, u16 begin = UINT16\_MAX, u16 end = UINT16\_MIN) {**

**Pos = {x, y};**

**Duration = duration;**

**Begin = begin;**

**End = end;**

**}**

**// 开始播放指定动画(索引);**

**void runAnimation(ANIM\_INDEX index) {**

**Index = index;**

**AnimController();**

**}**

**} anim;**

**// OLED显示消防预警;**

**void ShowFireWarning() {**

**oled.OLED\_DrawBMP(0, 0, 32, 64, FireWarning\_32x64[0]);**

**oled.OLED\_DrawBMP(32, 0, 32, 64, FireWarning\_32x64[1]);**

**oled.OLED\_DrawBMP(64, 0, 32, 64, FireWarning\_32x64[2]);**

**oled.OLED\_DrawBMP(96, 0, 32, 64, FireWarning\_32x64[3]);**

**}**

**class Desktop {**

**private:**

**// 任务栏图标排序表(储存了任务栏图标的动态显示位置);**

**typedef struct StatusBars\_Ranked {**

**// 图标编号(对应oledfont.h中的编号);**

**unsigned char Clear\_Icon = 46;**

**unsigned char Charging = 0;**

**unsigned char WIFI = 1;**

**unsigned char ProgramDownload = 2;**

**unsigned char Disconnected = 3;**

**unsigned char Battery = 4;**

**unsigned char CMDCP = 5;**

**unsigned char Sunny\_Day\_0 = 6;**

**unsigned char Clear\_Night\_1 = 7;**

**unsigned char Sunny\_Day\_2 = 8;**

**unsigned char Clear\_Night\_3 = 9;**

**unsigned char Cloudy\_4 = 10;**

**unsigned char Partly\_Cloudy\_Day\_5 = 11;**

**unsigned char Partly\_Cloudy\_Night\_6 = 12;**

**unsigned char Mostly\_Cloudy\_Day\_7 = 13;**

**unsigned char Mostly\_Cloudy\_Night\_8 = 14;**

**unsigned char Overcast\_9 = 15;**

**unsigned char Shower\_10 = 16;**

**unsigned char Thundershower\_11 = 17;**

**unsigned char Thundershower\_with\_Hail\_12 = 18;**

**unsigned char Light\_Rain\_13 = 19;**

**unsigned char Moderate\_Rain\_14 = 20;**

**unsigned char Heavy\_Rain\_15 = 21;**

**unsigned char Storm\_16 = 22;**

**unsigned char Heavy\_Storm\_17 = 23;**

**unsigned char Severe\_Storm\_18 = 24;**

**unsigned char Ice\_Rain\_19 = 25;**

**unsigned char Sleet\_20 = 26;**

**unsigned char Snow\_Flurry\_21 = 27;**

**unsigned char Light\_Snow\_22 = 28;**

**unsigned char Moderate\_Snow\_23 = 29;**

**unsigned char Heavy\_Snow\_24 = 30;**

**unsigned char Snowstorm\_25 = 31;**

**unsigned char Dust\_26 = 32;**

**unsigned char Sand\_27 = 33;**

**unsigned char Duststorm\_28 = 34;**

**unsigned char Sandstorm\_29 = 35;**

**unsigned char Foggy\_30 = 36;**

**unsigned char Haze\_31 = 37;**

**unsigned char Windy\_32 = 38;**

**unsigned char Blustery\_33 = 39;**

**unsigned char Hurricane\_34 = 40;**

**unsigned char Tropical\_Storm\_35 = 41;**

**unsigned char Tornado\_36 = 42;**

**unsigned char Cold\_37 = 43;**

**unsigned char Hot\_38 = 44;**

**unsigned char Unknown\_99 = 45;**

**//......**

**unsigned char unit = 16;  // 图标显示单位(16x16图标);**

**vector<unsigned char> StatusBars\_Pos;  // 动态储存图标位置和注册状态, 图标在状态栏中的位置就是在vector中的位置, vector中的内容表示注册状态;**

**/\***

**// 图标注册标记状态(true:注册; false:注销);**

**bool Register\_State[64];**

**// 动态储存图标位置(初始位置设为-16);**

**int StatusBars\_Pos[8] = {-16, -16, -16, -16, -16, -16, -16, -16};**

**\*/**

**} StatusBars\_Ranked;**

**StatusBars\_Ranked SBR;**

**// 注册状态栏图标的位置(状态名称,模式[false:注销图标; true:注册图标]);**

**void Icon\_Register(unsigned char name, bool mode) {**

**/\***

**if (mode == false && SBR.Register\_State[name] == true) {  // 在状态栏注销图标;**

**SBR.Register\_State[name] = false;                     // 标记注销图标;**

**// 遍历所有图标的位置,将注销图标右方的所有图标左移一个图标显示单位;**

**for (auto& i : SBR.StatusBars\_Pos)**

**if (i > SBR.StatusBars\_Pos[name]) i -= SBR.unit;**

**// 清空注销图标(包括注销图标)右方的区域;**

**for (unsigned char i = SBR.StatusBars\_Pos[name]; i <= 112; i += SBR.unit) oled.OLED\_DrawBMP(i, 6, 16, 16, StatusBars[SBR.Clear\_Icon]);**

**SBR.StatusBars\_Pos[name] = 0;  // 将注销图标的位置归零;**

**} else if (mode == true && SBR.Register\_State[name] == false) {  // 在状态栏注册图标**

**SBR.Register\_State[name] = true;                             // 标记注册图标;**

**// 分配注册图标的位置: 注册图标的位置 = 所有图标位置的最大值 + 一个图标显示单位;**

**SBR.StatusBars\_Pos[name] = tool.findArrMax(SBR.StatusBars\_Pos, 8) + SBR.unit;**

**}**

**\*/**

**short Register\_Pos = -16;  // 将注册位置初始化为-16;**

**// 查找vector中"name"的下标, 下标位置就是注册位置(注册位置\*16就是图标在显示屏中的位置);**

**auto it = std::find(SBR.StatusBars\_Pos.begin(), SBR.StatusBars\_Pos.end(), name);**

**if (it != SBR.StatusBars\_Pos.end()) {**

**Register\_Pos = static\_cast<short>(std::distance(SBR.StatusBars\_Pos.begin(), it));**

**}**

**if (mode == false && Register\_Pos != -16) {  // 在状态栏注销图标;**

**// 清空注销图标(包括注销图标)右方的区域;**

**for (unsigned char i = Register\_Pos \* SBR.unit; i <= 112; i += SBR.unit) oled.OLED\_DrawBMP(i, 6, 16, 16, StatusBars[SBR.Clear\_Icon]);**

**// remove 函数在 vector 中删除所有与 name 相等的元素，erase 函数删除 vector 中剩余的空元素(注销图标)。**

**SBR.StatusBars\_Pos.erase(remove(SBR.StatusBars\_Pos.begin(), SBR.StatusBars\_Pos.end(), name), SBR.StatusBars\_Pos.end());**

**} else if (mode == true && Register\_Pos == -16) {  // 在状态栏注册图标**

**SBR.StatusBars\_Pos.push\_back(name);            // 在状态栏的最后添加新注册的图标;**

**Register\_Pos = SBR.StatusBars\_Pos.size() - 1;  // 获取最后一个图标位置(即新注册图标的位置);**

**}**

**if (mode == true) {**

**oled.OLED\_DrawBMP(Register\_Pos \* SBR.unit, 6, 16, 16, StatusBars[name]);  // 显示图标;**

**}**

**}**

**// 渲染状态栏(注意多线程使用循环可能会出问题);**

**void StatusBars\_Render() {**

**// 正在充电状态;**

**if (Charging\_State == true) {**

**Icon\_Register(SBR.Charging, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Charging, false);  // 注销图标位置;**

**}**

**// WIFI连接状态;**

**if (WIFI\_State == true) {**

**Icon\_Register(SBR.WIFI, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.WIFI, false);  // 注销图标位置;**

**}**

**// 程序下载模式状态;**

**if (digitalRead(0) == LOW) {**

**Icon\_Register(SBR.ProgramDownload, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.ProgramDownload, false);  // 注销图标位置;**

**}**

**// WIFI断开状态;**

**if (WIFI\_State == false) {**

**Icon\_Register(SBR.Disconnected, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Disconnected, false);  // 注销图标位置;**

**}**

**// 停止充电状态;**

**if (Charging\_State == false) {**

**Icon\_Register(SBR.Battery, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Battery, false);  // 注销图标位置;**

**}**

**// CMDCP被打开;**

**if (CMDCP\_State == true) {**

**Icon\_Register(SBR.CMDCP, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.CMDCP, false);  // 注销图标位置;**

**}**

**if (weatherNow.getWeatherCode() == 99) {**

**Icon\_Register(SBR.Unknown\_99, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Unknown\_99, false);  // 注销图标位置;**

**}**

**for (unsigned char i = SBR.Sunny\_Day\_0; i <= SBR.Hot\_38; ++i) {**

**if (weatherNow.getWeatherCode() == (i - SBR.Sunny\_Day\_0)) {**

**Icon\_Register(i, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(i, false);  // 注销图标位置;**

**}**

**}**

**/\***

**// 晴（国内城市白天晴）;**

**if (weather.weatherCode == 0) {**

**Icon\_Register(SBR.Sunny\_Day\_0, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Sunny\_Day\_0, false);  // 注销图标位置;**

**}**

**// 晴（国内城市夜晚晴）;**

**if (weather.weatherCode == 1) {**

**Icon\_Register(SBR.Clear\_Night\_1, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Clear\_Night\_1, false);  // 注销图标位置;**

**}**

**// 晴（国外城市白天晴）;**

**if (weather.weatherCode == 2) {**

**Icon\_Register(SBR.Sunny\_Day\_2, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Sunny\_Day\_2, false);  // 注销图标位置;**

**}**

**// 晴（国外城市夜晚晴）;**

**if (weather.weatherCode == 3) {**

**Icon\_Register(SBR.Clear\_Night\_3, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Clear\_Night\_3, false);  // 注销图标位置;**

**}**

**// 多云;**

**if (weather.weatherCode == 4) {**

**Icon\_Register(SBR.Cloudy\_4, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Cloudy\_4, false);  // 注销图标位置;**

**}**

**// 晴间多云(日);**

**if (weather.weatherCode == 5) {**

**Icon\_Register(SBR.Partly\_Cloudy\_Day\_5, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Partly\_Cloudy\_Day\_5, false);  // 注销图标位置;**

**}**

**// 晴间多云(夜);**

**if (weather.weatherCode == 6) {**

**Icon\_Register(SBR.Partly\_Cloudy\_Night\_6, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Partly\_Cloudy\_Night\_6, false);  // 注销图标位置;**

**}**

**// 大部多云(日);**

**if (weather.weatherCode == 7) {**

**Icon\_Register(SBR.Mostly\_Cloudy\_Day\_7, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Mostly\_Cloudy\_Day\_7, false);  // 注销图标位置;**

**}**

**// 大部多云(夜);**

**if (weather.weatherCode == 8) {**

**Icon\_Register(SBR.Mostly\_Cloudy\_Night\_8, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Mostly\_Cloudy\_Night\_8, false);  // 注销图标位置;**

**}**

**// 阴;**

**if (weather.weatherCode == 9) {**

**Icon\_Register(SBR.Overcast\_9, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Overcast\_9, false);  // 注销图标位置;**

**}**

**// 阵雨;**

**if (weather.weatherCode == 10) {**

**Icon\_Register(SBR.Shower\_10, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Shower\_10, false);  // 注销图标位置;**

**}**

**// 雷阵雨;**

**if (weather.weatherCode == 11) {**

**Icon\_Register(SBR.Thundershower\_11, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Thundershower\_11, false);  // 注销图标位置;**

**}**

**// 雷阵雨伴有冰雹;**

**if (weather.weatherCode == 12) {**

**Icon\_Register(SBR.Thundershower\_with\_Hail\_12, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Thundershower\_with\_Hail\_12, false);  // 注销图标位置;**

**}**

**// 小雨;**

**if (weather.weatherCode == 13) {**

**Icon\_Register(SBR.Light\_Rain\_13, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Light\_Rain\_13, false);  // 注销图标位置;**

**}**

**// 中雨;**

**if (weather.weatherCode == 14) {**

**Icon\_Register(SBR.Moderate\_Rain\_14, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Moderate\_Rain\_14, false);  // 注销图标位置;**

**}**

**// 大雨;**

**if (weather.weatherCode == 15) {**

**Icon\_Register(SBR.Heavy\_Rain\_15, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Heavy\_Rain\_15, false);  // 注销图标位置;**

**}**

**// 暴雨;**

**if (weather.weatherCode == 16) {**

**Icon\_Register(SBR.Storm\_16, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Storm\_16, false);  // 注销图标位置;**

**}**

**// 大暴雨;**

**if (weather.weatherCode == 17) {**

**Icon\_Register(SBR.Heavy\_Storm\_17, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Heavy\_Storm\_17, false);  // 注销图标位置;**

**}**

**// 特大暴雨;**

**if (weather.weatherCode == 18) {**

**Icon\_Register(SBR.Severe\_Storm\_18, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Severe\_Storm\_18, false);  // 注销图标位置;**

**}**

**// 冻雨;**

**if (weather.weatherCode == 19) {**

**Icon\_Register(SBR.Ice\_Rain\_19, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Ice\_Rain\_19, false);  // 注销图标位置;**

**}**

**// 雨夹雪;**

**if (weather.weatherCode == 20) {**

**Icon\_Register(SBR.Sleet\_20, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Sleet\_20, false);  // 注销图标位置;**

**}**

**// 阵雪;**

**if (weather.weatherCode == 21) {**

**Icon\_Register(SBR.Snow\_Flurry\_21, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Snow\_Flurry\_21, false);  // 注销图标位置;**

**}**

**// 小雪;**

**if (weather.weatherCode == 22) {**

**Icon\_Register(SBR.Light\_Snow\_22, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Light\_Snow\_22, false);  // 注销图标位置;**

**}**

**// 中雪;**

**if (weather.weatherCode == 23) {**

**Icon\_Register(SBR.Moderate\_Snow\_23, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Moderate\_Snow\_23, false);  // 注销图标位置;**

**}**

**// 大雪;**

**if (weather.weatherCode == 24) {**

**Icon\_Register(SBR.Heavy\_Snow\_24, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Heavy\_Snow\_24, false);  // 注销图标位置;**

**}**

**// 暴雪;**

**if (weather.weatherCode == 25) {**

**Icon\_Register(SBR.Snowstorm\_25, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Snowstorm\_25, false);  // 注销图标位置;**

**}**

**// 浮尘;**

**if (weather.weatherCode == 26) {**

**Icon\_Register(SBR.Dust\_26, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Dust\_26, false);  // 注销图标位置;**

**}**

**// 扬沙;**

**if (weather.weatherCode == 27) {**

**Icon\_Register(SBR.Sand\_27, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Sand\_27, false);  // 注销图标位置;**

**}**

**// 沙尘暴;**

**if (weather.weatherCode == 28) {**

**Icon\_Register(SBR.Duststorm\_28, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Duststorm\_28, false);  // 注销图标位置;**

**}**

**// 强沙尘暴;**

**if (weather.weatherCode == 29) {**

**Icon\_Register(SBR.Sandstorm\_29, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Sandstorm\_29, false);  // 注销图标位置;**

**}**

**// 雾;**

**if (weather.weatherCode == 30) {**

**Icon\_Register(SBR.Foggy\_30, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Foggy\_30, false);  // 注销图标位置;**

**}**

**// 霾;**

**if (weather.weatherCode == 31) {**

**Icon\_Register(SBR.Haze\_31, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Haze\_31, false);  // 注销图标位置;**

**}**

**// 风;**

**if (weather.weatherCode == 32) {**

**Icon\_Register(SBR.Windy\_32, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Windy\_32, false);  // 注销图标位置;**

**}**

**// 大风;**

**if (weather.weatherCode == 33) {**

**Icon\_Register(SBR.Blustery\_33, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Blustery\_33, false);  // 注销图标位置;**

**}**

**// 飓风;**

**if (weather.weatherCode == 34) {**

**Icon\_Register(SBR.Hurricane\_34, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Hurricane\_34, false);  // 注销图标位置;**

**}**

**// 热带风暴;**

**if (weather.weatherCode == 35) {**

**Icon\_Register(SBR.Tropical\_Storm\_35, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Tropical\_Storm\_35, false);  // 注销图标位置;**

**}**

**// 龙卷风;**

**if (weather.weatherCode == 36) {**

**Icon\_Register(SBR.Tornado\_36, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Tornado\_36, false);  // 注销图标位置;**

**}**

**// 冷;**

**if (weather.weatherCode == 37) {**

**Icon\_Register(SBR.Cold\_37, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Cold\_37, false);  // 注销图标位置;**

**}**

**// 热;**

**if (weather.weatherCode == 38) {**

**Icon\_Register(SBR.Hot\_38, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Hot\_38, false);  // 注销图标位置;**

**}**

**// 未知;**

**if (weather.weatherCode == 99) {**

**Icon\_Register(SBR.Unknown\_99, true);  // 注册图标位置;**

**} else {**

**Icon\_Register(SBR.Unknown\_99, false);  // 注销图标位置;**

**}**

**\*/**

**//......**

**}**

**public:**

**void begin() {**

**// 不触发警报的条件下每隔500ms刷新一次状态栏(多线程);**

**Desktop\_ticker.attach\_ms(500, [this](void) -> void {**

**// 如果系统进入freezeMode(浅度睡眠)停止定时调用函数;**

**if (freezeMode == true) Desktop\_ticker.detach();**

**// 如果处于开发者模式则停止刷新状态栏;**

**if (digitalRead(SENOUT) == HIGH && Developer\_Mode == false) {**

**// 如果每分钟中秒数走到0(即开头)则刷新一次桌面时钟;**

**if (timeRef.sysTime.second == 0) oled.OLED\_ShowString(4, -1, timeRef.timeRead().c\_str(), 49);  // 刷新时间;**

**StatusBars\_Render();  // 刷新状态栏;**

**}**

**});**

**}**

**// 渲染主桌面;**

**void Main\_Desktop() {**

**oled.OLED\_ShowString(4, -1, timeRef.timeRead().c\_str(), 49);  // 刷新时间;**

**StatusBars\_Render();                                          // 渲染状态栏;**

**}**

**} Desktop;**

**// 判断电池是否进入充电并显示电池开始充电的信息;**

**void Show\_Charging\_info() {**

**if (digitalRead(CHRG) == LOW && Charging\_State == false) {**

**Charging\_State = true;  // 充电状态设为true;**

**// 播放动画等待电压稳定(电压不稳就操作大功耗器件有可能造成MCU复位);**

**anim.setAnimation(112, 6, 400);                                      // 设置加载动画显示位置,设置播放时长为400ms;**

**for (u8 i = 0; i < 2; ++i) anim.runAnimation(anim.loading\_X16\_60F);  // 播放加载动画(两次);**

**oled.OLED\_DrawBMP(112, 6, 16, 16, Loading\_X16\_60F[60]);              // 清空动画播放区;**

**/\***

**delay(1000);                                     // 等待电压稳定(电压不稳就操作大功耗器件有可能造成MCU复位);**

**oled.OLED\_DrawBMP(0, 0, 128, 64, Charging\_IMG);  // 显示正在充电提示;**

**delay(1000);**

**oled.OLED\_Clear();  // 清除界面**

**Main\_Desktop();     // 渲染主桌面;**

**\*/**

**} else if (digitalRead(CHRG) == HIGH && Charging\_State == true) {**

**Charging\_State = false;  // 充电状态设为false;**

**}**

**}**

**// 程序下载模式(下载程序必须下拉GPIO0并且复位);**

**void ProgramDownloadMode() {**

**// 条件: GPIO0下拉, 允许进入下载模式, 不处于浅度休眠模式;**

**if (digitalRead(Decoder\_C) == LOW && allowDownloadMode == true && freezeMode == false) {**

**bool allowReset = true;**

**// 显示主桌面提示即将进入下载模式;**

**oled.OLED\_DrawBMP(0, 0, 128, 48, DownloadMode\_IMG);**

**// 设置加载动画显示位置,设置播放时长为400ms;**

**anim.setAnimation(112, 6, 400);**

**// 等待2400ms, 反悔时间约3s内断开GPIO0的下拉, 系统会进入开发者模式;**

**for (u8 i = 0; i < 6; ++i) {**

**anim.runAnimation(anim.loading\_X16\_60F);  // 播放等待动画;**

**if (digitalRead(0) == HIGH) {**

**oled.OLED\_DrawBMP(112, 6, 16, 16, Loading\_X16\_60F[60]);  // 清空动画播放区;**

**allowReset = false;                                      // 不允许复位进入下载模式;**

**break;**

**}**

**}**

**if (allowReset == true) {**

**oled.OLED\_Display\_Off();  // 关闭OLED显示屏;**

**// LittleFS.format();        // 格式化闪存文件系统;**

**digitalWrite(RST, LOW);  // 复位进入下载模式;**

**}**

**}**

**}**

**class SystemSleep {**

**private:**

**public:**

**// 读取主要GPIO管脚的状态;**

**String GPIO\_Read() {**

**// 首先读取译码器控制引脚的状态;**

**int decoderC = digitalRead(Decoder\_C);**

**int decoderB = digitalRead(Decoder\_B);**

**int decoderA = digitalRead(Decoder\_A);**

**// 解释译码器引脚的状态;**

**String decodedWith = "\nDecoded\_with=";**

**switch (decoderC << 2 | decoderB << 1 | decoderA) {**

**case 1:**

**decodedWith += "Buzzer\_Enable";**

**break;**

**case 2:**

**decodedWith += "RedLED\_Enable";**

**break;**

**case 3:**

**decodedWith += "GreenLED\_Enable";**

**break;**

**case 4:**

**decodedWith += "BlueLED\_Enable";**

**break;**

**case 6:**

**decodedWith += "Sensor\_and\_OLED\_disabled";**

**break;**

**default:**

**decodedWith += "NULL";**

**break;**

**}**

**// 其余引脚的状态直接读出;**

**String GPIO\_State = "RST=" + String(digitalRead(RST)) + "\nTXD=" + String(digitalRead(TXD)) + "\nRXD=" + String(digitalRead(RXD)) +**

**"\nSCL=" + String(digitalRead(SCL)) + "\nSDA=" + String(digitalRead(SDA)) + "\nCHRG=" + String(digitalRead(CHRG)) +**

**"\nLOWPOWER=" + String(digitalRead(LOWPOWER)) + "\nSENOUT=" + String(digitalRead(SENOUT)) + "\nDecoder\_C=" + String(decoderC) +**

**"\nDecoder\_B=" + String(decoderB) + "\nDecoder\_A=" + String(decoderA) + decodedWith;**

**return GPIO\_State;**

**}**

**// 获取系统模式和状态;**

**String getSysModeAndStatus() {**

**return "Charging\_State=" + String(Charging\_State) + "\nWIFI\_State=" + String(WIFI\_State) + "\nDeveloper\_Mode=" + String(Developer\_Mode) +**

**"\nallowResponse=" + String(allowResponse) + "\nallowDownloadMode=" + String(allowDownloadMode) + "\nfreezeMode=" + String(freezeMode) +**

**"\ndiskMode=" + String(diskMode);**

**}**

**// 删除diskMode深度睡眠缓存的数据文件;**

**void removeSleepFile() {**

**// 如果存放休眠文件的目录存在则删除这个目录;**

**if (LittleFS.exists("/SleepFile")) FFileS.removeDirector("", "/SleepFile");**

**}**

**// 从深度睡眠diskMode恢复系统;**

**void resumeFromDeepSleep() {**

**LittleFS.begin();       // 启动闪存文件系统**

**String SleepFile = "";  // 休眠文件字符串结构: String("Name1="+"Status1"+"\nName2="+"Status2"+"\nName3="+"Status3"......)**

**// 如果[GPIO\_Status]休眠文件存在则从该文件恢复系统;**

**if (LittleFS.exists("/SleepFile/GPIO\_Status.txt")) {**

**// [GPIO\_Status]-恢复深度睡眠前输出GPIO的状态;**

**SleepFile = FFileS.readFile("", "/SleepFile/GPIO\_Status.txt");**

**// 以换行符分割字符串;**

**for (auto& i : oled.strsplit(SleepFile, "\n")) {**

**vector<String> GPIO\_Status = oled.strsplit(i, "=");  // 以等于符分割字符串;**

**// 恢复译码器控制引脚的状态(恢复输出GPIO的状态);**

**if (GPIO\_Status[0] == "Decoder\_C") digitalWrite(Decoder\_C, GPIO\_Status[1].toInt());**

**if (GPIO\_Status[0] == "Decoder\_B") digitalWrite(Decoder\_B, GPIO\_Status[1].toInt());**

**if (GPIO\_Status[0] == "Decoder\_A") digitalWrite(Decoder\_A, GPIO\_Status[1].toInt());**

**}**

**FFileS.removeFile("", "/SleepFile/GPIO\_Status.txt");  // 恢复完成后删除[GPIO\_Status]深度休眠文件;**

**}**

**// 如果[系统模式和状态]休眠文件存在则从该文件恢复系统;**

**if (LittleFS.exists("/SleepFile/SysModeAndStatus.txt")) {**

**// [System Mode&Status(系统模式和状态)]-恢复深度睡眠前系统模式和状态;**

**SleepFile = FFileS.readFile("", "/SleepFile/SysModeAndStatus.txt");**

**// 以换行符分割字符串;**

**for (auto& i : oled.strsplit(SleepFile, "\n")) {**

**vector<String> SysModeAndStatus = oled.strsplit(i, "=");  // 以等于符分割字符串;**

**// if (SysModeAndStatus[0] == "Charging\_State") Charging\_State = (SysModeAndStatus[1].toInt() != 0);//实时检测不需要恢复;**

**// if (SysModeAndStatus[0] == "WIFI\_State") WIFI\_State = (SysModeAndStatus[1].toInt() != 0);//实时检测不需要恢复;**

**if (SysModeAndStatus[0] == "Developer\_Mode") Developer\_Mode = (SysModeAndStatus[1].toInt() != 0);  // 恢复"开发者模式"的设置;**

**if (SysModeAndStatus[0] == "allowResponse") allowResponse = (SysModeAndStatus[1].toInt() != 0);  // 恢复"允许服务器对客户端进行响应"的设置;**

**if (SysModeAndStatus[0] == "allowDownloadMode") allowDownloadMode = (SysModeAndStatus[1].toInt() != 0);  // 恢复"允许进入下载模式"的设置;**

**if (SysModeAndStatus[0] == "freezeMode") freezeMode = (SysModeAndStatus[1].toInt() != 0);                // 恢复"浅度睡眠"设置;**

**if (SysModeAndStatus[0] == "diskMode") diskMode = (SysModeAndStatus[1].toInt() != 0);                    // 恢复"深度睡眠"设置;**

**}**

**FFileS.removeFile("", "/SleepFile/SysModeAndStatus.txt");  // 恢复完成后删除[系统模式和状态]深度休眠文件;**

**}**

**diskMode = false;  // 禁用深度睡眠;**

**}**

**//[休眠模式-freeze], 冻结I/O设备, 关闭外设, ESP-12F进入Modem-sleep模式, 程序上只运行CMDControlPanel网络服务, 其他服务冻结;**

**void Sys\_freezeMode(bool Enable = true) {**

**if (Enable == true) {**

**freezeMode = true;  // 启用浅度睡眠;**

**oled.OLED\_Display\_Off();  // OLED显示屏停止显示;**

**// 关闭所有传感器(红外&气体),关闭OLED屏幕,关闭所有声光警报,切断除MCU外的一切供电;**

**digitalWrite(Decoder\_C, HIGH);**

**digitalWrite(Decoder\_B, HIGH);**

**digitalWrite(Decoder\_A, LOW);**

**WiFi.setSleepMode(WIFI\_MODEM\_SLEEP);  // ESP-12F进入Modem-sleep模式;**

**} else if (Enable == false) {**

**WiFi.setSleepMode(WIFI\_NONE\_SLEEP);  // ESP-12F离开睡眠模式;**

**freezeMode = false;  // 禁用浅度睡眠;**

**// 重新给所有传感器(红外&气体), OLED屏幕, 声光警报器上电;**

**digitalWrite(Decoder\_C, HIGH);**

**digitalWrite(Decoder\_B, HIGH);**

**digitalWrite(Decoder\_A, HIGH);**

**// 大功率器件上电可能会造成局部电压波动, 等待一段时间至电压稳定(最少等待1s, 最多等待10s);**

**for (uint8 i = 0; i < 10; ++i) {**

**delay(1000);**

**if (digitalRead(SENOUT) == HIGH) break;**

**}**

**oled.OLED\_Init();          // 初始化OLED**

**oled.OLED\_ColorTurn(0);    // 0正常显示 1反色显示**

**oled.OLED\_DisplayTurn(0);  // 0正常显示 1翻转180度显示**

**timeRef.getNetWorkTime();  // 获取网络时间;**

**oled.OLED\_Clear();         // 清除界面**

**Desktop.Main\_Desktop();    // 渲染主桌面;**

**timeRef.begin();  // 恢复时间刷新服务;**

**Desktop.begin();  // 恢复桌面刷新服务;**

**}**

**}**

**// disk [sleep time\_us];**

**//[深度休眠模式-disk] 运行状态(GPIO\_Status, 系统模式和状态, 文本框信息)数据存到Flash(醒来时恢复状态), 然后ESP12F进入深度睡眠;**

**void Sys\_diskMode(uint64\_t time\_us = 0) {**

**// 登出和锁定CMDCP;**

**CMDCP\_State = false;     // 用户关闭CMDCP;**

**Developer\_Mode = false;  // 退出开发者模式(显示状态栏和桌面时钟);**

**diskMode = true;   // 启用深度睡眠;**

**LittleFS.begin();  // 启动闪存文件系统**

**// [GPIO\_Status]-保存进入深度睡眠前的GPIO状态到Flash, 以便从深度睡眠醒来时恢复GPIO状态;**

**File GPIO\_StatusFile = LittleFS.open("/SleepFile/GPIO\_Status.txt", "w");  // 创建&覆盖并打开GPIO\_Status.txt文件;**

**GPIO\_StatusFile.print(GPIO\_Read());                                       // 向GPIO\_StatusFile写入GPIO状态信息;**

**GPIO\_StatusFile.close();                                                  // 完成文件写入后关闭文件;**

**//[System Mode&Status(系统模式和状态)]-保存系统现在处于的系统模式和状态到Flash, 以便从深度睡眠醒来时恢复;**

**File SysModeAndStatusFile = LittleFS.open("/SleepFile/SysModeAndStatus.txt", "w");  // 创建&覆盖并打开SysModeAndStatus.txt文件;**

**SysModeAndStatusFile.print(getSysModeAndStatus());  // 向SysModeAndStatusFile写入当前系统模式和状态状态信息;**

**SysModeAndStatusFile.close();                       // 完成文件写入后关闭文件;**

**// [PrintBox(文本框)]-保存进入深度睡眠前的OLED上输出的文本信息到Flash;**

**// 文本框休眠文件的数据结构: String("OLED屏幕第1行"+"\n"+"OLED屏幕第2行"+"\n"+"OLED屏幕第3行"+......)**

**File PrintBoxFile = LittleFS.open("/SleepFile/PrintBox.txt", "w");  // 创建&覆盖并打开PrintBox.txt文件;**

**for (auto& i : oled.getPrintBox()) {**

**PrintBoxFile.print(i + "\n");  // 向PrintBoxFile写入OLED屏幕打印的文本信息, 在OLED屏幕上的每行字符串的末尾追加"\n"后把所有行合并;**

**}**

**PrintBoxFile.close();  // 完成文件写入后关闭文件;**

**ESP.deepSleep(time\_us);  // ESP-12F进入深度睡眠;**

**}**

**} SysSleep;**

**class WebServer {**

**private:**

**File UploadFile;  // 建立文件对象用于文件上传至服务器闪存;**

**bool routeUploadEnabled = false;**

**public:**

**String UploadRespond = "";  // 用于回复终端文件是否上次成功;**

**// 检查WIFI是否连接,若没有连接则连接;**

**void WiFi\_Connect() {**

**// 读取时间数据(从RAM)如果数据为"00:00"则表示系统正在启动, 否则表示系统正常运行时需要确认WIFI连接正常(两种情况播放的动画不同);**

**if (timeRef.timeRead() == "00:00")**

**anim.setAnimation(67, 6);  // 设置动画播放位置(其他参数默认);**

**else**

**anim.setAnimation(112, 6);  // 设置动画播放位置(其他参数默认);**

**if (WiFi.status() != WL\_CONNECTED) {**

**WIFI\_State = false;**

**// 读取WIFI\_Config.ini文件(保存了WIFI名称和密码), 以<SSID/PASSWD>分割字符串;**

**vector<String> SSID\_PASSWD = oled.strsplit(FFileS.readFile("", "/WIFI\_Config.ini"), "<SSID/PASSWD>");**

**// WiFi.begin(SSID\_PASSWD[0], SSID\_PASSWD[1]);**

**WiFi.begin(SSID, PASSWORD);**

**// 等待WIFI连接(超时时间为10s);**

**for (unsigned char i = 0; i < 100; ++i) {**

**if (WiFi.status() == WL\_CONNECTED) {**

**Serial.print("IP Address: ");**

**Serial.print(WiFi.localIP());**

**Serial.println(":" + String(ServerPort));**

**// WIFI连接完成后清空动画播放区域;**

**//  读取时间数据(从RAM)如果数据为"00:00"则表示系统正在启动, 否则表示系统正常运行时需要确认WIFI连接正常(两种情况播放的动画不同);**

**if (timeRef.timeRead() == "00:00")**

**// 清空进度条加载动画区域;**

**oled.OLED\_DrawBMP(67, 6, anim.loadingBar\_60x8\_30F.IMG\_Width, anim.loadingBar\_60x8\_30F.IMG\_Hight, LoadingBar\_60x8\_30F[30]);**

**else**

**// 清空加载动画区域;**

**oled.OLED\_DrawBMP(112, 6, anim.loading\_X16\_60F.IMG\_Width, anim.loading\_X16\_60F.IMG\_Hight, Loading\_X16\_60F[30]);**

**WIFI\_State = true;**

**break;**

**} else {**

**// 读取时间数据(从RAM)如果数据为"00:00"则表示系统正在启动, 否则表示系统正常运行时需要确认WIFI连接正常(两种情况播放的动画不同);**

**if (timeRef.timeRead() == "00:00")**

**anim.runAnimation(anim.loadingBar\_60x8\_30F);  // 播放进度条加载动画;**

**else**

**anim.runAnimation(anim.loading\_X16\_60F);  // 播放加载动画;**

**}**

**}**

**} else {**

**WIFI\_State = true;**

**}**

**}**

**// 配置MQTT;**

**void MQTT\_Begin() {**

**// 设置MQTT服务器**

**mqtt\_client.setServer(MQTT\_SERVER, 1883);**

**// 一定要设置keepAlive time为较大值，默认值15会无法建立连接，推荐60**

**mqtt\_client.setKeepAlive(60);**

**MQTT\_Client();  // 运行MQTT客户端;**

**}**

**// MQTT客户端;**

**void MQTT\_Client() {**

**if (mqtt\_client.connected()) {  // 如果开发板成功连接服务器**

**mqtt\_client.loop();         // 处理信息以及心跳**

**} else {                        // 如果开发板未能成功连接服务器**

**// 则尝试连接服务器**

**// 连接MQTT服务器**

**if (mqtt\_client.connect(MQTT\_CLIENT\_ID, MQTT\_USRNAME, MQTT\_PASSWD)) {**

**Serial.println("MQTT Server Connected.");**

**Serial.println("Server Address: ");**

**Serial.println(MQTT\_SERVER);**

**Serial.println("ClientId:");**

**Serial.println(MQTT\_CLIENT\_ID);**

**} else {**

**Serial.print("MQTT Server Connect Failed. Client State:");**

**Serial.println(mqtt\_client.state());**

**delay(2000);**

**}**

**}**

**}**

**bool getRouteUploadStatus() { return routeUploadEnabled; }**

**// 处理上传文件函数(用于将终端文件上传到服务器Flash);**

**void handleFileUpload() {**

**HTTPUpload& upload = server.upload();**

**if (upload.status == UPLOAD\_FILE\_START) {                           // 如果上传状态为UPLOAD\_FILE\_START**

**String filepath = FFileS.getWorkDirectory() + upload.filename;  // 建立字符串变量用于存放上传文件路径**

**UploadFile = LittleFS.open(filepath, "w");  // 在LittleFS中建立文件用于写入用户上传的文件数据**

**} else if (upload.status == UPLOAD\_FILE\_WRITE) {                       // 如果上传状态为UPLOAD\_FILE\_WRITE**

**if (UploadFile) UploadFile.write(upload.buf, upload.currentSize);  // 向LittleFS文件写入浏览器发来的文件数据**

**} else if (upload.status == UPLOAD\_FILE\_END) {                                                       // 如果上传状态为UPLOAD\_FILE\_END**

**if (UploadFile) {                                                                                // 如果文件成功建立**

**UploadFile.close();                                                                          // 将文件关闭**

**UploadRespond = "Size: " + String(upload.totalSize) + " Byte" + "\nFile upload succeeded.";  // 返回完成信息;**

**} else {                                                                                         // 如果文件未能成功建立**

**UploadRespond = "File upload failed.";                                                       // 返回错误信息;**

**}**

**}**

**}**

**String uploadFile() {**

**UploadRespond = "";         // 清空响应字符串;**

**routeUploadEnabled = true;  // 授予"Upload"路由开启许可, 打开上传通道, 允许文件上传;**

**// 发送"上传许可"通知, 告诉客户端服务器已就绪可以上传文件;**

**server.send(200, "text/plain", "EnableUpload");**

**// 接收文件流;**

**while (true) {**

**server.handleClient();**

**// 如果"UploadRespond"有字符串, 则表示文件已经上传完成(或者失败);**

**if (UploadRespond != "") {**

**// 关闭"/upload"路由(文件上传通道关闭);**

**routeUploadEnabled = false;**

**return UploadRespond;**

**}**

**}**

**}**

**} WebServer;**

**class CMDControlPanel {**

**private:**

**String CMD = "";           // 用来缓存CMD指令;**

**String PassWord = "";      // 使用CMDControlPanel前需要输入密码(系统启动时将flash中的密码缓存到此);**

**bool LockerState = false;  // PassLocker为true时表示CMD已经解锁, 解锁状态将一直保存在RAM中直到MCU断电或用户通过命令吊销;**

**String CMDCP\_Online\_Response = "";  // 临时储存CMD的响应数据;**

**vector<String> clientLogedIP;       // 这里储存了已登录并且正在使用还未登出CMD用户的IP地址(登出时删除IP);**

**// 使用CMDControlPanel前需要输入密码(无密码会要求用户设置密码, 密码正确返回true, 密码错误返回false);**

**bool PassLocker() {**

**allowResponse = false;  // 禁止服务器对客户端进行响应;**

**CMD = "";               // 清空CMD缓存;**

**String PassWord\_Temp1 = "", PassWord\_Temp2 = "";**

**LittleFS.begin();  // 启动闪存文件系统;**

**// 判断Flash中储存密匙的文件是否存在, 如果不存在则要求用户设置密匙, 如果存在则要用户输入密匙;**

**if (LittleFS.exists("/PassWord.txt") == false) {**

**// for循环决定了超时时间, 如果什么都不做则90s后超时;**

**for (unsigned char j = 0; j < 3; ++j) {**

**CMDCP\_Response("Set password:");**

**server.send(200, "text/html", CMDCP\_Online\_Response);  // 向客户端发送请设置密码的字符串(主动发送不是响应);**

**// 等待30s, 期间可以输入第一次密码;**

**for (unsigned char i = 0; i < 60; ++i) {**

**// SerialReceived();       // 获取串口或WIFI数据;**

**server.handleClient();  // 接收网络请求(获取从网络输入的密匙);**

**// 如果串口没有发送数据并且CMD不为空则将CMD的内容作为第一次输入的密匙;**

**if (CMD != "") {**

**PassWord\_Temp1 = CMD;**

**CMD = "";  // 清空CMD缓存;**

**break;**

**}**

**delay(500);**

**}**

**// 若等待超时后仍然没有输入密匙则跳过本次的循环进入下一次循环;**

**if (PassWord\_Temp1 == "") continue;**

**CMDCP\_Response("Re-enter password:");**

**server.send(200, "text/html", CMDCP\_Online\_Response);  // 向客户端发送请再次输入密码的字符串(主动发送不是响应);**

**// 等待30s, 期间可以输入第二次密码;**

**for (unsigned char i = 0; i < 60; ++i) {**

**// SerialReceived();       // 获取串口或WIFI数据;**

**server.handleClient();  // 接收网络请求(获取从网络输入的密匙);**

**// 如果串口没有发送数据并且CMD不为空则将CMD的内容作为第二次输入的密匙;**

**if (CMD != "") {**

**PassWord\_Temp2 = CMD;**

**CMD = "";  // 清空CMD缓存;**

**break;**

**}**

**delay(500);**

**}**

**// 比较两次输入的密码(通过的条件是两次密码相同并且不为空);**

**if (PassWord\_Temp1 == PassWord\_Temp2 && PassWord\_Temp1 != "") {**

**// 将设置的密码进行哈希加密后存入Flash中;**

**File dataFile = LittleFS.open("/PassWord.txt", "w");  // 创建&覆盖并打开PassWord.txt文件;**

**dataFile.print(String(sha1(PassWord\_Temp1)));         // 向dataFile写入哈希加密后的密匙信息;**

**dataFile.close();                                     // 完成文件写入后关闭文件**

**CMDCP\_Response("Accepted");**

**server.send(200, "text/html", CMDCP\_Online\_Response);  // 向客户端发送密码设置成功的字符串(主动发送不是响应);**

**allowResponse = true;                                  // 允许服务器对客户端进行响应;**

**CMDCP\_State = false;                                   // 用户离开CMDCP;**

**return false;  // 密码设置成功结束PassLocker程序, 重新启动PassLocker程序输入新密码即可;**

**}**

**}**

**} else {**

**// 将Flash中的密匙文件缓存到PassWord变量中;**

**PassWord = FFileS.readFile("", "/PassWord.txt");**

**// for循环决定了超时时间, 如果什么都不做则90s后超时, 并且用户有三次机会输入正确的密码;**

**for (unsigned char j = 0; j < 3; ++j) {**

**CMDCP\_Response("Enter password:");**

**server.send(200, "text/html", CMDCP\_Online\_Response);  // 向客户端发送请输入密码的字符串(主动发送不是响应);**

**// 等待30s, 期间可以输入密码;**

**for (unsigned char i = 0; i < 60; ++i) {**

**// SerialReceived();       // 获取串口或WIFI数据;**

**server.handleClient();  // 接收网络请求(获取从网络输入的密匙);**

**// 如果串口没有发送数据并且CMD不为空则将CMD的内容作为第二次输入的密匙;**

**if (CMD != "") {**

**PassWord\_Temp1 = String(sha1(CMD));  // 将输入的密码进行哈希加密后缓存到变量PassWord\_Temp1;**

**CMD = "";                            // 清空CMD缓存;**

**break;**

**}**

**delay(500);**

**}**

**// 比较输入密码与PassWord缓存中的密码是否一致, 如果一致则输入的密码正确, 返回true;**

**if (PassWord == PassWord\_Temp1) {**

**CMDCP\_Response("Passed");**

**server.send(200, "text/html", CMDCP\_Online\_Response);  // 向客户端发送密码正确的字符串(主动发送不是响应);**

**allowResponse = true;                                  // 允许服务器对客户端进行响应;**

**return true;**

**}**

**}**

**}**

**CMDCP\_State = false;   // 用户离开CMDCP;**

**allowResponse = true;  // 允许服务器对客户端进行响应;**

**return false;          // 密码错误返回false;**

**}**

**void CMDCP\_Response(String Response) {**

**CMDCP\_Online\_Response = "";  // 清空网络响应缓存;**

**if (Response != "") {**

**CMDCP\_Online\_Response = Response;  // 设置网络响应缓存;**

**oled.print(Response);**

**Serial.println(Response);**

**}**

**}**

**// 保存执行的命令;**

**void saveCmdHistory(String CMD, String clientIP) {**

**LittleFS.begin();  // 启动闪存文件系统**

**File cmdHistory = LittleFS.open("/CMD\_History.txt", "a");                       // 打开CMD\_History.txt追加日志;**

**cmdHistory.print(clientIP + "-" + timeRef.timeRead(false) + "-" + CMD + "\n");  // IP地址+时间+命令+换行;**

**cmdHistory.close();**

**}**

**public:**

**bool allow = false;**

**void begin() {**

**oled.setTextBox(0, 0, 128, 48);  // 设置文本框;**

**// CMDControlPanel\_ticker.attach\_ms(1000, [this](void) -> void { allow = true; });**

**}**

**String CMDControlPanelOnlinePortal(String CMDCP\_Online\_Message) {**

**// 如果客户端请求了空指令则直接结束函数响应空字符串;**

**if (CMDCP\_Online\_Message == "") return "";**

**CMD = CMDCP\_Online\_Message;  // 更新在线控制台发送的网络命令到CMD缓存;**

**String clientLogingIP = server.client().remoteIP().toString();  // 获取用户的IP地址;**

**// 检查用户的IP地址;**

**bool LockerState = false;        // 将CMD设为锁定状态;**

**for (auto& i : clientLogedIP) {  // 将请求用户的IP与已登录并且正在使用还未登出CMD用户的IP地址进行比对;**

**if (i == clientLogingIP) {**

**LockerState = true;  // 如果发现该IP还在登录状态(未登出)则为此用户开放CMD;**

**break;**

**}**

**}**

**// 当处于用户未登录状态(锁定状态)并且接收到进入CMDCP的命令;**

**if (LockerState == false && (CMD == "CMD" || CMD == "cmd" || CMD == "login")) {**

**CMDCP\_State = true;  // 用户打开CMDCP;**

**LockerState = PassLocker();  // 新用户登录或已退出登录的用户重新登录, 需要输入密码更新PassLocker状态;**

**// 如果用户输入了正确的密码;**

**if (LockerState == true) {**

**// 只有clientLogedIP为空时(即首个用户登入)才进入开发模式, 并设置文本框全屏;**

**if (clientLogedIP.empty() == true) {**

**Developer\_Mode = true;           // 进入开发者模式(此模式下不会显示状态栏, 不会显示桌面时钟);**

**oled.setTextBox(0, 0, 128, 64);  // 使控制台文本框全屏显示;**

**}**

**clientLogedIP.push\_back(clientLogingIP);  // 将该用户的IP地址标记为已登录并且正在使用还未登出CMD用户的IP;**

**/\*记录用户的登录时间和IP地址\*/**

**LittleFS.begin();                                                                  // 启动闪存文件系统**

**File cmdLoggedInfo = LittleFS.open("/CMD\_Logged\_Info.txt", "a");                   // 打开CMD\_Logged\_Info.txt追加日志;**

**cmdLoggedInfo.print(clientLogingIP + "-" + timeRef.timeRead(false) + "-login\n");  // IP地址+时间+登入记录;**

**cmdLoggedInfo.close();**

**}**

**}**

**if (LockerState == true) {**

**saveCmdHistory(CMD, clientLogingIP);  // 保存执行的命令;**

**commandIndexer();                     // 已登录用户可使用CMDCP;**

**}**

**allow = false;**

**return CMDCP\_Online\_Response;**

**}**

**void CMDControlPanelSerialPortal() {**

**SerialReceived();  // 获取和更新通过串口发送的命令到CMD缓存;**

**// 当处于锁定状态并且没有在接收串口数据并且接收到进入CMDCP的命令;**

**if (LockerState == false && Serial.available() == false && (CMD == "CMD" || CMD == "cmd" || CMD == "login")) {**

**LockerState = PassLocker();  // 更新PassLocker状态;**

**// 如果用户进入了CMDCP;**

**if (LockerState == true) {**

**Developer\_Mode = true;           // 进入开发者模式(此模式下不会显示状态栏, 不会显示桌面时钟);**

**oled.setTextBox(0, 0, 128, 64);  // 使控制台文本框全屏显示;**

**}**

**}**

**if (LockerState == true) commandIndexer();  // 处于解锁状态时可使用CMDCP;**

**allow = false;**

**}**

**void SerialReceived() {**

**CMD = "";                                                            // 清空缓存CMD指令;**

**while (Serial.available()) CMD += static\_cast<char>(Serial.read());  // 获取指令;**

**}**

**void commandIndexer() {**

**oled.print("> " + CMD);**

**Serial.println("> " + CMD);**

**// 以空格分割字符串;**

**vector<String> CMD\_Index = oled.strsplit(CMD, " ");**

**// {CMDCP指令帮助}help**

**if (CMD\_Index[0] == "help") {**

**CMDCP\_Response(CMDCP\_HELP);**

**}**

**//{显示当前工作目录(print work directory)}pwd**

**if (CMD\_Index[0] == "pwd") {**

**CMDCP\_Response(FFileS.getWorkDirectory());**

**}**

**// {显示工作目录下的文件列表(List files)}ls**

**if (CMD\_Index[0] == "ls") {**

**String listDirectory = FFileS.listDirectoryContents();**

**CMDCP\_Response(listDirectory);**

**}**

**/\***

**{切换当前工作目录(Change directory)}cd [dirName]**

**[cd ~][cd /] : 切换到Flash根目录;**

**[cd -] : 返回上一个打开的目录;**

**\*/**

**if (CMD\_Index[0] == "cd") {**

**if (CMD\_Index[1] == "~") {**

**FFileS.changeDirectory("/");**

**} else if (CMD\_Index[1] == "-") {**

**FFileS.backDirectory();**

**} else {**

**// 检查字符串 CMD\_Index[1] 的最后一个字符是否是斜杠"/", 如果不是，就在字符串末尾添加一个斜杠。**

**if (CMD\_Index[1].charAt(CMD\_Index[1].length() - 1) != '/') CMD\_Index[1] += '/';**

**FFileS.changeDirectory(CMD\_Index[1]);**

**}**

**CMDCP\_Response("");  // 空响应(该指令无响应内容);**

**}**

**// {打开工作目录下的文件(concatenate)}cat [fileName]**

**if (CMD\_Index[0] == "cat") {**

**String File\_Info = FFileS.readFile(CMD\_Index[1]);**

**CMDCP\_Response(File\_Info);**

**}**

**// {在工作目录下创建空文件}touch [fileName]**

**if (CMD\_Index[0] == "touch") {**

**FFileS.createFile(CMD\_Index[1]);**

**CMDCP\_Response("");  // 空响应(该指令无响应内容);**

**}**

**// {在工作目录下新建文件夹(Make Directory)}mkdir [dirName]**

**if (CMD\_Index[0] == "mkdir") {**

**FFileS.makeDirector(CMD\_Index[1]);**

**CMDCP\_Response("");  // 空响应(该指令无响应内容);**

**}**

**/\***

**echo [string] ：内容打印到控制台;**

**echo [string] > [fileName] ：将内容直接覆盖到工作目录的文件中;**

**echo [string] >> [fileName] ：将内容追加到工作目录的文件中;**

**\*/**

**if (CMD\_Index[0] == "echo") {**

**if (CMD\_Index[2] == ">") {**

**FFileS.fileCover(CMD\_Index[1], CMD\_Index[3]);**

**CMDCP\_Response("");  // 空响应(该指令无响应内容);**

**} else if (CMD\_Index[2] == ">>") {**

**FFileS.fileAppend(CMD\_Index[1], CMD\_Index[3]);**

**CMDCP\_Response("");  // 空响应(该指令无响应内容);**

**} else {**

**CMDCP\_Response(CMD\_Index[1]);**

**}**

**}**

**/\***

**{删除一个文件或者目录(Remove)}**

**rm [fileName] : 删除工作目录下的文件;**

**rm -r [dirName] : 删除工作目录下的文件夹;**

**\*/**

**if (CMD\_Index[0] == "rm") {**

**if (CMD\_Index[1] == "-r") {**

**FFileS.removeDirector(CMD\_Index[2]);**

**} else {**

**FFileS.removeFile(CMD\_Index[1]);**

**}**

**CMDCP\_Response("");  // 空响应(该指令无响应内容);**

**}**

**/\***

**{复制一个文件或目录(Copy file)}cp [-options] [sourcePath] [targetPath];**

**cp [源文件路径] [目标文件路径] : 复制一个文件到另一个文件;**

**cp -r [源目录路径] [目标目录路径] : 复制一个目录到另一个目录;**

**\*/**

**if (CMD\_Index[0] == "cp") {**

**if (CMD\_Index[1] == "-r") {**

**FFileS.copyDir(CMD\_Index[2], CMD\_Index[3]);**

**} else {**

**FFileS.copyFile(CMD\_Index[1], CMD\_Index[2]);**

**}**

**CMDCP\_Response("");  // 空响应(该指令无响应内容);**

**}**

**/\***

**{文件或目录改名或将文件或目录移入其它位置(Move)}mv [-options] [sourcePath] [targetPath];**

**mv [源文件路径] [目标文件路径] : 移动一个文件到另一个文件;**

**mv -r [源目录路径] [目标目录路径] : 移动一个目录到另一个目录;**

**\*/**

**if (CMD\_Index[0] == "mv") {**

**if (CMD\_Index[1] == "-r") {**

**FFileS.copyDir(CMD\_Index[2], CMD\_Index[3], true);**

**} else {**

**FFileS.copyFile(CMD\_Index[1], CMD\_Index[2], true);**

**}**

**CMDCP\_Response("");  // 空响应(该指令无响应内容);**

**}**

**/\***

**{查找指定目录下的文件(包含子目录的文件)}find [dirPath] [fileName] : 在dirPath目录下按文件名查找文件;**

**[fileName] = \*.\* : 查找所有文件;**

**[fileName] = \*.txt : 查找所有扩展名为txt的文件;**

**[fileName] = a.txt : 查找a.txt文件;**

**\*/**

**if (CMD\_Index[0] == "find") {**

**// 检查字符串 CMD\_Index[1] 的最后一个字符是否是斜杠"/", 如果不是，就在字符串末尾添加一个斜杠。**

**if (CMD\_Index[1].charAt(CMD\_Index[1].length() - 1) != '/') CMD\_Index[1] += '/';**

**String foundFile = FFileS.findFiles(CMD\_Index[1], CMD\_Index[2]);**

**CMDCP\_Response(foundFile);**

**}**

**//{显示操作系统版本信息}osinfo**

**if (CMD\_Index[0] == "osinfo") {**

**CMDCP\_Response(GSG3\_Os\_Info);**

**oled.OLED\_DrawBMP(0, 0, 128, 48, GasSensorGen3OS\_Info);  // 显示 GasSensorGen3OS\_Info;**

**}**

**//{立刻重新启动MCU}reboot**

**if (CMD\_Index[0] == "reboot") {**

**digitalWrite(RST, LOW);  // MCU复位;**

**CMDCP\_Response("");      // 空响应(该指令无响应内容);**

**}**

**//{打印主要GPIO引脚的状态(Print GPIO status)}pios**

**if (CMD\_Index[0] == "pios") {**

**CMDCP\_Response(SysSleep.GPIO\_Read());**

**}**

**//{打印主要系统状态(Print System status)}pss**

**if (CMD\_Index[0] == "pss") {**

**CMDCP\_Response(SysSleep.getSysModeAndStatus());**

**}**

**/\***

**{点亮板载的RGBLED}led [color] [state]**

**led r 1/true/enable : 点亮红色的LED**

**led g 1/true/enable : 点亮绿色的LED**

**led b 1/true/enable : 点亮蓝色的LED**

**led r 0/false/disable : 熄灭红色的LED**

**led g 0/false/disable : 熄灭绿色的LED**

**led b 0/false/disable : 熄灭蓝色的LED**

**note: 点亮红灯和绿灯会触发下载模式进而导致复位, 因此我们要先禁用下载模式;**

**\*/**

**if (CMD\_Index[0] == "led") {**

**if (CMD\_Index[2] == "1" || CMD\_Index[2] == "true" || CMD\_Index[2] == "enable") {**

**if (CMD\_Index[1] == "r") {**

**allowDownloadMode = false;  // 禁用下载模式;**

**alert.LED\_R\_Enable(0, true);**

**} else if (CMD\_Index[1] == "g") {**

**allowDownloadMode = false;  // 禁用下载模式;**

**alert.LED\_G\_Enable(0, true);**

**} else if (CMD\_Index[1] == "b") {**

**alert.LED\_B\_Enable(0, true);**

**}**

**} else if (CMD\_Index[2] == "0" || CMD\_Index[2] == "false" || CMD\_Index[2] == "disable") {**

**if (CMD\_Index[1] == "r") {**

**alert.LED\_R\_Enable(0, false);**

**allowDownloadMode = true;  // 启用下载模式;**

**} else if (CMD\_Index[1] == "g") {**

**alert.LED\_G\_Enable(0, false);**

**allowDownloadMode = true;  // 启用下载模式;**

**} else if (CMD\_Index[1] == "b") {**

**alert.LED\_B\_Enable(0, false);**

**}**

**}**

**CMDCP\_Response("");  // 空响应(该指令无响应内容);**

**}**

**/\***

**{打开蜂鸣器}buzz [state]**

**state = 1/true/enable : 打开蜂鸣器**

**state = 0/false/disable : 关闭蜂鸣器**

**\*/**

**if (CMD\_Index[0] == "buzz") {**

**if (CMD\_Index[1] == "1" || CMD\_Index[1] == "true" || CMD\_Index[1] == "enable") {**

**allowDownloadMode = false;     // 禁用下载模式;**

**alert.BUZZER\_Enable(0, true);  // 打开蜂鸣器**

**} else if (CMD\_Index[1] == "0" || CMD\_Index[1] == "false" || CMD\_Index[1] == "disable") {**

**alert.BUZZER\_Enable(0, false);  // 关闭蜂鸣器**

**allowDownloadMode = true;       // 启用下载模式;**

**}**

**CMDCP\_Response("");  // 空响应(该指令无响应内容);**

**}**

**//{关闭所有声光警报(Alert disable)}alertdis**

**if (CMD\_Index[0] == "alertdis") {**

**alert.ALERT\_Disable();  // 关闭所有声光警报;**

**CMDCP\_Response("");     // 空响应(该指令无响应内容);**

**}**

**/\***

**{浅休眠模式}freeze [enable]**

**[休眠模式-freeze], 冻结I/O设备, 关闭外设, ESP-12F进入Modem-sleep模式, 程序上只运行CMDControlPanel网络服务, 其他服务冻结;**

**freeze 1/true/enable : 进入浅度休眠模式;**

**freeze 0/false/disable : 离开浅度休眠模式;**

**\*/**

**if (CMD\_Index[0] == "freeze") {**

**if (CMD\_Index[1] == "1" || CMD\_Index[1] == "true" || CMD\_Index[1] == "enable") {**

**SysSleep.Sys\_freezeMode(true);**

**} else if (CMD\_Index[1] == "0" || CMD\_Index[1] == "false" || CMD\_Index[1] == "disable") {**

**SysSleep.Sys\_freezeMode(false);**

**}**

**CMDCP\_Response("");  // 空响应(该指令无响应内容);**

**}**

**/\***

**{深度睡眠模式}disk [time\_us]**

**[深度休眠模式-disk] 运行状态(GPIO\_Status, 系统模式和状态, 文本框信息)数据存到Flash(醒来时恢复状态), 然后ESP12F进入深度睡眠;**

**time\_us(微秒) = 0 : 无限期进入深度睡眠, 只有手动按RST复位才能恢复;**

**\*/**

**if (CMD\_Index[0] == "disk") {**

**if (CMD\_Index[1].toInt() != 0) SysSleep.Sys\_diskMode(CMD\_Index[1].toInt());**

**CMDCP\_Response("");  // 空响应(该指令无响应内容);**

**}**

**/\***

**{显示历史执行过的命令}history [-options]**

**history : 显示历史执行过的命令;**

**history -s : (history -sleep)显示深度睡眠前执行过的命令;**

**history -c : (history -clear)清空所有的命令历史记录;**

**\*/**

**if (CMD\_Index[0] == "history") {**

**if (CMD\_Index[1] == "-s") {**

**CMDCP\_Response(FFileS.readFile("", "/SleepFile/PrintBox.txt"));**

**} else if (CMD\_Index[1] == "-c") {**

**FFileS.removeFile("", "/CMD\_History.txt");**

**CMDCP\_Response("");**

**} else {**

**CMDCP\_Response(FFileS.readFile("", "/CMD\_History.txt"));**

**}**

**}**

**// {查看当前登入主机的用户终端IP}who**

**if (CMD\_Index[0] == "who") {**

**String who = "";**

**for (auto& i : clientLogedIP) who += (i + "\n");**

**CMDCP\_Response(who);**

**}**

**/\***

**{查看所有系统登录记录}last [-options];**

**last : 查看所有系统登录记录;**

**last -c : 清空登录记录;**

**\*/**

**if (CMD\_Index[0] == "last") {**

**if (CMD\_Index[1] == "-c") {**

**FFileS.removeFile("", "/CMD\_Logged\_Info.txt");**

**CMDCP\_Response("");**

**} else {**

**CMDCP\_Response(FFileS.readFile("", "/CMD\_Logged\_Info.txt"));**

**}**

**}**

**/\***

**{显示或设置系统时间}date [-options] [timeStr];**

**date : 显示系统时间;**

**date -n : 同步网络时间;**

**date -s [timeStr] : 根据字符串设置(set)系统时间, timeStr = 20230203121601 (Year Month Day Hour Minute Second);**

**\*/**

**if (CMD\_Index[0] == "date") {**

**if (CMD\_Index[1] == "-n") {**

**timeRef.getNetWorkTime();  // 同步网络时间;**

**CMDCP\_Response("");        // 空响应(该指令无响应内容);**

**} else if (CMD\_Index[1] == "-s") {**

**timeRef.sysTime.setSystemTime(CMD\_Index[2]);  // 设置系统时间;**

**CMDCP\_Response("");                           // 空响应(该指令无响应内容);**

**} else {**

**CMDCP\_Response(timeRef.timeRead(false));  // 显示系统时间;**

**}**

**}**

**/\***

**{显示当前实时天气或修改城市}weather [-options] [cityID];**

**weather : 显示当前实时天气;**

**weather -n : 同步网络实时天气;**

**weather -s [cityID] : 设置城市, "cityID"中请填写心知天气的城市ID;**

**\*/**

**if (CMD\_Index[0] == "weather") {**

**if (CMD\_Index[1] == "-n") {**

**CMDCP\_Response(weather.updateWeather());  // 同步网络实时天气(如果更新失败则返回错误信息);**

**} else if (CMD\_Index[1] == "-s") {**

**CMDCP\_Response(weather.setCityID(CMD\_Index[2]));  // 设置城市ID(设置完成后会同步一次天气, 如果天气更新失败则返回错误信息);**

**} else {**

**// 显示当前实时天气;**

**CMDCP\_Response("CityID: " + weatherNow.getCityID() + "\nCityName: " + weatherNow.getCityName() + "\nCountry: " + weatherNow.getCountry() +**

**"\nPath: " + weatherNow.getPath() + "\nTimezone: " + weatherNow.getTimezone() +**

**"\nTimezoneOffset: " + weatherNow.getTimezoneOffset() + "\nWeatherNow: " + weatherNow.getWeatherText() +**

**"\nTemperature: " + String(weatherNow.getTemperature()) + "C" + "\nLastUpdate: " + weatherNow.getLastUpdate());**

**}**

**}**

**/\***

**{OLED显示屏文本框滚动}pgup/pgdn [line];**

**pgup : 向上滚动一行;**

**pgup -s [line] : 向上滚动, "line"为滚动的行数;**

**pgdn : 向下滚动一行;**

**pgdn -s [line] : 向下滚动, "line"为滚动的行数;**

**\*/**

**if (CMD\_Index[0] == "pgup") {**

**if (CMD\_Index[1] == "-s") {**

**for (unsigned char i = 0; i < CMD\_Index[2].toInt(); ++i) oled.moveScrollBar(false);**

**} else {**

**oled.moveScrollBar(false);**

**}**

**CMDCP\_Response("");  // 空响应(该指令无响应内容);**

**} else if (CMD\_Index[0] == "pgdn") {**

**if (CMD\_Index[1] == "-s") {**

**for (unsigned char i = 0; i < CMD\_Index[2].toInt(); ++i) oled.moveScrollBar(true);**

**} else {**

**oled.moveScrollBar(true);**

**}**

**CMDCP\_Response("");  // 空响应(该指令无响应内容);**

**}**

**//{清空控制台同时释放内存}clear**

**if (CMD\_Index[0] == "clear") {**

**oled.clearTextBox();**

**CMDCP\_Response("");  // 空响应(该指令无响应内容);**

**}**

**//{显示Flash信息(disk free)}df**

**if (CMD\_Index[0] == "df") {**

**String Flash\_Info = FFileS.getFlash\_info();**

**CMDCP\_Response(Flash\_Info);**

**}**

**// {显示剩余内存}free**

**if (CMD\_Index[0] == "free") {**

**String FreeHeap = "FreeRAM: " + String(ESP.getFreeHeap()) + " Byte";**

**CMDCP\_Response(FreeHeap);**

**}**

**// {配置WIFI连接, 设置WIFI名称和密码}wifi [SSID] [PASSWORD]**

**if (CMD\_Index[0] == "wifi") {**

**FFileS.fileCover(CMD\_Index[1] + "<SSID/PASSWD>" + CMD\_Index[2], "", "/WIFI\_Config.ini");**

**CMDCP\_Response("");**

**}**

**//{关闭电源(并不会真的关闭电源, 只是无限期的深度休眠)}poweroff**

**if (CMD\_Index[0] == "poweroff") {**

**SysSleep.Sys\_diskMode(0);**

**CMDCP\_Response("");  // 空响应(该指令无响应内容);**

**}**

**//  {从终端上传文件到服务器}upload [-options];**

**// upload -s : 查看上一次文件上传的结果(终端在上传完成后会自动请求一次该指令, 以返回结果);**

**if (CMD\_Index[0] == "upload") {**

**if (CMD\_Index[1] == "-s") {**

**CMDCP\_Response(WebServer.UploadRespond);**

**} else {**

**WebServer.uploadFile();  // 准备接收文件;**

**CMDCP\_Response("");      // 空响应(该指令无响应内容);**

**}**

**}**

**/\***

**{登出和锁定CMDCP}logout [-options] [clientIP];**

**logout : 自己登出, 不会影响其他终端;**

**logout -k [clientIP] : 将指定IP地址的终端登出(kill);**

**logout -k other : 登出(kill)除自己外的其他终端;**

**logout -k all : 登出所有终端;**

**\*/**

**auto clientLogout = [this](String clientIP) -> void {**

**// 将该用户的IP从已登录并且正在使用还未登出CMD用户的IP地址中删除(即注销该用户的IP, 标记为未登录状态);**

**clientLogedIP.erase(remove(clientLogedIP.begin(), clientLogedIP.end(), clientIP), clientLogedIP.end());**

**// 在上面的代码中，remove 函数在 vector 中删除所有与 clientIP 字符串相等的字符串，erase 函数删除 vector 中剩余的空元素。**

**/\*记录用户的登出时间和IP地址\*/**

**File cmdLoggedInfo = LittleFS.open("/CMD\_Logged\_Info.txt", "a");              // 打开CMD\_Logged\_Info.txt追加日志;**

**cmdLoggedInfo.print(clientIP + "-" + timeRef.timeRead(false) + "-logout\n");  // IP地址+时间+登出记录;**

**cmdLoggedInfo.close();                                                        // 关闭文件;**

**};**

**if (CMD\_Index[0] == "logout") {**

**LittleFS.begin();  // 启动闪存文件系统**

**if (CMD\_Index[1] == "-k") {**

**if (CMD\_Index[2] == "other") {**

**String clientLogoutIP = server.client().remoteIP().toString();  // 获取当前用户的IP地址;**

**vector<String> allClientIP = clientLogedIP;  // 这里必须先拷贝一份clientLogedIP, 因为在执行clientLogout()时会删除clientLogedIP的元素.**

**for (auto& i : allClientIP)**

**if (i != clientLogoutIP) clientLogout(i);  // 登出除自身外的所有终端;**

**} else if (CMD\_Index[2] == "all") {**

**vector<String> allClientIP = clientLogedIP;  // 这里必须先拷贝一份clientLogedIP, 因为在执行clientLogout()时会删除clientLogedIP的元素.**

**for (auto& i : allClientIP) clientLogout(i);  // 登出所有终端;**

**} else {**

**clientLogout(CMD\_Index[2]);  // 登出指定IP地址的终端;**

**}**

**} else {**

**String clientLogoutIP = server.client().remoteIP().toString();  // 获取当前用户的IP地址(自身登出);**

**clientLogout(clientLogoutIP);                                   // 登出指定IP地址的终端;**

**}**

**// 只有clientLogedIP为空时(所有用户都登出)才回到桌面;**

**if (clientLogedIP.empty() == true) {**

**CMDCP\_State = false;             // 用户关闭CMDCP;**

**LockerState = false;             // 锁定CMDCP;**

**Developer\_Mode = false;          // 退出开发者模式(显示状态栏和桌面时钟);**

**oled.setTextBox(0, 0, 128, 48);  // 设置文本框使其不遮挡状态栏;**

**oled.OLED\_Clear();               // 清空OLED屏幕;**

**Desktop.Main\_Desktop();          // 刷新桌面时钟;**

**}**

**CMDCP\_Response("");  // 空响应(该指令无响应内容);**

**}**

**CMD = "";  // 清空命令;**

**}**

**} CMDCP;**

**class DrawingBoard {**

**private:**

**POINT last\_pos = {0, 0};**

**public:**

**void mouse(unsigned char x, unsigned char y) {**

**oled.clearrectangle(last\_pos.x, last\_pos.y, last\_pos.x + 2, last\_pos.y + 2);**

**oled.fillrectangle(x, y, x + 2, y + 2);**

**last\_pos.x = x;**

**last\_pos.y = y;**

**}**

**} DB;**

**void webServerBegin() {**

**server.begin();  // 启动服务器;**

**/\*打开网络路由\*/**

**// 打开"/"Route, 发送CMDCP\_Online页面;**

**// server.on("/", []() { server.send(200, "text/html", FFileS.readFile("", "/WebServer/CMDCP\_Online.html")); });// - 01;**

**server.on("/", []() { server.send(200, "text/html", CMDCP\_Online); });  // - 02;**

**// CMDCP\_Online**

**//  FFileS.readFile("", "/WebServer/CMDCP\_Online.html")**

**// 打开"/CMD"Route, 接收CMD指令和发送指令执行结果;**

**server.on("/CMD", []() {**

**// 从浏览器发送的信息中获取指令（字符串格式）**

**String CMDCP\_Online\_Message = server.arg("message");**

**// 将字符串送入CMDCP解析命令;**

**String CMDCP\_Send\_Message = CMDCP.CMDControlPanelOnlinePortal(CMDCP\_Online\_Message);**

**// 向终端响应CMDCP的返回值;**

**if (allowResponse == true) server.send(200, "text/html", CMDCP\_Send\_Message);**

**});**

**// 回复状态码 200 给客户端**

**auto respondOK = [](void) -> void { server.send(200); };**

**// "/upload"路由, 用于接收网页文件;**

**server.on("/upload", HTTP\_POST, respondOK, []() {**

**// 获取路由开启许可, 如果获得许可则打开上传通道立即接收文件, 否则返回404(许可的作用是保证服务器安全).**

**if (WebServer.getRouteUploadStatus() == true) {**

**WebServer.handleFileUpload();**

**} else {**

**server.send(404, "text/plain", "404 Not Found");**

**}**

**});**

**/\***

**server.on("/DB", []() {**

**String mousePosition = server.arg("plain");**

**unsigned char commaIndex = mousePosition.indexOf(',');**

**unsigned char x = mousePosition.substring(0, commaIndex).toInt();**

**unsigned char y = mousePosition.substring(commaIndex + 1).toInt();**

**DB.mouse(x, y);**

**server.send(200, "text/plain", "Received");**

**});**

**\*/**

**}**

**void setup(void) {**

**Serial.begin(115200);**

**// 初始化OLED**

**oled.OLED\_Init();**

**oled.OLED\_ColorTurn(0);                       // 0正常显示 1反色显示**

**oled.OLED\_DisplayTurn(0);                     // 0正常显示 1翻转180度显示**

**oled.OLED\_DrawBMP(0, 0, 128, 64, RMSHE\_IMG);  // 显示 RMSHE\_Infinity LOGO;**

**LittleFS.begin();  // 启动闪存文件系统**

**/\*----初始化GPIO----\*/**

**alert.AlertInit();  // 声光警报器初始化;**

**pinMode(RST, OUTPUT);     // 复位引脚初始化;**

**digitalWrite(RST, HIGH);  // 复位引脚设为高电平;**

**pinMode(LOWPOWER, INPUT\_PULLUP);  // 电池低电量输入引脚初始化(上拉输入);**

**pinMode(SENOUT, INPUT\_PULLUP);    // 传感器输入引脚初始化(上拉输入);**

**pinMode(CHRG, INPUT\_PULLUP);      // 电池状态输入引脚初始化(上拉输入);**

**/\*----GPIO初始化完成----\*/**

**WebServer.WiFi\_Connect();  // 连接WIFI;**

**WebServer.MQTT\_Begin();    // 配置MQTT并连接阿里云MQTT服务器;**

**http.setTimeout(TimeOut);           // 设置连接超时时间;**

**http.begin(client, GetSysTimeUrl);  // 初始化获取网络时间;**

**timeRef.getNetWorkTime();           // 获取网络时间;**

**weather.beginWeather();   // 初始化天气信息获取;**

**weather.updateWeather();  // 更新天气信息;**

**// 不触发警报的条件下每隔5min检查一次WIFI是否连接若没有连接则连接(多线程);**

**WIFI\_Test.attach(300, [](void) -> void {**

**// 检测WIFI是否连接, 若没有连接则修改标志, 主线程识别到WIFI\_State = false后会执行连接WIFI的函数;**

**if (digitalRead(SENOUT) == HIGH && WiFi.status() != WL\_CONNECTED) WIFI\_State = false;**

**});**

**timeRef.begin();  // 用于更新时间(多线程);**

**Desktop.begin();  // 用于刷新桌面(多线程);**

**CMDCP.begin();    // 启动CMDControlPanel服务;**

**SysSleep.resumeFromDeepSleep();  // 从深度睡眠恢复系统(恢复深度睡眠前的状态, 如果是从深度睡眠中醒来的话);**

**webServerBegin();  // 启动网络服务器;**

**oled.OLED\_Clear();       // 清空屏幕;**

**Desktop.Main\_Desktop();  // 渲染主桌面;**

**}**

**void loop(void) {**

**if (digitalRead(SENOUT) == LOW && freezeMode == false) {**

**alert.flashWriteAlertLog("S" + timeRef.timeRead(false));  // 写警报日志(开始报警);**

**unsigned int WarningCycle = 0;**

**while (digitalRead(SENOUT) == LOW) {**

**alert.LED\_B\_Enable(100 \* cos(0.1 \* WarningCycle + 3.14) + 100);  // 蓝色警报灯亮**

**oled.OLED\_DrawBMP(0, 0, 128, 64, Fire\_Warning);                  // 显示火灾警报**

**alert.LED\_R\_Enable(100 \* cos(0.1 \* WarningCycle) + 200);         // 红色警报灯亮**

**ShowFireWarning();                                               // 显示消防预警**

**// 同时点亮RGB三色灯90ms;**

**for (unsigned char i = 0; i < 30; ++i) {**

**alert.LED\_R\_Enable(1);**

**alert.LED\_G\_Enable(1);**

**alert.LED\_B\_Enable(1);**

**}**

**alert.BUZZER\_Enable(400 \* cos(0.1 \* WarningCycle) + 500);  // 蜂鸣器报警**

**++WarningCycle;**

**}**

**alert.flashWriteAlertLog("E" + timeRef.timeRead(false));  // 写警报日志(结束报警);**

**alert.ALERT\_Disable();                                    // 关闭所有声光警报;**

**oled.OLED\_Clear();                                        // 清除界面**

**Desktop.Main\_Desktop();                                   // 渲染主桌面;**

**}**

**Show\_Charging\_info();   // 判断电池是否进入充电并显示电池开始充电的信息;**

**ProgramDownloadMode();  // 程序下载模式(下载程序必须下拉GPIO0并且复位);**

**// 当允许获取网络时间时会执行此程序(同步本地系统时间);**

**if (timeRef.allow == true) {**

**timeRef.getNetWorkTime();  // 获取网络时间;**

**timeRef.allow = false;     // 吊销许可证(网络时间获取许可由定时器授予);**

**}**

**if (WIFI\_State == false) WebServer.WiFi\_Connect();  // 如果WIFI连接标志为false则连接WIFI;**

**WebServer.MQTT\_Client();**

**server.handleClient();**

**}**

**// oledfont.h---------------------------------------------------------------------------------------------------------**

**#pragma once**

**#ifndef \_\_OLEDFONT\_H**

**#define \_\_OLEDFONT\_H**

**#include <avr/pgmspace.h>**

**/\*"RMSHE\_Infinity.png"\*/**

**const unsigned char RMSHE\_IMG[] PROGMEM = {**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0xC0, 0xC0, 0xE0, 0xF0, 0xF0, 0xF8, 0xB8, 0x08, 0x1C,**

**0x1C, 0x3C, 0x7C, 0xFC, 0xF8, 0xF0, 0xE0, 0xC0, 0x80, 0xE0, 0xF0, 0xF8, 0xF8, 0xFC, 0x7C, 0x3C, 0x1C, 0x18, 0x18, 0xF8, 0xF0, 0xF0, 0xE0, 0xC0, 0xC0,**

**0x80, 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, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF8, 0xFC, 0xFE, 0xFF, 0x81, 0x01, 0x03, 0x3F, 0xFF, 0xFF, 0xFF, 0x0F,**

**0x0F, 0x1F, 0x3E, 0x3E, 0x7C, 0x78, 0x70, 0x61, 0x77, 0x7F, 0x3F, 0x3F, 0x7F, 0x7F, 0x63, 0x61, 0x70, 0x78, 0x7C, 0x3E, 0x1F, 0x1F, 0x0F, 0xFF, 0xFF,**

**0xFF, 0xFF, 0x03, 0x03, 0x03, 0xE3, 0xFE, 0xFC, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0xFF, 0xFF, 0xFF, 0x41,**

**0x41, 0xE1, 0xBF, 0xBE, 0x0C, 0x00, 0x01, 0x01, 0xFF, 0x07, 0x3F, 0xFC, 0xF0, 0x80, 0x80, 0x78, 0x04, 0xFF, 0xFF, 0xFF, 0x01, 0x00, 0x00, 0x1C, 0x3E,**

**0x71, 0x71, 0xE1, 0xE1, 0xC7, 0x80, 0x00, 0x00, 0x01, 0xFF, 0xFF, 0xFF, 0x41, 0x40, 0x40, 0x41, 0xFF, 0xFF, 0xFF, 0x01, 0x00, 0x01, 0xFF, 0xFF, 0xFF,**

**0x41, 0x41, 0xF1, 0x03, 0x07, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xE0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF9, 0x7F, 0x7F, 0x7F, 0xFF, 0xFC, 0xF0, 0x70, 0x37,**

**0x1F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x0F, 0x1F, 0x3F, 0x70, 0xF0, 0xF8, 0xFE, 0xFF, 0xFF, 0xFF, 0xF3, 0xF0, 0xF0, 0xF0, 0xE0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x08, 0x0F,**

**0x0F, 0x0F, 0x08, 0x00, 0x01, 0x0F, 0x0F, 0x0C, 0x08, 0x08, 0x08, 0x0F, 0x08, 0x08, 0x01, 0x0F, 0x0F, 0x01, 0x00, 0x08, 0x0F, 0x0F, 0x0F, 0x08, 0x00,**

**0x00, 0x07, 0x08, 0x08, 0x08, 0x08, 0x0C, 0x07, 0x03, 0x00, 0x00, 0x08, 0x0F, 0x0F, 0x0F, 0x08, 0x00, 0x00, 0x08, 0x0F, 0x0F, 0x0F, 0x08, 0x00, 0x08,**

**0x0F, 0x0F, 0x0F, 0x08, 0x08, 0x08, 0x0C, 0x0E, 0x00, 0x00, 0x00, 0x7C, 0xFF, 0xFB, 0xE1, 0xE1, 0xF0, 0xF0, 0xF8, 0x78, 0x78, 0x3C, 0x3C, 0x1D, 0x1F,**

**0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x3D, 0x3C, 0x78, 0x78, 0xF8, 0xF0, 0xF1, 0xE1, 0xE1, 0xE3, 0xF7, 0xFF, 0xFC, 0x00, 0x00,**

**0x00, 0x04, 0xFC, 0xFC, 0xFC, 0x04, 0x00, 0x20, 0xE0, 0xE0, 0xE0, 0x40, 0x20, 0x20, 0xE0, 0xE0, 0xC0, 0x00, 0x20, 0xF0, 0xF8, 0xFC, 0x24, 0x22, 0x22,**

**0xE6, 0xE4, 0xE0, 0x00, 0x20, 0xE0, 0xE0, 0xE0, 0x40, 0x20, 0x20, 0xE0, 0xE0, 0xC0, 0x00, 0x40, 0xC4, 0xEE, 0xE4, 0x00, 0x20, 0xF0, 0xFC, 0xFC, 0x20,**

**0x20, 0x00, 0x20, 0x60, 0xE0, 0xE0, 0x20, 0x00, 0x20, 0xE0, 0x20, 0x00, 0x00, 0x00, 0x3F, 0xFF, 0xEF, 0xC7, 0x87, 0x87, 0x8F, 0x0F, 0x1F, 0x1E, 0x1E,**

**0x3C, 0xBC, 0xF8, 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, 0xE0, 0xF8, 0x38, 0x3C, 0x3C, 0x1E, 0x1E, 0x1F, 0x0F, 0x0F, 0x87, 0x87, 0xDF, 0xFF,**

**0x3E, 0x00, 0x00, 0x00, 0x20, 0x3F, 0x3F, 0x3F, 0x20, 0x00, 0x20, 0x3F, 0x3F, 0x3F, 0x20, 0x00, 0x20, 0x3F, 0x3F, 0x3F, 0x20, 0x20, 0x3F, 0x3F, 0x3F,**

**0x20, 0x00, 0x20, 0x3F, 0x3F, 0x3F, 0x20, 0x20, 0x3F, 0x3F, 0x3F, 0x20, 0x00, 0x20, 0x3F, 0x3F, 0x3F, 0x20, 0x20, 0x3F, 0x3F, 0x3F, 0x20, 0x00, 0x1F,**

**0x3F, 0x3F, 0x20, 0x20, 0x00, 0x00, 0x00, 0x03, 0x1F, 0xFE, 0x18, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x07, 0x0F, 0x0F, 0x0F,**

**0xCF, 0xFF, 0xFF, 0xFF, 0x7F, 0x1F, 0x0F, 0x0E, 0xFC, 0xF8, 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, 0xF0, 0xF8, 0xEC, 0x0E, 0x0F, 0x3F, 0xFF, 0xFE, 0xFE, 0xFE, 0x9F, 0x0F, 0x0F, 0x0F, 0x0F,**

**0x07, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x1F, 0x3F, 0x7F, 0xC7, 0xC0, 0xC0, 0xC0, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0xF8, 0xF8, 0x7C, 0x3E, 0x1E, 0x0E, 0x86, 0xC6, 0xFE, 0xFE,**

**0xFC, 0xFC, 0xFE, 0xEE, 0x86, 0x0E, 0x1E, 0x3E, 0x7C, 0x7C, 0xF8, 0xF0, 0xF0, 0xFF, 0xFF, 0xFF, 0xFC, 0xC0, 0x80, 0x81, 0xFF, 0x7F, 0x3F, 0x1F, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x07, 0x0F, 0x0F, 0x1F, 0x18, 0x18, 0x38, 0x3C, 0x3E, 0x3F, 0x1F,**

**0x0F, 0x0F, 0x07, 0x01, 0x03, 0x07, 0x0F, 0x1F, 0x3F, 0x3E, 0x3C, 0x38, 0x38, 0x10, 0x1D, 0x1F, 0x0F, 0x0F, 0x07, 0x03, 0x03, 0x01, 0x01, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**};**

**// GasSensorGen3OS\_Info.png;**

**const unsigned char GasSensorGen3OS\_Info[] PROGMEM = {**

**0x00, 0xFE, 0x02, 0x02, 0xC2, 0xF2, 0x32, 0x32, 0x32, 0x32, 0xC2, 0xC2, 0x02, 0x02, 0x82, 0x82, 0x82, 0x82, 0x02, 0x02, 0x02, 0x02, 0x82, 0x82, 0x82,**

**0x82, 0x02, 0x02, 0x02, 0xC2, 0xF2, 0xB2, 0x32, 0x32, 0x32, 0x02, 0x02, 0x02, 0x82, 0x82, 0x82, 0x82, 0x02, 0x02, 0x82, 0x82, 0x82, 0x82, 0x82, 0x02,**

**0x02, 0x02, 0x02, 0x82, 0x82, 0x82, 0x82, 0x02, 0x02, 0x02, 0x02, 0x82, 0x82, 0x82, 0x82, 0x02, 0x02, 0x02, 0x02, 0x82, 0x82, 0x82, 0x82, 0x02, 0xC2,**

**0xF2, 0x32, 0x32, 0x32, 0x32, 0xC2, 0xC2, 0x02, 0xC2, 0xF2, 0x32, 0x32, 0xF2, 0xC2, 0x02, 0x02, 0x02, 0x02, 0x02, 0x02, 0xF2, 0xF2, 0xF2, 0xF2, 0xF2,**

**0xF2, 0xF2, 0xF2, 0x02, 0x02, 0x02, 0x02, 0x02, 0x02, 0x02, 0x02, 0x02, 0x02, 0xF2, 0xF2, 0xF2, 0xF2, 0xF2, 0xF2, 0xF2, 0xF2, 0x02, 0x02, 0x02, 0x02,**

**0x02, 0xFE, 0x00, 0x00, 0xFF, 0x00, 0x00, 0x3F, 0x7F, 0x60, 0x66, 0x66, 0x66, 0x7E, 0x7E, 0x00, 0x30, 0x7D, 0x6D, 0x6D, 0x7F, 0x7E, 0x00, 0x00, 0x66,**

**0x6F, 0x6D, 0x6D, 0x7D, 0x30, 0x00, 0x00, 0x30, 0x71, 0x61, 0x66, 0x7E, 0x3C, 0x00, 0x00, 0x3E, 0x7F, 0x6D, 0x6D, 0x6F, 0x6E, 0x00, 0x7F, 0x7F, 0x01,**

**0x01, 0x7F, 0x7E, 0x00, 0x00, 0x66, 0x6F, 0x6D, 0x6D, 0x7D, 0x30, 0x00, 0x00, 0x3E, 0x7F, 0x61, 0x61, 0x7F, 0x3E, 0x00, 0x00, 0x7E, 0x7F, 0x01, 0x01,**

**0x01, 0x00, 0x3F, 0x7F, 0x60, 0x66, 0x66, 0x66, 0x7E, 0x7E, 0x00, 0x30, 0x76, 0x66, 0x66, 0x7F, 0x3D, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x7F, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0xF8, 0x80,**

**0x80, 0x80, 0x00, 0x00, 0xFF, 0x00, 0x00, 0xFF, 0x00, 0x00, 0xFE, 0x28, 0xEE, 0x7C, 0xAA, 0x4A, 0xBA, 0x00, 0x44, 0x34, 0x4C, 0xFE, 0x4C, 0x34, 0x44,**

**0x00, 0x48, 0x00, 0x00, 0x38, 0x44, 0x54, 0x74, 0x00, 0x48, 0x54, 0x54, 0x20, 0x00, 0x40, 0x00, 0x64, 0x54, 0x48, 0x00, 0x78, 0x44, 0x3C, 0x00, 0x64,**

**0x54, 0x48, 0x00, 0x44, 0x54, 0x2C, 0x00, 0x78, 0x44, 0x3C, 0x00, 0x08, 0x7C, 0x00, 0x44, 0x54, 0x2C, 0x00, 0x78, 0x44, 0x3C, 0x00, 0x00, 0x40, 0x00,**

**0x7C, 0x08, 0x30, 0x08, 0x7C, 0x00, 0x68, 0x58, 0x70, 0x00, 0x70, 0x08, 0x00, 0x7E, 0x10, 0x68, 0x00, 0x78, 0x44, 0x3C, 0x00, 0x00, 0x00, 0x07, 0x07,**

**0x07, 0x3F, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3F, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C,**

**0x3C, 0x3F, 0x07, 0x07, 0x07, 0x00, 0x00, 0xFF, 0x00, 0x00, 0xFF, 0x00, 0x00, 0x44, 0x7C, 0xD4, 0x7C, 0x54, 0x7C, 0x04, 0x00, 0xA8, 0x78, 0xEC, 0xB8,**

**0xFC, 0x14, 0xF4, 0x00, 0x90, 0x00, 0x00, 0x60, 0x90, 0xF0, 0x00, 0x00, 0xF4, 0x00, 0x00, 0x78, 0x90, 0x00, 0x00, 0xFC, 0x10, 0xE0, 0x00, 0x00, 0x70,**

**0x80, 0xF0, 0x00, 0x00, 0xFC, 0x90, 0x60, 0x00, 0x00, 0x80, 0x00, 0x00, 0x60, 0x90, 0xA0, 0x00, 0x00, 0x60, 0x90, 0x60, 0x00, 0x00, 0xF0, 0x10, 0xE0,**

**0x10, 0xE0, 0x00, 0x00, 0xC0, 0x30, 0x0C, 0x00, 0xF8, 0x28, 0x68, 0x90, 0x00, 0x00, 0xF8, 0x10, 0x60, 0x10, 0xF8, 0x00, 0x00, 0x90, 0xA8, 0xA8, 0x40,**

**0x00, 0xF8, 0x20, 0x20, 0xF8, 0x00, 0x00, 0xF8, 0xA8, 0xA8, 0x00, 0x00, 0x20, 0x20, 0x00, 0x00, 0xF8, 0x10, 0x60, 0x10, 0xF8, 0x00, 0x00, 0x90, 0xA8,**

**0xA8, 0x40, 0x00, 0x00, 0xF8, 0x20, 0x20, 0xF8, 0x00, 0x00, 0xFF, 0x00, 0x00, 0xFF, 0x00, 0x00, 0xC9, 0x39, 0xE8, 0xD9, 0xF9, 0xD9, 0xE9, 0x00, 0x20,**

**0xF1, 0x49, 0xB0, 0xA1, 0xF8, 0xA1, 0x00, 0x20, 0x00, 0x00, 0xE2, 0x12, 0x51, 0xD0, 0x00, 0x00, 0x20, 0x50, 0x50, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xE0, 0x10, 0x50, 0xD0, 0x00, 0x00, 0xC0, 0xA0, 0x60, 0x00, 0x00, 0xE0, 0x20, 0xC0, 0x00, 0x00, 0x10, 0x50, 0xB0, 0x00, 0x00, 0x00, 0x00, 0x90, 0x50,**

**0x20, 0x00, 0x00, 0xE0, 0x10, 0xF0, 0x00, 0x00, 0x90, 0x50, 0x20, 0x00, 0x00, 0x10, 0x50, 0xB0, 0x00, 0x00, 0xE0, 0x10, 0xF0, 0x00, 0x00, 0x20, 0xF0,**

**0x00, 0x00, 0x20, 0xF0, 0x00, 0x00, 0xE0, 0x10, 0xF0, 0x00, 0x00, 0x00, 0x00, 0xF0, 0x20, 0xC0, 0x20, 0xF0, 0x00, 0x00, 0xA0, 0x60, 0xC0, 0x00, 0x00,**

**0xC0, 0x20, 0x00, 0x00, 0xF8, 0x40, 0xA0, 0x00, 0x00, 0x20, 0xF0, 0x00, 0x00, 0xFF, 0x00, 0x00, 0x1F, 0x10, 0x10, 0x11, 0x11, 0x11, 0x12, 0x11, 0x12,**

**0x12, 0x10, 0x10, 0x13, 0x10, 0x10, 0x10, 0x13, 0x10, 0x10, 0x11, 0x10, 0x10, 0x10, 0x11, 0x11, 0x11, 0x10, 0x10, 0x11, 0x11, 0x11, 0x10, 0x10, 0x10,**

**0x11, 0x10, 0x10, 0x10, 0x11, 0x11, 0x11, 0x10, 0x10, 0x10, 0x11, 0x11, 0x10, 0x10, 0x11, 0x10, 0x11, 0x10, 0x10, 0x11, 0x11, 0x10, 0x10, 0x10, 0x11,**

**0x10, 0x11, 0x11, 0x11, 0x10, 0x10, 0x11, 0x11, 0x10, 0x10, 0x10, 0x11, 0x11, 0x11, 0x10, 0x10, 0x11, 0x11, 0x10, 0x10, 0x10, 0x11, 0x11, 0x10, 0x10,**

**0x10, 0x10, 0x11, 0x10, 0x10, 0x10, 0x11, 0x10, 0x10, 0x11, 0x11, 0x10, 0x10, 0x10, 0x11, 0x10, 0x11, 0x10, 0x10, 0x10, 0x11, 0x10, 0x10, 0x11, 0x11,**

**0x11, 0x10, 0x10, 0x11, 0x10, 0x10, 0x10, 0x11, 0x10, 0x11, 0x10, 0x10, 0x10, 0x11, 0x10, 0x10, 0x1F, 0x00,**

**};**

**/\*"火灾警报\_128x64.png"\*/**

**const unsigned char Fire\_Warning[] PROGMEM = {**

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

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

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

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

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

**0xC0, 0xE0, 0xF0, 0x04, 0x04, 0x04, 0x04, 0x05, 0x05, 0x05, 0x05, 0x05, 0x05, 0x05, 0x05, 0x05, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0x84, 0xC4, 0xE5,**

**0xE5, 0xE5, 0xE5, 0xC5, 0x85, 0x05, 0x05, 0x05, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0x05, 0x05, 0x05, 0x05, 0x05, 0x05, 0x05, 0x85, 0x85,**

**0x84, 0x84, 0xE4, 0xE4, 0xE4, 0xE4, 0xA4, 0xA4, 0x85, 0x85, 0xE5, 0xE5, 0xE5, 0xE5, 0xA5, 0xE5, 0xE5, 0xC4, 0xC4, 0x84, 0x04, 0x84, 0xF4, 0xF4, 0xE4,**

**0xE5, 0xE5, 0x65, 0x25, 0x05, 0x05, 0x85, 0xC5, 0x85, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0x05, 0x05, 0x05, 0xF5, 0xF5, 0xF5, 0xF5, 0xE5,**

**0x25, 0x24, 0x04, 0x04, 0x04, 0x04, 0xE4, 0xE4, 0xC4, 0xC5, 0x85, 0x85, 0x85, 0x85, 0x85, 0x85, 0x85, 0x85, 0x84, 0x84, 0x84, 0xC4, 0xE4, 0xF4, 0xC4,**

**0xC4, 0x85, 0x05, 0x05, 0x05, 0x05, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xF8, 0x7E, 0x1F,**

**0x07, 0x03, 0x01, 0x01, 0x01, 0x01, 0x03, 0x07, 0x1F, 0x3E, 0xFC, 0xF0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x80, 0xC0, 0x70, 0xFE, 0xFF, 0xFF, 0xCF, 0x4F, 0x4A, 0xE8, 0xF8, 0xE8, 0xCB, 0x49, 0x09, 0xFD, 0xFE, 0xFC, 0xFC, 0x08, 0x00, 0x60, 0x38, 0x1F,**

**0x1F, 0x0F, 0x3F, 0xF3, 0xE2, 0xF2, 0xFE, 0xFE, 0x7F, 0x3F, 0x1F, 0x03, 0x03, 0x03, 0x02, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0xFF, 0xFF,**

**0xFF, 0xFF, 0xFF, 0xC0, 0xF0, 0xF8, 0xF0, 0xE0, 0xC0, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xE0, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x7F, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF8, 0x7C, 0x1F, 0x0F, 0x03,**

**0xF0, 0xF8, 0xC0, 0xC0, 0xEC, 0x9C, 0xD8, 0x78, 0x3E, 0x7C, 0xF0, 0xC0, 0x60, 0xF0, 0x03, 0x07, 0x1F, 0x3E, 0xFC, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1F, 0x0F, 0x0F, 0x0F, 0x02, 0x02, 0x07, 0x07, 0x07, 0x07, 0x06, 0x1F, 0x1F, 0x1F, 0x7F, 0xF3, 0xF0, 0xE8,**

**0xE8, 0x64, 0x04, 0x06, 0x03, 0x03, 0x03, 0x01, 0x01, 0x03, 0x87, 0xC7, 0xEF, 0xEF, 0xDF, 0x9E, 0x06, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80,**

**0x80, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xC0, 0x60, 0x60, 0x20, 0x20, 0x10, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x40, 0xC0, 0x40, 0x40, 0x40, 0x41, 0x41, 0x4F,**

**0x4F, 0x4F, 0x4F, 0xE7, 0xF7, 0xFB, 0xF0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0xFC, 0x3F, 0x0F, 0x07, 0x01, 0x00,**

**0x00, 0x00, 0x00, 0x38, 0xFF, 0x1F, 0x07, 0xE1, 0xE1, 0xF3, 0xF0, 0xF8, 0xC0, 0xE1, 0x07, 0x06, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0x1F,**

**0x3E, 0xFC, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11,**

**0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x99, 0xDF, 0xFF, 0xDD, 0x99, 0x11, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x07,**

**0x7F, 0x3F, 0x1F, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x0F, 0x7C, 0xF0,**

**0xE0, 0xC0, 0x80, 0x00, 0xE0, 0xFE, 0xFF, 0xFF, 0xFF, 0x3F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0xFE, 0xFF, 0xFF, 0xC3, 0x80, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x10, 0x10, 0x10, 0x11, 0x12, 0x14, 0x15, 0x17, 0x17, 0x17, 0x15, 0x10, 0x12, 0x11, 0x11, 0x10, 0x10, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x80, 0xC3, 0xFF, 0xFF, 0xFE, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFE, 0xFD, 0xFD, 0xF9, 0xF9, 0x11, 0x11, 0x11,**

**0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0xF9, 0xFF, 0xFF, 0xFD, 0xF9, 0x10, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x20, 0x20, 0x60, 0xE0, 0xE0, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x81, 0xE7, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFB, 0xF0, 0xF0, 0xE0, 0xC0, 0xC0, 0x40, 0x00, 0xA0, 0xA0, 0xA0, 0xA0, 0x21, 0x23, 0x23,**

**0x27, 0x27, 0x27, 0x27, 0x27, 0xA7, 0xA7, 0xA7, 0xA7, 0xA7, 0xA7, 0xA7, 0xA7, 0xA7, 0x27, 0x27, 0x27, 0x27, 0x27, 0x27, 0x27, 0x27, 0xA7, 0xA7, 0xA7,**

**0xA7, 0xA7, 0xA7, 0xA7, 0xA7, 0xA7, 0x27, 0x23, 0x23, 0x21, 0x21, 0x20, 0x20, 0x20, 0xA0, 0xA0, 0xA0, 0xA0, 0xA0, 0xA0, 0xAF, 0xAF, 0xAF, 0x2F, 0x27,**

**0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0xA2, 0xA2, 0xA2, 0xA2, 0xA2, 0xA2, 0xA2, 0xA2, 0xA2, 0x2F, 0x2F, 0x2F, 0x2F, 0x27, 0x20, 0x20, 0x20, 0xA0, 0xA0,**

**0xA0, 0xA0, 0xA0, 0xA0, 0xA0, 0xA0, 0xA0, 0x20, 0x21, 0x2F, 0x2F, 0x2F, 0x27, 0x23, 0x21, 0xA0, 0xA0, 0xA0, 0xA0, 0xA0, 0xA0, 0xAF, 0xAF, 0xAF, 0x2F,**

**0x27, 0x28, 0x2C, 0x26, 0x27, 0x23, 0x23, 0xA1, 0xA0, 0xA0, 0xA1, 0xA1, 0xA3, 0xA7, 0xAF, 0xA3, 0x21, 0x20, 0x20, 0x20, 0x20, 0x07, 0x03, 0x01, 0x00,**

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

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

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

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

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

**};**

**// 5\*8 GSOS\_ASCII字符集点阵字体;**

**const unsigned char GSOS\_ASCII\_0805[][5] PROGMEM = {**

**{**

**0x00,**

**0x00,**

**0x00,**

**0x00,**

**0x00,**

**}, //" ",0;**

**{**

**0x00,**

**0x00,**

**0x5E,**

**0x00,**

**0x00,**

**}, //"!",1;**

**{**

**0x00,**

**0x18,**

**0x00,**

**0x18,**

**0x00,**

**}, //""",2;**

**{**

**0x00,**

**0x7E,**

**0x24,**

**0x7E,**

**0x00,**

**}, //"#",3;**

**{**

**0x00,**

**0x2C,**

**0x7E,**

**0x34,**

**0x00,**

**}, //"$",4;**

**{**

**0x00,**

**0x24,**

**0x10,**

**0x48,**

**0x00,**

**}, //"%",0;**

**{**

**0x76,**

**0x4A,**

**0x36,**

**0x50,**

**0x00,**

**}, //"&",0;**

**{**

**0x00,**

**0x00,**

**0x06,**

**0x00,**

**0x00,**

**}, //"'",2;**

**{**

**0x00,**

**0x7C,**

**0x82,**

**0x00,**

**0x00,**

**}, //"(",8;**

**{**

**0x00,**

**0x82,**

**0x7C,**

**0x00,**

**0x00,**

**}, //")",9;**

**{**

**0x00,**

**0x28,**

**0x1C,**

**0x28,**

**0x00,**

**}, //"\*",10;**

**{**

**0x00,**

**0x10,**

**0x7C,**

**0x10,**

**0x00,**

**}, //"+",11;**

**{**

**0x00,**

**0x10,**

**0x08,**

**0x00,**

**0x00,**

**}, //",",12;**

**{**

**0x00,**

**0x10,**

**0x10,**

**0x10,**

**0x00,**

**}, //"-",13;**

**{**

**0x00,**

**0x00,**

**0x20,**

**0x00,**

**0x00,**

**}, //".",0;**

**{**

**0x00,**

**0x60,**

**0x18,**

**0x06,**

**0x00,**

**}, //"/",15;**

**{**

**0x00,**

**0x7E,**

**0x42,**

**0x7E,**

**0x00,**

**}, //"0",0;**

**{**

**0x00,**

**0x44,**

**0x7E,**

**0x40,**

**0x00,**

**}, //"1",1;**

**{**

**0x00,**

**0x72,**

**0x52,**

**0x5E,**

**0x00,**

**}, //"2",2;**

**{**

**0x00,**

**0x52,**

**0x52,**

**0x7E,**

**0x00,**

**}, //"3",3;**

**{**

**0x00,**

**0x1E,**

**0x10,**

**0x7C,**

**0x00,**

**}, //"4",4;**

**{**

**0x00,**

**0x5E,**

**0x52,**

**0x72,**

**0x00,**

**}, //"5",5;**

**{**

**0x00,**

**0x7E,**

**0x52,**

**0x72,**

**0x00,**

**}, //"6",6;**

**{**

**0x00,**

**0x06,**

**0x02,**

**0x7E,**

**0x00,**

**}, //"7",7;**

**{**

**0x00,**

**0x7E,**

**0x4A,**

**0x7E,**

**0x00,**

**}, //"8",8;**

**{**

**0x00,**

**0x4E,**

**0x52,**

**0x7E,**

**0x00,**

**}, //"9",9;**

**{**

**0x00,**

**0x00,**

**0x28,**

**0x00,**

**0x00,**

**}, //":",26;**

**{**

**0x00,**

**0x20,**

**0x14,**

**0x00,**

**0x00,**

**}, //";",27;**

**{**

**0x00,**

**0x10,**

**0x28,**

**0x44,**

**0x00,**

**}, //"<",28;**

**{**

**0x00,**

**0x28,**

**0x28,**

**0x28,**

**0x00,**

**}, //"=",29;**

**{**

**0x00,**

**0x44,**

**0x28,**

**0x10,**

**0x00,**

**}, //">",30;**

**{**

**0x00,**

**0x04,**

**0x52,**

**0x0C,**

**0x00,**

**}, //"?",31;**

**{**

**0x3C,**

**0x42,**

**0x5A,**

**0x1C,**

**0x00,**

**}, //"@",32;**

**{**

**0x00,**

**0x7C,**

**0x12,**

**0x7C,**

**0x00,**

**}, //"A",33;**

**{**

**0x00,**

**0x7E,**

**0x4A,**

**0x34,**

**0x00,**

**}, //"B",34;**

**{**

**0x00,**

**0x3C,**

**0x42,**

**0x42,**

**0x00,**

**}, //"C",35;**

**{**

**0x00,**

**0x7E,**

**0x42,**

**0x3C,**

**0x00,**

**}, //"D",36;**

**{**

**0x00,**

**0x7E,**

**0x4A,**

**0x42,**

**0x00,**

**}, //"E",37;**

**{**

**0x00,**

**0x7E,**

**0x0A,**

**0x02,**

**0x00,**

**}, //"F",38;**

**{**

**0x00,**

**0x3C,**

**0x42,**

**0x72,**

**0x00,**

**}, //"G",39;**

**{**

**0x00,**

**0x7E,**

**0x10,**

**0x7E,**

**0x00,**

**}, //"H",40;**

**{**

**0x00,**

**0x42,**

**0x7E,**

**0x42,**

**0x00,**

**}, //"I",41;**

**{**

**0x00,**

**0x20,**

**0x40,**

**0x3E,**

**0x00,**

**}, //"J",42;**

**{**

**0x00,**

**0x7E,**

**0x08,**

**0x76,**

**0x00,**

**}, //"K",43;**

**{**

**0x00,**

**0x7E,**

**0x40,**

**0x40,**

**0x00,**

**}, //"L",44;**

**{**

**0x00,**

**0x7E,**

**0x0C,**

**0x7E,**

**0x00,**

**}, //"M",45;**

**{**

**0x00,**

**0x7E,**

**0x02,**

**0x7C,**

**0x00,**

**}, //"N",46;**

**{**

**0x00,**

**0x3C,**

**0x42,**

**0x3C,**

**0x00,**

**}, //"O",47;**

**{**

**0x00,**

**0x7E,**

**0x12,**

**0x0C,**

**0x00,**

**}, //"P",48;**

**{**

**0x00,**

**0x3C,**

**0x42,**

**0xBC,**

**0x00,**

**}, //"Q",49;**

**{**

**0x00,**

**0x7E,**

**0x12,**

**0x6C,**

**0x00,**

**}, //"R",50;**

**{**

**0x00,**

**0x44,**

**0x4A,**

**0x32,**

**0x00,**

**}, //"S",51;**

**{**

**0x00,**

**0x02,**

**0x7E,**

**0x02,**

**0x00,**

**}, //"T",52;**

**{**

**0x00,**

**0x3E,**

**0x40,**

**0x3E,**

**0x00,**

**}, //"U",53;**

**{**

**0x00,**

**0x1E,**

**0x60,**

**0x1E,**

**0x00,**

**}, //"V",54;**

**{**

**0x00,**

**0x7E,**

**0x30,**

**0x7E,**

**0x00,**

**}, //"W",55;**

**{**

**0x00,**

**0x66,**

**0x18,**

**0x66,**

**0x00,**

**}, //"X",56;**

**{**

**0x00,**

**0x06,**

**0x78,**

**0x06,**

**0x00,**

**}, //"Y",57;**

**{**

**0x00,**

**0x62,**

**0x5A,**

**0x46,**

**0x00,**

**}, //"Z",58;**

**{**

**0x00,**

**0xFE,**

**0x82,**

**0x00,**

**0x00,**

**}, //"[",59;**

**{**

**0x00,**

**0x06,**

**0x18,**

**0x60,**

**0x00,**

**}, //"\",60;**

**{**

**0x00,**

**0x82,**

**0xFE,**

**0x00,**

**0x00,**

**}, //"]",61;**

**{**

**0x00,**

**0x10,**

**0x08,**

**0x10,**

**0x00,**

**}, //"^",62;**

**{**

**0x00,**

**0x20,**

**0x20,**

**0x20,**

**0x00,**

**}, //"\_",2;**

**{**

**0x00,**

**0x08,**

**0x10,**

**0x00,**

**0x00,**

**}, //"`",64;**

**{**

**0x00,**

**0x34,**

**0x2C,**

**0x38,**

**0x00,**

**}, //"a",1;**

**{**

**0x00,**

**0x7E,**

**0x48,**

**0x30,**

**0x00,**

**}, //"b",66;**

**{**

**0x00,**

**0x18,**

**0x24,**

**0x24,**

**0x00,**

**}, //"c",67;**

**{**

**0x00,**

**0x30,**

**0x48,**

**0x7E,**

**0x00,**

**}, //"d",68;**

**{**

**0x00,**

**0x18,**

**0x34,**

**0x2C,**

**0x00,**

**}, //"e",69;**

**{**

**0x00,**

**0x08,**

**0x7E,**

**0x0A,**

**0x00,**

**}, //"f",70;**

**{**

**0x00,**

**0x4C,**

**0x52,**

**0x3E,**

**0x00,**

**}, //"g",71;**

**{**

**0x00,**

**0x7E,**

**0x08,**

**0x70,**

**0x00,**

**}, //"h",72;**

**{**

**0x00,**

**0x00,**

**0x7A,**

**0x00,**

**0x00,**

**}, //"i",73;**

**{**

**0x00,**

**0x80,**

**0x7A,**

**0x00,**

**0x00,**

**}, //"j",74;**

**{**

**0x00,**

**0x7E,**

**0x10,**

**0x68,**

**0x00,**

**}, //"k",75;**

**{**

**0x00,**

**0x00,**

**0x7E,**

**0x00,**

**0x00,**

**}, //"l",76;**

**{**

**0x00,**

**0x3C,**

**0x08,**

**0x3C,**

**0x00,**

**}, //"m",77;**

**{**

**0x00,**

**0x3C,**

**0x04,**

**0x38,**

**0x00,**

**}, //"n",78;**

**{**

**0x00,**

**0x18,**

**0x24,**

**0x18,**

**0x00,**

**}, //"o",79;**

**{**

**0x00,**

**0x7C,**

**0x24,**

**0x18,**

**0x00,**

**}, //"p",80;**

**{**

**0x00,**

**0x18,**

**0x24,**

**0x7C,**

**0x00,**

**}, //"q",81;**

**{**

**0x00,**

**0x3C,**

**0x08,**

**0x04,**

**0x00,**

**}, //"r",82;**

**{**

**0x00,**

**0x28,**

**0x3C,**

**0x14,**

**0x00,**

**}, //"s",83;**

**{**

**0x00,**

**0x08,**

**0x7C,**

**0x48,**

**0x00,**

**}, //"t",84;**

**{**

**0x00,**

**0x3C,**

**0x20,**

**0x3C,**

**0x00,**

**}, //"u",85;**

**{**

**0x00,**

**0x0C,**

**0x30,**

**0x0C,**

**0x00,**

**}, //"v",86;**

**{**

**0x00,**

**0x3C,**

**0x10,**

**0x3C,**

**0x00,**

**}, //"w",87;**

**{**

**0x00,**

**0x24,**

**0x18,**

**0x24,**

**0x00,**

**}, //"x",88;**

**{**

**0x00,**

**0x4C,**

**0x50,**

**0x3C,**

**0x00,**

**}, //"y",89;**

**{**

**0x00,**

**0x24,**

**0x34,**

**0x2C,**

**0x00,**

**}, //"z",90;**

**{**

**0x00,**

**0x10,**

**0x6C,**

**0x82,**

**0x00,**

**}, //"{",91;**

**{**

**0x00,**

**0x00,**

**0xFE,**

**0x00,**

**0x00,**

**}, //"|",92;**

**{**

**0x00,**

**0x82,**

**0x6C,**

**0x10,**

**0x00,**

**}, //"}",93;**

**{**

**0x10,**

**0x08,**

**0x10,**

**0x08,**

**0x00,**

**}, //"~",94;**

**};**

**// 8\*16 ASCII字符集点阵(Consolas)**

**const unsigned char asc2\_1608[][16] PROGMEM = {**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*" ",32\*/**

**{0x00, 0x00, 0x00, 0xF8, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x11, 0x1B, 0x00, 0x00, 0x00}, /\*"!",33\*/**

**{0x00, 0x00, 0x78, 0x08, 0x00, 0x78, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*""",34\*/**

**{0x00, 0x40, 0xE0, 0x50, 0x40, 0xF0, 0x50, 0x40, 0x00, 0x04, 0x1F, 0x04, 0x04, 0x1F, 0x04, 0x00}, /\*"#",35\*/**

**{0x00, 0xE0, 0xB0, 0x10, 0xF8, 0x10, 0x10, 0x00, 0x00, 0x10, 0x10, 0x7F, 0x11, 0x13, 0x0E, 0x00}, /\*"$",36\*/**

**{0x30, 0x58, 0x48, 0x70, 0x80, 0x60, 0x30, 0x08, 0x10, 0x18, 0x04, 0x03, 0x0C, 0x12, 0x12, 0x0C}, /\*"%",37\*/**

**{0x00, 0x60, 0xF0, 0x90, 0x90, 0x70, 0x00, 0x00, 0x04, 0x1F, 0x11, 0x11, 0x16, 0x1C, 0x1F, 0x10}, /\*"&",38\*/**

**{0x00, 0x00, 0x00, 0x78, 0x78, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"'",39\*/**

**{0x00, 0x00, 0x80, 0xE0, 0x30, 0x08, 0x00, 0x00, 0x00, 0x00, 0x07, 0x3F, 0x60, 0xC0, 0x00, 0x00}, /\*"(",40\*/**

**{0x00, 0x00, 0x08, 0x10, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x80, 0x60, 0x38, 0x0F, 0x00, 0x00}, /\*")",41\*/**

**{0x00, 0x80, 0xA0, 0x60, 0xD8, 0x60, 0x90, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00}, /\*"\*",42\*/**

**{0x00, 0x00, 0x00, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x02, 0x02, 0x1F, 0x1F, 0x02, 0x02, 0x02}, /\*"+",43\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x40, 0x58, 0x38, 0x00, 0x00, 0x00}, /\*",",44\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x02, 0x02, 0x02, 0x02, 0x00, 0x00}, /\*"-",45\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x18, 0x18, 0x00, 0x00, 0x00}, /\*".",46\*/**

**{0x00, 0x00, 0x00, 0x00, 0xC0, 0x70, 0x18, 0x00, 0x00, 0x20, 0x18, 0x06, 0x01, 0x00, 0x00, 0x00}, /\*"/",47\*/**

**{0x00, 0xE0, 0x30, 0x10, 0x10, 0xB0, 0xE0, 0x80, 0x00, 0x0F, 0x1A, 0x13, 0x11, 0x18, 0x0F, 0x03}, /\*"0",48\*/**

**{0x00, 0x60, 0x20, 0x30, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x10, 0x10, 0x10, 0x1F, 0x10, 0x10, 0x00}, /\*"1",49\*/**

**{0x00, 0x20, 0x10, 0x10, 0x10, 0xF0, 0xE0, 0x00, 0x00, 0x10, 0x18, 0x14, 0x12, 0x11, 0x10, 0x00}, /\*"2",50\*/**

**{0x00, 0x10, 0x10, 0x10, 0x10, 0xF0, 0x60, 0x00, 0x00, 0x10, 0x11, 0x11, 0x11, 0x19, 0x0E, 0x00}, /\*"3",51\*/**

**{0x00, 0x00, 0x80, 0x60, 0x30, 0xF0, 0x00, 0x00, 0x04, 0x07, 0x05, 0x04, 0x04, 0x1F, 0x04, 0x04}, /\*"4",52\*/**

**{0x00, 0xF0, 0xF0, 0x10, 0x10, 0x10, 0x10, 0x00, 0x00, 0x11, 0x11, 0x11, 0x11, 0x19, 0x0F, 0x00}, /\*"5",53\*/**

**{0x00, 0xC0, 0x60, 0x30, 0x90, 0x10, 0x10, 0x00, 0x00, 0x0F, 0x19, 0x11, 0x10, 0x11, 0x0F, 0x00}, /\*"6",54\*/**

**{0x00, 0x10, 0x10, 0x10, 0x10, 0xD0, 0x70, 0x00, 0x00, 0x00, 0x10, 0x1C, 0x07, 0x01, 0x00, 0x00}, /\*"7",55\*/**

**{0x00, 0x60, 0xB0, 0x90, 0x10, 0x90, 0xE0, 0x00, 0x00, 0x0E, 0x12, 0x11, 0x11, 0x13, 0x0E, 0x00}, /\*"8",56\*/**

**{0x00, 0xE0, 0x30, 0x10, 0x10, 0x30, 0xE0, 0x00, 0x00, 0x11, 0x11, 0x12, 0x12, 0x09, 0x07, 0x00}, /\*"9",57\*/**

**{0x00, 0x00, 0x00, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x18, 0x18, 0x00, 0x00, 0x00}, /\*":",58\*/**

**{0x00, 0x00, 0x00, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x40, 0x58, 0x38, 0x00, 0x00, 0x00}, /\*";",59\*/**

**{0x00, 0x00, 0x00, 0x80, 0x80, 0x40, 0x00, 0x00, 0x00, 0x02, 0x03, 0x05, 0x08, 0x18, 0x10, 0x00}, /\*"<",60\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x05, 0x05, 0x05, 0x05, 0x05, 0x05, 0x00}, /\*"=",61\*/**

**{0x00, 0x00, 0x40, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10, 0x08, 0x04, 0x07, 0x02, 0x00}, /\*">",62\*/**

**{0x00, 0x00, 0x08, 0x18, 0x10, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x1B, 0x01, 0x00, 0x00, 0x00}, /\*"?",63\*/**

**{0x80, 0xE0, 0x10, 0x88, 0x88, 0x88, 0x10, 0xE0, 0x3F, 0x60, 0x8F, 0x91, 0x88, 0x9F, 0x08, 0x0F}, /\*"@",64\*/**

**{0x00, 0x00, 0xC0, 0x70, 0x30, 0xE0, 0x00, 0x00, 0x10, 0x1E, 0x07, 0x04, 0x04, 0x05, 0x0F, 0x18}, /\*"A",65\*/**

**{0x00, 0xF0, 0x10, 0x10, 0x10, 0xB0, 0xE0, 0x00, 0x00, 0x1F, 0x11, 0x11, 0x11, 0x11, 0x0E, 0x00}, /\*"B",66\*/**

**{0x00, 0xC0, 0x20, 0x10, 0x10, 0x10, 0x10, 0x00, 0x00, 0x0F, 0x18, 0x10, 0x10, 0x10, 0x10, 0x00}, /\*"C",67\*/**

**{0x00, 0xF0, 0x10, 0x10, 0x10, 0x30, 0xE0, 0x80, 0x00, 0x1F, 0x10, 0x10, 0x10, 0x08, 0x0F, 0x03}, /\*"D",68\*/**

**{0x00, 0xF0, 0xF0, 0x10, 0x10, 0x10, 0x10, 0x00, 0x00, 0x1F, 0x1F, 0x11, 0x11, 0x11, 0x11, 0x00}, /\*"E",69\*/**

**{0x00, 0xF0, 0xF0, 0x10, 0x10, 0x10, 0x10, 0x00, 0x00, 0x1F, 0x1F, 0x01, 0x01, 0x01, 0x01, 0x00}, /\*"F",70\*/**

**{0x00, 0xE0, 0x20, 0x10, 0x10, 0x10, 0x10, 0x00, 0x01, 0x0F, 0x18, 0x10, 0x11, 0x11, 0x1F, 0x00}, /\*"G",71\*/**

**{0x00, 0xF0, 0x00, 0x00, 0x00, 0x00, 0xF0, 0x00, 0x00, 0x1F, 0x01, 0x01, 0x01, 0x01, 0x1F, 0x00}, /\*"H",72\*/**

**{0x00, 0x10, 0x10, 0xF0, 0xF0, 0x10, 0x10, 0x00, 0x00, 0x10, 0x10, 0x1F, 0x1F, 0x10, 0x10, 0x00}, /\*"I",73\*/**

**{0x00, 0x10, 0x10, 0x10, 0x10, 0xF0, 0x00, 0x00, 0x00, 0x18, 0x10, 0x10, 0x10, 0x0F, 0x00, 0x00}, /\*"J",74\*/**

**{0x00, 0xF0, 0x00, 0x80, 0x40, 0x20, 0x10, 0x00, 0x00, 0x1F, 0x01, 0x03, 0x06, 0x0C, 0x10, 0x00}, /\*"K",75\*/**

**{0x00, 0x00, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1F, 0x10, 0x10, 0x10, 0x10, 0x00}, /\*"L",76\*/**

**{0x00, 0xF0, 0x70, 0x80, 0x80, 0xE0, 0xF0, 0xC0, 0x1C, 0x1F, 0x00, 0x03, 0x03, 0x00, 0x03, 0x1F}, /\*"M",77\*/**

**{0x00, 0xF0, 0x70, 0xC0, 0x00, 0x00, 0xF0, 0x00, 0x00, 0x1F, 0x00, 0x01, 0x07, 0x1C, 0x1F, 0x00}, /\*"N",78\*/**

**{0x80, 0xE0, 0x30, 0x10, 0x10, 0x10, 0xE0, 0xC0, 0x03, 0x0F, 0x18, 0x10, 0x10, 0x10, 0x0F, 0x07}, /\*"O",79\*/**

**{0x00, 0xF0, 0x10, 0x10, 0x10, 0x30, 0xE0, 0x00, 0x00, 0x1F, 0x02, 0x02, 0x02, 0x01, 0x01, 0x00}, /\*"P",80\*/**

**{0x80, 0xE0, 0x30, 0x10, 0x10, 0x10, 0xE0, 0xC0, 0x03, 0x0F, 0x18, 0x30, 0x70, 0x50, 0x4F, 0x47}, /\*"Q",81\*/**

**{0x00, 0xF0, 0xF0, 0x10, 0x10, 0xB0, 0xE0, 0x00, 0x00, 0x1F, 0x1F, 0x01, 0x03, 0x0F, 0x18, 0x00}, /\*"R",82\*/**

**{0x00, 0xE0, 0xB0, 0x10, 0x10, 0x10, 0x10, 0x00, 0x00, 0x10, 0x10, 0x11, 0x11, 0x13, 0x0E, 0x00}, /\*"S",83\*/**

**{0x00, 0x10, 0x10, 0xF0, 0xF0, 0x10, 0x10, 0x10, 0x00, 0x00, 0x00, 0x1F, 0x1F, 0x00, 0x00, 0x00}, /\*"T",84\*/**

**{0x00, 0xF0, 0x00, 0x00, 0x00, 0x00, 0xF0, 0x00, 0x00, 0x0F, 0x18, 0x10, 0x10, 0x10, 0x0F, 0x00}, /\*"U",85\*/**

**{0x10, 0xF0, 0x80, 0x00, 0x00, 0x00, 0xE0, 0x30, 0x00, 0x00, 0x07, 0x1C, 0x18, 0x0F, 0x01, 0x00}, /\*"V",86\*/**

**{0x30, 0xF0, 0x00, 0x00, 0x80, 0x00, 0x00, 0xF0, 0x00, 0x1F, 0x1C, 0x07, 0x03, 0x1C, 0x1F, 0x03}, /\*"W",87\*/**

**{0x00, 0x30, 0x60, 0xC0, 0x80, 0x60, 0x30, 0x10, 0x10, 0x18, 0x0C, 0x03, 0x03, 0x0E, 0x18, 0x10}, /\*"X",88\*/**

**{0x10, 0x70, 0xC0, 0x80, 0x00, 0xC0, 0x60, 0x10, 0x00, 0x00, 0x00, 0x1F, 0x1F, 0x01, 0x00, 0x00}, /\*"Y",89\*/**

**{0x00, 0x10, 0x10, 0x10, 0x90, 0xD0, 0x30, 0x00, 0x00, 0x18, 0x1C, 0x13, 0x11, 0x10, 0x10, 0x00}, /\*"Z",90\*/**

**{0x00, 0x00, 0x00, 0xF8, 0x08, 0x08, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0x80, 0x80, 0x00, 0x00}, /\*"[",91\*/**

**{0x00, 0x08, 0x30, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1C, 0x30, 0x00}, /\*"\",92\*/**

**{0x00, 0x00, 0x08, 0x08, 0xF8, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0xFF, 0xFF, 0x00, 0x00}, /\*"]",93\*/**

**{0x00, 0x80, 0xC0, 0x30, 0x30, 0x60, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"^",94\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80}, /\*"\_",95\*/**

**{0x00, 0x00, 0x08, 0x10, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"`",96\*/**

**{0x00, 0x00, 0x40, 0x40, 0x40, 0xC0, 0x80, 0x00, 0x00, 0x1C, 0x12, 0x12, 0x12, 0x0A, 0x1F, 0x00}, /\*"a",97\*/**

**{0x00, 0xF8, 0x80, 0x40, 0x40, 0x40, 0x80, 0x00, 0x00, 0x1F, 0x10, 0x10, 0x10, 0x10, 0x0F, 0x00}, /\*"b",98\*/**

**{0x00, 0x00, 0x80, 0x40, 0x40, 0x40, 0x40, 0x00, 0x00, 0x07, 0x1D, 0x10, 0x10, 0x10, 0x10, 0x00}, /\*"c",99\*/**

**{0x00, 0x80, 0xC0, 0x40, 0x40, 0x40, 0xF8, 0x00, 0x00, 0x0F, 0x18, 0x10, 0x10, 0x08, 0x1F, 0x00}, /\*"d",100\*/**

**{0x00, 0x80, 0xC0, 0x40, 0x40, 0x40, 0x80, 0x00, 0x00, 0x0F, 0x1A, 0x12, 0x12, 0x12, 0x13, 0x00}, /\*"e",101\*/**

**{0x00, 0x80, 0x80, 0xF0, 0x98, 0x88, 0x88, 0x08, 0x00, 0x00, 0x00, 0x1F, 0x00, 0x00, 0x00, 0x00}, /\*"f",102\*/**

**{0x00, 0x80, 0xC0, 0x40, 0x40, 0xC0, 0xC0, 0x40, 0x00, 0x6F, 0x92, 0x94, 0x94, 0x92, 0x73, 0x20}, /\*"g",103\*/**

**{0x00, 0xF8, 0x80, 0x40, 0x40, 0x40, 0x80, 0x00, 0x00, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x1F, 0x00}, /\*"h",104\*/**

**{0x00, 0x40, 0x40, 0x58, 0xD8, 0x00, 0x00, 0x00, 0x00, 0x10, 0x10, 0x10, 0x1F, 0x10, 0x10, 0x00}, /\*"i",105\*/**

**{0x00, 0x40, 0x40, 0x40, 0x50, 0xD8, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0xC0, 0x7F, 0x00, 0x00}, /\*"j",106\*/**

**{0x00, 0xF8, 0xF8, 0x00, 0x80, 0x80, 0x40, 0x00, 0x00, 0x1F, 0x1F, 0x03, 0x05, 0x08, 0x10, 0x00}, /\*"k",107\*/**

**{0x00, 0x08, 0x08, 0x08, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x10, 0x10, 0x10, 0x1F, 0x10, 0x10, 0x00}, /\*"l",108\*/**

**{0x00, 0xC0, 0x40, 0x40, 0x80, 0x40, 0xC0, 0x80, 0x00, 0x1F, 0x00, 0x00, 0x1F, 0x00, 0x1F, 0x1F}, /\*"m",109\*/**

**{0x00, 0xC0, 0x80, 0x40, 0x40, 0x40, 0x80, 0x00, 0x00, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x1F, 0x00}, /\*"n",110\*/**

**{0x00, 0x80, 0xC0, 0x40, 0x40, 0x40, 0x80, 0x00, 0x00, 0x0F, 0x18, 0x10, 0x10, 0x10, 0x0F, 0x02}, /\*"o",111\*/**

**{0x00, 0xC0, 0x80, 0x40, 0x40, 0x40, 0x80, 0x00, 0x00, 0xFF, 0x10, 0x10, 0x10, 0x10, 0x0F, 0x00}, /\*"p",112\*/**

**{0x00, 0x80, 0xC0, 0x40, 0x40, 0x40, 0xC0, 0x00, 0x00, 0x0F, 0x18, 0x10, 0x10, 0x08, 0xFF, 0x00}, /\*"q",113\*/**

**{0x00, 0xC0, 0xC0, 0x80, 0x40, 0x40, 0xC0, 0x00, 0x00, 0x1F, 0x1F, 0x00, 0x00, 0x00, 0x01, 0x01}, /\*"r",114\*/**

**{0x00, 0x00, 0xC0, 0x40, 0x40, 0x40, 0x40, 0x00, 0x00, 0x10, 0x13, 0x12, 0x12, 0x16, 0x0C, 0x00}, /\*"s",115\*/**

**{0x40, 0x40, 0x40, 0xF0, 0x40, 0x40, 0x40, 0x00, 0x00, 0x00, 0x00, 0x1F, 0x10, 0x10, 0x10, 0x00}, /\*"t",116\*/**

**{0x00, 0xC0, 0x00, 0x00, 0x00, 0x00, 0xC0, 0x00, 0x00, 0x0F, 0x18, 0x10, 0x10, 0x08, 0x1F, 0x00}, /\*"u",117\*/**

**{0x00, 0xC0, 0x00, 0x00, 0x00, 0x00, 0xC0, 0x40, 0x00, 0x01, 0x07, 0x1C, 0x18, 0x07, 0x01, 0x00}, /\*"v",118\*/**

**{0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0x00, 0x1F, 0x18, 0x07, 0x07, 0x1C, 0x1F, 0x01}, /\*"w",119\*/**

**{0x00, 0x40, 0xC0, 0x00, 0x00, 0x80, 0xC0, 0x00, 0x00, 0x10, 0x18, 0x07, 0x07, 0x0D, 0x18, 0x00}, /\*"x",120\*/**

**{0x00, 0xC0, 0x00, 0x00, 0x00, 0x00, 0xC0, 0x40, 0x80, 0x81, 0xC7, 0x7C, 0x38, 0x0F, 0x01, 0x00}, /\*"y",121\*/**

**{0x00, 0x40, 0x40, 0x40, 0x40, 0xC0, 0xC0, 0x00, 0x00, 0x10, 0x18, 0x16, 0x13, 0x11, 0x10, 0x00}, /\*"z",122\*/**

**{0x00, 0x00, 0x00, 0xF0, 0x18, 0x08, 0x08, 0x00, 0x00, 0x02, 0x03, 0x7F, 0xC0, 0x80, 0x80, 0x00}, /\*"{",123\*/**

**{0x00, 0x00, 0x00, 0x00, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0x00, 0x00, 0x00}, /\*"|",124\*/**

**{0x00, 0x00, 0x08, 0x18, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0x7F, 0x03, 0x02, 0x00}, /\*"}",125\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x01, 0x01, 0x02, 0x04, 0x06, 0x03}, /\*"~",126\*/**

**};**

**// 16\*32 ASCII字符集点阵(Consolas)**

**const unsigned char asc2\_3216[][64] PROGMEM = {**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*" ",0\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xCF, 0xCF, 0x80, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"!",1\*/**

**{0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1F, 0x1F,**

**0x07, 0x00, 0x00, 0x07, 0x1F, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*""",2\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0x30, 0x30, 0xF0, 0xFF,**

**0x3F, 0x30, 0x30, 0x30, 0xFE, 0xFF, 0x33, 0x30, 0x30, 0x00, 0x00, 0x0C, 0x0C, 0xEC, 0xFF, 0x3F, 0x0C, 0x0C, 0x0C, 0xFC, 0xFF, 0x0F,**

**0x0C, 0x0C, 0x0C, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"#",35\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7C, 0xFE, 0xE6, 0xC3,**

**0x83, 0xFB, 0xFF, 0x0F, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0x80, 0x80, 0x81, 0xFD, 0xFF, 0x87, 0x83, 0xC7, 0xEF,**

**0xFE, 0x7C, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x0F, 0x0F, 0x03, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"$",36\*/**

**{0x00, 0x80, 0x80, 0xC0, 0xC0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0x40, 0x00, 0x1F, 0x3F, 0x30, 0x30, 0x30,**

**0x1F, 0x8F, 0xE0, 0x70, 0x3C, 0x0E, 0x07, 0x01, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0x38, 0x1C, 0x07, 0x03, 0xF9, 0xFC, 0x86, 0x06,**

**0x86, 0xFC, 0xFC, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"%",37\*/**

**{0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x3F, 0xFF, 0xE1,**

**0xC1, 0xC1, 0x61, 0x7F, 0x3F, 0x1E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7C, 0xFF, 0xE7, 0x83, 0x81, 0x83, 0x87, 0x8F, 0xDC, 0xF8, 0xF0,**

**0xFF, 0xDF, 0x80, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01}, /\*"&",38\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x01, 0x1F, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"'",39\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0x40, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF0,**

**0xFC, 0x3E, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1F, 0xFF, 0xFF, 0xC0, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0x0F, 0x1C, 0x38, 0x30, 0x00, 0x00, 0x00, 0x00}, /\*"(",40\*/**

**{0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01,**

**0x03, 0x0F, 0x3E, 0xFC, 0xF0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xFF, 0xFF, 0x0F,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0x38, 0x1C, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*")",41\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x62, 0x66, 0x24,**

**0x3C, 0xFF, 0xFF, 0x3C, 0x24, 0x66, 0x67, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"\*",42\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xF8, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x06, 0x06, 0x06, 0x06, 0x06, 0x06, 0xFF, 0xFF, 0x06, 0x06, 0x06,**

**0x06, 0x06, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"+",43\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 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, 0xE0, 0xE0, 0xC0, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0x10, 0x18, 0x18, 0x1F, 0x0F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*",",44\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x06, 0x06, 0x06, 0x06, 0x06, 0x06, 0x06, 0x06,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"-",45\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 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, 0xE0, 0xE0, 0xE0, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*".",46\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0x40, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x80, 0xE0, 0xFC, 0x1F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF8, 0x3E, 0x0F, 0x03, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x0F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"/",47\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF8, 0xFE, 0x0E, 0x03,**

**0x03, 0x83, 0x83, 0xC3, 0xE7, 0x6F, 0xFE, 0xF8, 0xC0, 0x00, 0x00, 0x07, 0x3F, 0xFF, 0xEE, 0xC6, 0x87, 0x83, 0x81, 0x81, 0xC0, 0xE0,**

**0x7F, 0x3F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"0",48\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x0C, 0x0E, 0x06,**

**0x07, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0xFF, 0xFF, 0xFF, 0x80, 0x80,**

**0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"1",49\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x02, 0x06, 0x07, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0xCF, 0xFE, 0x7C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF0, 0xB8, 0x9C, 0x8E, 0x87, 0x83, 0x81,**

**0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"2",50\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x83,**

**0x83, 0x83, 0x83, 0xC3, 0xFF, 0x7E, 0x3C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x81, 0x81, 0x81, 0x81, 0x83, 0xC3, 0xFF,**

**0x7E, 0x3C, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"3",51\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xF0,**

**0x38, 0x1E, 0x07, 0x03, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1C, 0x1F, 0x1F, 0x19, 0x18, 0x18, 0x18, 0x18, 0x18, 0xFF, 0xFF,**

**0x18, 0x18, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00}, /\*"4",52\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x83,**

**0x83, 0x83, 0x83, 0x83, 0x83, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xC3, 0xFF,**

**0x7F, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"5",53\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF8, 0xBC, 0x8E,**

**0x86, 0xC3, 0xC3, 0xC3, 0x83, 0x83, 0x83, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0xFF, 0xF1, 0xC1, 0x81, 0x80, 0x80, 0x80, 0x81, 0xC3,**

**0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"6",54\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0xC3, 0xF3, 0x7F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0x7C, 0x1F, 0x07, 0x01, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"7",55\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3C, 0x7E, 0xFF, 0xE3,**

**0xC3, 0x81, 0x81, 0xC3, 0xC3, 0x7F, 0x7E, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x7C, 0xFE, 0xCE, 0x87, 0x83, 0x81, 0x81, 0x83, 0x87, 0xCE,**

**0xFE, 0x7C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00}, /\*"8",56\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0xFC, 0xFE, 0x87, 0x03,**

**0x03, 0x01, 0x01, 0x03, 0x07, 0x9E, 0xFE, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x01, 0x81, 0x83, 0x83, 0x83, 0x82, 0xC3, 0xC3, 0xE3, 0x7D,**

**0x3F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"9",57\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x70, 0xF0, 0xF0, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xE0, 0xC0, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*":",58\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x70, 0xF0, 0xF0, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0, 0xE0, 0xE0, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10, 0x18, 0x18, 0x1F, 0x0F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*";",59\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80,**

**0xC0, 0xE0, 0x60, 0x70, 0x38, 0x1C, 0x08, 0x00, 0x00, 0x00, 0x00, 0x00, 0x02, 0x06, 0x0F, 0x1F, 0x19, 0x30, 0x70, 0xE0, 0xC0, 0x80,**

**0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x00}, /\*"<",60\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18,**

**0x18, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"=",61\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x08, 0x1C, 0x38,**

**0x70, 0x60, 0xC0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0xC0, 0xE0, 0x70, 0x30, 0x19, 0x1F, 0x0F,**

**0x06, 0x02, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*">",62\*/**

**{0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x81, 0x81, 0x81, 0xC3, 0xFF, 0xFE, 0x7C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xCF, 0xCF, 0x81, 0x01, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"?",63\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0xC0, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF8, 0x1E, 0x07, 0x83,**

**0xC1, 0x60, 0x60, 0x60, 0xE0, 0xE1, 0x03, 0x0F, 0xFE, 0xF0, 0xFF, 0xFF, 0x01, 0x00, 0x7E, 0xFF, 0xC3, 0xC0, 0x60, 0x7F, 0xFF, 0xC1,**

**0xC0, 0xF8, 0x3F, 0x07, 0x00, 0x07, 0x1F, 0x38, 0x30, 0x30, 0x20, 0x20, 0x20, 0x20, 0x30, 0x30, 0x00, 0x00, 0x00, 0x00}, /\*"@",64\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xFC,**

**0x3F, 0x07, 0x0F, 0x7F, 0xFC, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xFC, 0x7F, 0x1F, 0x19, 0x18, 0x18, 0x18, 0x18, 0x1B, 0x1F,**

**0x7F, 0xFC, 0xE0, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01}, /\*"A",65\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x83,**

**0x83, 0x83, 0x83, 0x83, 0xC7, 0xFF, 0x7E, 0x18, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x81, 0x81, 0x81, 0x81, 0x81, 0xC3, 0xE7,**

**0xFE, 0x7E, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"B",66\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF8, 0xFC, 0x1E, 0x07,**

**0x07, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x07, 0x3F, 0x7F, 0xF0, 0xE0, 0xC0, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"C",67\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x07, 0x0E, 0xFC, 0xF8, 0xE0, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x80, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xC0, 0xF0,**

**0x7F, 0x3F, 0x07, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"D",68\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x83,**

**0x83, 0x83, 0x83, 0x83, 0x83, 0x83, 0x83, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00}, /\*"E",69\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x83,**

**0x83, 0x83, 0x83, 0x83, 0x83, 0x83, 0x83, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,**

**0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"F",70\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF8, 0xFC, 0x0E, 0x07,**

**0x03, 0x03, 0x83, 0x83, 0x83, 0x83, 0x83, 0x83, 0x00, 0x00, 0x00, 0x0F, 0x3F, 0x7F, 0xE0, 0xC0, 0x80, 0x80, 0x81, 0x81, 0x81, 0x81,**

**0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"G",71\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,**

**0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00}, /\*"H",72\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0xFF, 0xFF, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"I",73\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0x80, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xFF, 0x7F,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"J",74\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x80,**

**0xC0, 0xE0, 0x38, 0x1C, 0x0E, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x01, 0x07, 0x0F, 0x1C, 0x38, 0xF0, 0xE0,**

**0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"K",75\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"L",76\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFF, 0x03, 0x1F, 0xFC,**

**0xE0, 0x80, 0x80, 0xF0, 0x7C, 0x0F, 0xFF, 0xFF, 0xFC, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x03, 0x07, 0x07, 0x01, 0x00, 0x00,**

**0x01, 0xFF, 0xFF, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00}, /\*"M",77\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x0F, 0x3E,**

**0xF8, 0xE0, 0x80, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x3E, 0xF8, 0xE0,**

**0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"N",78\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xFC, 0x7E, 0x0F, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x0F, 0x7E, 0xFC, 0xF0, 0x00, 0x00, 0x1F, 0x7F, 0xFC, 0xE0, 0xC0, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xE0,**

**0xFC, 0x3F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"O",79\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x07, 0x8E, 0xFE, 0xFC, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x06, 0x06, 0x06, 0x06, 0x07, 0x03, 0x03,**

**0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"P",80\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xFC, 0x7E, 0x0F, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x0F, 0x7E, 0xFC, 0xF0, 0x00, 0x00, 0x1F, 0x7F, 0xFC, 0xE0, 0xC0, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xE0,**

**0xFC, 0x3F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x0F, 0x1F, 0x19, 0x39, 0x30, 0x30, 0x30, 0x18, 0x10}, /\*"Q",81\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x83,**

**0x83, 0x83, 0x83, 0x83, 0xC7, 0xFE, 0x7E, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x01, 0x01, 0x03, 0x03, 0x0F, 0x3E, 0xF8,**

**0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00}, /\*"R",82\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3C, 0x7E, 0xEF, 0xC3,**

**0xC3, 0x83, 0x83, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x81, 0x81, 0x81, 0x83, 0x83, 0xC7, 0xFF,**

**0xFE, 0x7C, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"S",83\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"T",84\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x7F, 0xFF, 0xC0, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xE0,**

**0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"U",85\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x0F, 0x7F, 0xFE, 0xF0, 0x80,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xFC, 0x3F, 0x0F, 0x01, 0x00, 0x00, 0x00, 0x01, 0x0F, 0x7F, 0xFC, 0xE0, 0xE0, 0xFC, 0x7F, 0x0F,**

**0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"V",86\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00,**

**0x80, 0xC0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x07, 0xFF, 0xFF, 0xE0, 0xFC, 0x1F, 0x03, 0x07, 0x3F, 0xFC, 0xE0,**

**0xFE, 0xFF, 0x07, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"W",87\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x0F, 0x1E, 0x7C,**

**0xF0, 0xC0, 0xE0, 0xF0, 0x7C, 0x1E, 0x0F, 0x03, 0x01, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0x78, 0x3C, 0x0F, 0x07, 0x07, 0x0F, 0x3C, 0xF8,**

**0xE0, 0xC0, 0x80, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00}, /\*"X",88\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x0F, 0x3E, 0x7C, 0xF0,**

**0xC0, 0x00, 0x00, 0xC0, 0xF0, 0x78, 0x1E, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0xFF, 0xFF, 0x03, 0x01, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"Y",89\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x83, 0xC3, 0xF3, 0x7B, 0x1F, 0x0F, 0x07, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xF8, 0xBC, 0x8F, 0x87, 0x83, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"Z",90\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x20, 0x20, 0x20, 0x20, 0x20, 0x00, 0x00, 0x00, 0x00}, /\*"[",91\*/**

**{0x00, 0x00, 0x40, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x3F,**

**0xF8, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x3E, 0xF8, 0xC0,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x0F, 0x0C, 0x00, 0x00}, /\*"\",92\*/**

**{0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x20, 0x20, 0x20, 0x20, 0x20, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"]",93\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0x70, 0x3C,**

**0x0F, 0x03, 0x07, 0x0E, 0x3C, 0xF0, 0xE0, 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, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"^",94\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30}, /\*"\_",95\*/**

**{0x00, 0x00, 0x00, 0x40, 0xC0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01,**

**0x01, 0x03, 0x02, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"`",96\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0x30, 0x30,**

**0x30, 0x10, 0x30, 0x30, 0x30, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0xF8, 0xFC, 0xDC, 0x8E, 0x86, 0x86, 0x86, 0xC6, 0xC6, 0xFF,**

**0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"a",97\*/**

**{0x00, 0x00, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x60,**

**0x70, 0x30, 0x30, 0x30, 0x30, 0x70, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x80, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xE0,**

**0x7F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"b",98\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xE0,**

**0x70, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1F, 0x7F, 0xFF, 0xE0, 0xC0, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00}, /\*"c",99\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xE0, 0x30,**

**0x30, 0x30, 0x30, 0x30, 0x30, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x7F, 0xFF, 0xE0, 0x80, 0x80, 0x80, 0x80, 0xC0, 0x60, 0xFF,**

**0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"d",100\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xE0, 0x30,**

**0x30, 0x10, 0x10, 0x30, 0x30, 0xF0, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x3F, 0xFF, 0xE6, 0xC6, 0x86, 0x86, 0x86, 0x86, 0x86, 0x86,**

**0x87, 0x87, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"e",101\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0,**

**0xFF, 0xFF, 0xC3, 0xC1, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"f",102\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0x30,**

**0x10, 0x10, 0x10, 0x30, 0x70, 0xF0, 0xF0, 0x30, 0x30, 0x00, 0x00, 0x00, 0x73, 0xFF, 0xDF, 0x8C, 0x98, 0x98, 0x98, 0x8C, 0x8E, 0x8F,**

**0x87, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x1E, 0x3F, 0x31, 0x31, 0x21, 0x61, 0x61, 0x21, 0x31, 0x31, 0x3F, 0x1F, 0x06, 0x00}, /\*"g",103\*/**

**{0x00, 0x00, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x60,**

**0x30, 0x30, 0x30, 0x30, 0x30, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"h",104\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0x30, 0x30,**

**0x31, 0xF3, 0xF3, 0xF1, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0xFF, 0xFF, 0xFF, 0x80, 0x80,**

**0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"i",105\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0x30, 0x30, 0x30,**

**0x30, 0x30, 0x30, 0xF3, 0xF3, 0xF3, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0x30, 0x30, 0x60, 0x60, 0x30, 0x30, 0x3F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00}, /\*"j",106\*/**

**{0x00, 0x00, 0x00, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00,**

**0x00, 0x80, 0x80, 0xC0, 0xE0, 0x70, 0x30, 0x10, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x02, 0x07, 0x1F, 0x3D, 0x79, 0xF0, 0xE0,**

**0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x00}, /\*"k",107\*/**

**{0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0xFF, 0xFF, 0xFF, 0x80, 0x80,**

**0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"l",108\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xC0, 0x70, 0x30,**

**0x30, 0xF0, 0xE0, 0xE0, 0x30, 0x10, 0xF0, 0xF0, 0xC0, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00}, /\*"m",109\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xE0, 0x60,**

**0x30, 0x30, 0x30, 0x30, 0x30, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"n",110\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0x70, 0x30,**

**0x30, 0x30, 0x30, 0x30, 0x30, 0x70, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x0E, 0x7F, 0xFF, 0xE0, 0x80, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xE0,**

**0xFF, 0x3F, 0x0E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"o",111\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0x60,**

**0x70, 0x30, 0x30, 0x30, 0x30, 0x70, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x80, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xE0,**

**0x7F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"p",112\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xE0, 0x30,**

**0x30, 0x30, 0x10, 0x30, 0x30, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x7F, 0xFF, 0xE0, 0x80, 0x80, 0x80, 0x80, 0xC0, 0x60, 0xFF,**

**0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x00, 0x00}, /\*"q",113\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xC0,**

**0x60, 0x30, 0x30, 0x30, 0x30, 0x30, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"r",114\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF0, 0x30,**

**0x30, 0x10, 0x10, 0x10, 0x30, 0x30, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x81, 0x83, 0x87, 0x86, 0x86, 0x8E, 0x8C, 0x9C, 0xFC,**

**0xF8, 0x20, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"s",115\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0x30, 0x30, 0x30, 0xFF,**

**0xFF, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7F, 0xFF, 0xC0, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"t",116\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x3F, 0xFF, 0xFF, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xE0, 0xFF,**

**0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"u",117\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10, 0x70, 0xF0, 0xC0, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0x70, 0x10, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x7F, 0xF8, 0xE0, 0xE0, 0xF8, 0x3F, 0x0F,**

**0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"v",118\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0x00, 0x00, 0x00,**

**0x00, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0x00, 0x00, 0x03, 0xFF, 0xFF, 0xC0, 0xF0, 0x3E, 0x07, 0x0F, 0x7E, 0xF8, 0xC0,**

**0xFF, 0xFF, 0x03, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00}, /\*"w",119\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10, 0x70, 0xF0, 0xE0,**

**0x80, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0x70, 0x10, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0x71, 0x3F, 0x1F, 0x1F, 0x3F, 0x71, 0xE0,**

**0xC0, 0x80, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x00}, /\*"x",120\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10, 0x70, 0xF0, 0xC0, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0x70, 0x10, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x3F, 0xFC, 0xE0, 0xE0, 0xF8, 0x7F, 0x0F,**

**0x03, 0x00, 0x00, 0x00, 0x00, 0x30, 0x30, 0x30, 0x30, 0x38, 0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"y",121\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0x30, 0x30,**

**0x30, 0x30, 0x30, 0xB0, 0xF0, 0xF0, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF0, 0xBC, 0x9E, 0x87, 0x83, 0x81, 0x80,**

**0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00}, /\*"z",122\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFE, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x07, 0x07, 0xFD, 0xFD, 0xF0, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x3F, 0x30, 0x30, 0x20, 0x20, 0x00, 0x00, 0x00}, /\*"{",123\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF8, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"|",124\*/**

**{0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x7F, 0xFF, 0xFE, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xFD, 0xFD, 0x07, 0x03,**

**0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x20, 0x20, 0x30, 0x30, 0x3F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"}",125\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80,**

**0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0E, 0x0F, 0x03, 0x01, 0x01, 0x03, 0x07, 0x0E, 0x0C, 0x0C, 0x0C,**

**0x0C, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"~",126\*/**

**};**

**// 32\*64 ASCII字符集点阵(Consolas)**

**const unsigned char asc2\_6432[][256] PROGMEM = {**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*" ",32\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0,**

**0xF0, 0xF0, 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, 0xFF, 0xFF, 0xFF, 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, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x01, 0x00, 0x00, 0x00, 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, 0x00, 0x00, 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, 0xF0, 0xF0, 0xF8, 0xF8, 0xF0, 0xF0, 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, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"!",33\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x10, 0x00, 0x00,**

**0x00, 0x00, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x10, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*""",34\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xF8, 0xFF, 0xFF, 0xFF, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x07, 0x07, 0xE7, 0xFF, 0xFF, 0xFF, 0x3F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0xFF, 0xFF, 0xFF, 0xFF,**

**0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xFF, 0xFF, 0xFF, 0xFF, 0xE0, 0xE0, 0xE0, 0xE0,**

**0xE0, 0xE0, 0xE0, 0xFF, 0xFF, 0xFF, 0xFF, 0xE3, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFF, 0xFF, 0xFF, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"#",35\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80,**

**0xF0, 0xFC, 0xFC, 0xFC, 0x9C, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF0, 0xF8, 0xFC, 0xFE,**

**0xFE, 0x1F, 0x0F, 0x0F, 0x0F, 0x07, 0x07, 0xFF, 0xFF, 0xFF, 0xFF, 0x0F, 0x07, 0x07, 0x07, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x01, 0x0F, 0x1F, 0x3F, 0x7F, 0xFF, 0xFC, 0xF8, 0xF0, 0xF0, 0xE0, 0xFC, 0xFF, 0xFF, 0xFF, 0x87, 0x80, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0xFF, 0xFF, 0xFF,**

**0xFF, 0x07, 0x0F, 0x0F, 0x1F, 0x1F, 0x3F, 0xFE, 0xFC, 0xFC, 0xF8, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xE0, 0xE0, 0xE0, 0xE0,**

**0xC0, 0xC0, 0xC0, 0xC0, 0xFF, 0xFF, 0xFF, 0xFF, 0xC1, 0xC0, 0xE0, 0xE0, 0xE0, 0xF0, 0xF8, 0xFF, 0x7F, 0x7F, 0x3F, 0x0F, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x03, 0x03, 0x03, 0xC3, 0xFF, 0xFF, 0xFF, 0x7F, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x01, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"$",36\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0xF0, 0xF0, 0xF0, 0xE0, 0xC0, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF0, 0xF0, 0x70, 0x10, 0x00, 0x00, 0x7E, 0xFF, 0xFF, 0xFF, 0xC3, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xC1, 0xFF, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xF0, 0xFC, 0x7E, 0x3F, 0x1F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x01, 0x03, 0x07, 0x07, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x07, 0x07, 0x83, 0xC1, 0xF0, 0xF8, 0xFE, 0x7F, 0x1F, 0x0F, 0x03, 0x01, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xF0, 0xFC, 0x7E, 0x3F, 0x1F, 0x07,**

**0x83, 0xE0, 0xF0, 0xF8, 0x78, 0x38, 0x3C, 0x3C, 0x3C, 0x78, 0xF8, 0xF8, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF8, 0xFE, 0x7F,**

**0x1F, 0x0F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x7F, 0xFF, 0xFF, 0xFF, 0xE0, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xF0, 0xFF, 0xFF, 0x7F, 0x1F, 0x00,**

**0x00, 0x02, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"%",37\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xC0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0,**

**0xE0, 0xE0, 0xE0, 0xC0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFF, 0xFF,**

**0xFF, 0xFF, 0x07, 0x03, 0x01, 0x01, 0x01, 0x01, 0x01, 0x03, 0x87, 0xFF, 0xFF, 0xFF, 0xFF, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x87, 0x9F, 0xFF, 0xFF, 0xFF, 0xFC, 0xF8, 0xF0, 0xF8, 0xFC, 0x3C, 0x3E, 0x1F, 0x0F, 0x0F, 0x07, 0x01, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xFC, 0xFE, 0xFF, 0xFF, 0x1F, 0x07, 0x03, 0x03, 0x03, 0x07, 0x0F, 0x1F, 0x3F,**

**0x7F, 0xFE, 0xFC, 0xF8, 0xE0, 0xC0, 0x80, 0xE0, 0xFF, 0xFF, 0xFF, 0xFF, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1F, 0x3F, 0xFF, 0xFF, 0xFF, 0xF8,**

**0xE0, 0xE0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC1, 0xE3, 0xF7, 0xFF, 0xFF, 0x7F, 0xFF, 0xFF, 0xFF, 0xF7, 0xE0, 0xC0, 0x80, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00,**

**0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"&",38\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0,**

**0xF0, 0xF0, 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, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"'",39\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x80, 0xC0, 0xE0, 0xF0, 0xF8, 0xFC, 0x78, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xE0, 0xF8, 0xFC, 0xFF, 0x7F, 0x1F, 0x0F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xFE, 0xFF, 0xFF, 0xFF, 0x0F, 0x01, 0x00, 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, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x07, 0x3F, 0xFF, 0xFF, 0xFF, 0xF8, 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, 0x00, 0x01, 0x07, 0x0F, 0x3F, 0x7F, 0xFF, 0xFC, 0xF8, 0xE0, 0xC0, 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,**

**0x00, 0x01, 0x03, 0x07, 0x0F, 0x1F, 0x0F, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"(",40\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0x78, 0xFC, 0xF8, 0xF0, 0xE0, 0xC0, 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, 0x00, 0x01, 0x07, 0x0F, 0x3F, 0xFF, 0xFF, 0xFC, 0xF8, 0xE0, 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, 0x00, 0x03, 0x1F, 0xFF, 0xFF, 0xFF, 0xFE, 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, 0xFF, 0xFF, 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, 0xC0, 0xF8, 0xFF, 0xFF, 0xFF, 0x3F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF8, 0xFC, 0xFF, 0x7F, 0x1F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x04, 0x0E, 0x1F, 0x1F, 0x0F, 0x07, 0x03, 0x01, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*")",41\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0,**

**0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x0F, 0x1E,**

**0x1E, 0x3C, 0x38, 0x78, 0x70, 0xE0, 0xEF, 0xFF, 0xFF, 0xE3, 0xF0, 0x70, 0x78, 0x38, 0x3C, 0x1E, 0x1E, 0x0F, 0x0C, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x18, 0x78, 0x3C, 0x1C, 0x1E, 0x0E, 0x0F, 0x07, 0x03, 0xFB, 0xFF, 0xFF, 0xF3, 0x03, 0x07, 0x0F, 0x0E, 0x1E, 0x3C,**

**0x3C, 0x38, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07,**

**0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"\*",42\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 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, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0xFF, 0xFF,**

**0xFF, 0xFF, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 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, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"+",43\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x7C, 0xFC, 0xFE, 0xFE, 0xFE, 0xFC, 0xF8, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xE1, 0xFF, 0xFF, 0xFF, 0x7F, 0x3F, 0x0F, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x07, 0x07, 0x03, 0x03, 0x01, 0x01,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*",",44\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3E, 0x3E, 0x3E, 0x3E, 0x3E, 0x3E, 0x3E, 0x3E, 0x3E,**

**0x3E, 0x3E, 0x3E, 0x3E, 0x3E, 0x3E, 0x3E, 0x3E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"-",45\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xF0, 0xF8, 0xFC, 0xFC, 0xFC, 0xFC, 0xFC, 0xFC, 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, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*".",46\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF0, 0xF0, 0xF0, 0xF0, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF8, 0xFE, 0xFF, 0x7F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xFC, 0xFF, 0xFF, 0x3F, 0x0F, 0x01, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF8, 0xFE, 0xFF, 0x7F,**

**0x1F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0,**

**0xF0, 0xFC, 0xFF, 0xFF, 0x3F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xE0, 0xF8, 0xFF, 0xFF, 0x7F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"/",47\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0xC0,**

**0xC0, 0xC0, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xF8, 0xFC,**

**0xFE, 0x3F, 0x1F, 0x0F, 0x07, 0x07, 0x07, 0x03, 0x03, 0x07, 0x07, 0x07, 0x0F, 0x1F, 0x7F, 0xFE, 0xFE, 0xFC, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFE, 0xFF, 0xFF, 0xFF, 0x7F, 0x01, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0xC0, 0xE0, 0xE0, 0xF0, 0xF8, 0xF8, 0x7C, 0x3E, 0x3F,**

**0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFC, 0xF8, 0x7C, 0x7C, 0x3E, 0x1F, 0x1F, 0x0F, 0x07,**

**0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x0F, 0x1F, 0x7F, 0xFF,**

**0xFF, 0xFC, 0xF0, 0xE0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xF0, 0xFC, 0xFF, 0x7F, 0x3F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"0",48\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0xF0, 0xF8, 0xF8,**

**0x7C, 0x7C, 0x3E, 0x3E, 0x1F, 0x1F, 0xFF, 0xFF, 0xFF, 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, 0xFF, 0xFF, 0xFF, 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, 0xFF, 0xFF,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0,**

**0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"1",49\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xC0,**

**0xC0, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x3E, 0x3E,**

**0x1F, 0x0F, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x0F, 0x3F, 0xFF, 0xFF, 0xFE, 0xFC, 0xF0, 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, 0x80, 0xE0, 0xFF, 0xFF, 0xFF,**

**0x7F, 0x3F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF0, 0xF8,**

**0xFC, 0x7E, 0x3F, 0x1F, 0x0F, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0, 0xF0, 0xF8,**

**0xFC, 0xFE, 0xFF, 0xDF, 0xCF, 0xC7, 0xC3, 0xC1, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"2",50\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xC0, 0xC0, 0xC0,**

**0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x07,**

**0x07, 0x07, 0x07, 0x07, 0x07, 0x03, 0x03, 0x07, 0x07, 0x07, 0x07, 0x0F, 0x3F, 0xFF, 0xFF, 0xFE, 0xFC, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xE0, 0xF0, 0x7C, 0x7F, 0x3F, 0x1F,**

**0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x07, 0x07, 0x0F, 0x0F, 0x3F, 0xFE, 0xFE, 0xFC, 0xF8, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0,**

**0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xF0, 0xF8, 0xFE, 0x7F, 0x7F, 0x3F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x01, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"3",51\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF8, 0xFE, 0x7F, 0x3F, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xF8, 0xFC, 0xFF, 0x3F, 0x1F, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xF8, 0xFE, 0xFF, 0xFF, 0xCF, 0xC7, 0xC1, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0,**

**0xC0, 0xC0, 0xC0, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"4",52\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0xFF, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xC0, 0xC0, 0xC0, 0x80,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,**

**0x01, 0x01, 0x03, 0x03, 0x07, 0x0F, 0xFF, 0xFF, 0xFF, 0xFE, 0xFC, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0,**

**0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xE0, 0xF0, 0xFC, 0xFF, 0x7F, 0x3F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"5",53\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0,**

**0xF0, 0xF8, 0xFC, 0x7E, 0x3E, 0x1F, 0x0F, 0x0F, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0xF8, 0xFF, 0xFF, 0xFF, 0xFF, 0xC3, 0xC0, 0xE0, 0xE0, 0xE0, 0xE0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xE0, 0xE0, 0xE0, 0xE0,**

**0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x01, 0x01, 0x01, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x03, 0x0F, 0xFF, 0xFF, 0xFF, 0xFE, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x1F, 0x3F, 0xFF,**

**0xFF, 0xFC, 0xF0, 0xE0, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0x80, 0xC0, 0xC0, 0xC0, 0xE0, 0xF0, 0xFC, 0xFF, 0x7F, 0x3F, 0x1F, 0x03, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"6",54\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x07, 0x07,**

**0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x87, 0xE7, 0xFF, 0xFF, 0xFF, 0x7F, 0x1F, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xF8, 0xFE, 0xFF, 0x7F, 0x1F,**

**0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xFC,**

**0xFF, 0xFF, 0x7F, 0x1F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x80, 0xC0, 0xF0, 0xFC, 0xFF, 0xFF, 0x7F, 0x1F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x02, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"7",55\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xC0,**

**0xC0, 0xC0, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF8, 0xFC, 0xFE,**

**0xFF, 0x1F, 0x0F, 0x07, 0x07, 0x07, 0x03, 0x03, 0x03, 0x03, 0x07, 0x07, 0x07, 0x0F, 0x3F, 0xFF, 0xFF, 0xFE, 0xFC, 0xE0, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x1F, 0x3F, 0x7F, 0x7F, 0xFC, 0xF8, 0xF0, 0xF0, 0xE0, 0xE0, 0xC0, 0xE0, 0xE0, 0xF0, 0xF8, 0x7C, 0x7F, 0x3F,**

**0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF0, 0xF8, 0xFC, 0xFE, 0x3E, 0x1F, 0x0F, 0x0F, 0x07, 0x07, 0x03,**

**0x03, 0x07, 0x07, 0x0F, 0x1F, 0x3F, 0x7E, 0xFE, 0xFC, 0xF8, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x3F, 0x7F, 0xFF, 0xFF,**

**0xFF, 0xF0, 0xE0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xF0, 0xFF, 0xFF, 0xFF, 0x7F, 0x1F, 0x03, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"8",56\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xC0,**

**0xC0, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xF8, 0xFC, 0xFE,**

**0x7F, 0x1F, 0x0F, 0x07, 0x07, 0x03, 0x03, 0x03, 0x03, 0x03, 0x07, 0x07, 0x0F, 0x1F, 0xFF, 0xFE, 0xFC, 0xF8, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x3F, 0xFF, 0xFF, 0xFF, 0xFF, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0xFF,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x07, 0x0F, 0x0F, 0x0F, 0x0F, 0x1E, 0x1E, 0x0E,**

**0x0E, 0x0E, 0x0F, 0x0F, 0x0F, 0x07, 0x87, 0xFF, 0xFF, 0xFF, 0xFF, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0,**

**0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xE0, 0xE0, 0xF0, 0xF0, 0xF8, 0x7C, 0x7E, 0x3F, 0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"9",57\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 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, 0x1F, 0x3F, 0x7F, 0x7F, 0x7F, 0x7F, 0x3F, 0x1F, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0xF0, 0xF8, 0xFC, 0xFC, 0xFC, 0xF8, 0xF8, 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, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*":",58\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 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, 0x1F, 0x3F, 0x7F, 0x7F, 0x7F, 0x7F, 0x3F, 0x1F, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x78, 0xFC, 0xFC, 0xFE, 0xFE, 0xFC, 0xFC, 0xF8, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xE0, 0xF3, 0xFF, 0xFF, 0xFF, 0x3F, 0x1F, 0x03, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x07, 0x07, 0x03, 0x03, 0x03, 0x01,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*";",59\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xE0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xE0, 0xF0, 0xF8, 0x7C, 0x7E, 0x3E, 0x1F, 0x0F, 0x0F, 0x07, 0x03, 0x01,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x08, 0x1C, 0x1E, 0x3F, 0x7F, 0xFF, 0xFF, 0xF7, 0xE3, 0xC1, 0xC0, 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, 0x01, 0x01, 0x03, 0x07, 0x0F, 0x1F, 0x1F, 0x3E, 0x7C, 0xFC, 0xF8, 0xF0, 0xE0, 0xC0, 0xC0, 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, 0x01, 0x01, 0x03, 0x03,**

**0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"<",60\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 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, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0,**

**0xE0, 0xE0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC1, 0xC1, 0xC1, 0xC1, 0xC1, 0xC1, 0xC1, 0xC1, 0xC1, 0xC1, 0xC1, 0xC1,**

**0xC1, 0xC1, 0xC1, 0xC1, 0xC1, 0xC1, 0xC1, 0xC1, 0xC1, 0xC1, 0xC1, 0xC1, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"=",61\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0,**

**0xE0, 0xE0, 0xE0, 0xC0, 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, 0x01, 0x01, 0x03, 0x07, 0x0F, 0x0F, 0x1F, 0x3E, 0x7E, 0xFC, 0xF8, 0xF0, 0xE0, 0xE0, 0xC0, 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,**

**0x80, 0xC0, 0xC1, 0xE3, 0xF7, 0xFF, 0xFF, 0x7F, 0x3E, 0x1E, 0x1C, 0x08, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0,**

**0xE0, 0xE0, 0xF0, 0xF8, 0xFC, 0x7C, 0x3E, 0x1F, 0x0F, 0x0F, 0x07, 0x03, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*">",62\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xE0, 0xE0,**

**0xE0, 0xE0, 0xC0, 0xC0, 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, 0x01, 0x01, 0x01, 0x03, 0x03, 0x07, 0x0F, 0x3F, 0xFF, 0xFF, 0xFE, 0xF8, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xF0, 0xFC, 0xFF, 0x7F, 0x7F,**

**0x3F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0xFF, 0xFF, 0xFF, 0xFF,**

**0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xE0, 0xF0, 0xF0, 0xF8, 0xF8, 0xF0, 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, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"?",63\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xC0, 0xE0, 0xE0, 0xF0, 0x70,**

**0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xF0, 0xE0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xFC, 0xFE,**

**0x3F, 0x1F, 0x07, 0x03, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x0F, 0x3F, 0xFF, 0xFE, 0xF8, 0xC0, 0x00,**

**0x00, 0xC0, 0xFC, 0xFF, 0xFF, 0x3F, 0x07, 0x00, 0x00, 0x00, 0xE0, 0xF0, 0xF8, 0xFC, 0x7C, 0x1E, 0x1E, 0x1E, 0x1E, 0xFE, 0xFC, 0xFC, 0xFE, 0x3E,**

**0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0xF0, 0xFF, 0xFF, 0xFF, 0x0F, 0x00, 0x00, 0x00, 0xF8, 0xFF, 0xFF, 0xFF, 0x7F, 0x01, 0x00, 0x00,**

**0x00, 0x00, 0xFC, 0xFF, 0xFF, 0xFF, 0x1F, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x1F, 0xFF, 0xFF, 0xFF, 0xE0, 0x00, 0x00, 0x00,**

**0x1F, 0x7F, 0xFF, 0xFF, 0xFE, 0xE0, 0xE0, 0xF0, 0x78, 0x3F, 0x1F, 0x7F, 0xFF, 0xFF, 0xE0, 0xE0, 0xF0, 0xF8, 0x7E, 0x7F, 0x1F, 0x0F, 0x01, 0x00,**

**0x00, 0x07, 0x3F, 0xFF, 0xFF, 0xFC, 0xE0, 0xC0, 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, 0x00, 0x01, 0x03, 0x07, 0x0F, 0x0F, 0x0F, 0x1E, 0x1E, 0x1C, 0x1C, 0x1C, 0x1C,**

**0x1C, 0x1C, 0x1C, 0x1E, 0x0E, 0x0E, 0x0F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"@",64\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xE0, 0xFC, 0xFF, 0xFF, 0x7F, 0x0F, 0x1F, 0xFF, 0xFF, 0xFF, 0xFC, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF8, 0xFF, 0xFF, 0xFF, 0x1F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x07, 0x3F, 0xFF, 0xFF, 0xFF, 0xF8,**

**0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF8, 0xFF, 0xFF, 0xFF, 0xFF, 0xC7, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0,**

**0xC0, 0xC0, 0xC0, 0xC0, 0xC1, 0xCF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF8, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF0, 0xFE, 0xFF, 0xFF, 0x7F, 0x0F,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x1F, 0xFF, 0xFF, 0xFF, 0xFE, 0xF0, 0x80, 0x00,**

**0x02, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"A",65\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF,**

**0xFF, 0xFF, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x0F, 0x0F, 0x1F, 0xFF, 0xFE, 0xFE, 0xFC, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xE0, 0xF0, 0xF8, 0x7F, 0x3F,**

**0x3F, 0x1F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x07, 0x07, 0x0F, 0x1F, 0xFF, 0xFF, 0xFE, 0xFC, 0xF8, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF,**

**0xFF, 0xFF, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xE0, 0xF0, 0xF8, 0xFF, 0x7F, 0x7F, 0x3F, 0x0F, 0x03, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x01, 0x01, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"B",66\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF8,**

**0xFC, 0xFE, 0x7E, 0x3F, 0x1F, 0x0F, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x0F, 0x0F, 0x1F, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFC, 0xFF, 0xFF, 0xFF, 0xFF, 0x0F, 0x01, 0x00, 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, 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, 0x00, 0x07, 0x0F, 0x3F, 0x7F,**

**0x7F, 0xFE, 0xFC, 0xF8, 0xF0, 0xE0, 0xE0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xE0, 0xF0, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"C",67\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,**

**0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x0F, 0x0F, 0x0F, 0x1F, 0x3F, 0x7E, 0xFE, 0xFC, 0xF8, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01,**

**0x7F, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xFC, 0xFF, 0xFF, 0xFF, 0xFF, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,**

**0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xE0, 0xE0, 0xF0, 0xF8, 0xFC, 0x7E, 0x7F, 0x3F, 0x1F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"D",68\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0,**

**0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0xFF, 0xFF, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"E",69\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0,**

**0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**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, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"F",70\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80,**

**0x80, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xF0, 0xF8,**

**0xFC, 0xFE, 0x3F, 0x1F, 0x1F, 0x0F, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xF8, 0xFF, 0xFF, 0xFF, 0xFF, 0x1F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0,**

**0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7F, 0xFF, 0xFF, 0xFF, 0xFF, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x1F, 0x3F, 0x7F,**

**0xFF, 0xFC, 0xF8, 0xF0, 0xE0, 0xE0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"G",71\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0,**

**0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"H",72\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07,**

**0x07, 0x07, 0x07, 0x07, 0x07, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 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, 0xFF, 0xFF, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0,**

**0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"I",73\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07,**

**0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0xFF, 0xFF, 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, 0xFF, 0xFF, 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, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xE0,**

**0xE0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xF0, 0xFC, 0xFF, 0xFF, 0x7F, 0x1F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"J",74\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF,**

**0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF8, 0xFC, 0x7E, 0x3F, 0x1F, 0x0F, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x80, 0xC0, 0xE0, 0xF0, 0xFC, 0x7E, 0x3F, 0x1F, 0x0F, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x01, 0x03, 0x07, 0x1F, 0x3F, 0x7F, 0xFE,**

**0xFC, 0xF8, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF,**

**0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0x0F, 0x1F, 0x7F, 0xFF, 0xFE, 0xF8, 0xF0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x02, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"K",75\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 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, 0x00, 0xFF,**

**0xFF, 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, 0xFF, 0xFF, 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, 0xFF, 0xFF, 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, 0xFF,**

**0xFF, 0xFF, 0xFF, 0xFF, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"L",76\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x3F,**

**0xFF, 0xFF, 0xFC, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF0, 0xFC, 0xFF, 0x7F, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xE0, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x01, 0x07, 0x3F, 0xFF, 0xFF, 0xFC, 0xE0, 0x00, 0x80, 0xF0, 0xFE, 0xFF, 0x3F, 0x0F, 0x01, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xC0, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x3F, 0x7F, 0x7F,**

**0x7F, 0x3F, 0x0F, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"M",77\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,**

**0x3F, 0xFF, 0xFF, 0xFC, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x07, 0x1F, 0x7F, 0xFF, 0xFF, 0xFC, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x1F,**

**0x7F, 0xFF, 0xFF, 0xFC, 0xF0, 0x80, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x1F, 0x7F, 0xFF, 0xFE, 0xF8, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"N",78\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0xC0,**

**0xC0, 0xC0, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF8, 0xFC, 0xFE,**

**0xFE, 0x3F, 0x1F, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x0F, 0x1F, 0x3F, 0x7F, 0xFE, 0xFC, 0xF8, 0xF0, 0xC0, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFE, 0xFF, 0xFF, 0xFF, 0xFF, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x03, 0xFF, 0xFF, 0xFF, 0xFF, 0xFE, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xFF, 0xFF, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x3F, 0x7F, 0xFF,**

**0xFC, 0xF8, 0xF0, 0xE0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xF0, 0xF8, 0xFE, 0x7F, 0x7F, 0x1F, 0x0F, 0x03, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"O",79\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF,**

**0xFF, 0xFF, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x0F, 0x0F, 0x1F, 0x3F, 0xFE, 0xFE, 0xFC, 0xF8, 0xF0, 0x80, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xF9,**

**0xFF, 0xFF, 0xFF, 0x7F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E,**

**0x1E, 0x1E, 0x1E, 0x1F, 0x0F, 0x0F, 0x0F, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 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, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"P",80\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0xC0,**

**0xC0, 0xC0, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF8, 0xFC, 0xFE,**

**0xFE, 0x3F, 0x1F, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x0F, 0x1F, 0x3F, 0x7F, 0xFE, 0xFC, 0xF8, 0xF0, 0xC0, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFC, 0xFF, 0xFF, 0xFF, 0xFF, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x03, 0xFF, 0xFF, 0xFF, 0xFF, 0xFE, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xFF, 0xFF, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x3F, 0x7F, 0xFF,**

**0xFC, 0xF8, 0xF0, 0xE0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xF0, 0xF8, 0xFE, 0x7F, 0x7F, 0x1F, 0x0F, 0x03, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x0F, 0x7F, 0xFF, 0xFF, 0xFF, 0xF3, 0xC3, 0x83, 0x81, 0x81, 0x00,**

**0x00, 0x80, 0x80, 0x80, 0xC0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x01, 0x03, 0x07, 0x07, 0x07, 0x07, 0x0F, 0x0F, 0x0F, 0x0F, 0x07, 0x07, 0x07, 0x03, 0x03, 0x01}, /\*"Q",81\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF,**

**0xFF, 0xFF, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x0F, 0x0F, 0x1F, 0x7F, 0xFF, 0xFE, 0xFC, 0xF8, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x80, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xF0, 0xFC, 0xFF, 0x7F,**

**0x3F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x07, 0x07,**

**0x0F, 0x1F, 0x7F, 0xFF, 0xFD, 0xF8, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF,**

**0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x1F, 0x7F, 0xFF, 0xFF, 0xFC, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x02, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"R",82\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0xC0,**

**0xC0, 0xC0, 0xC0, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF8, 0xFC, 0xFE,**

**0xFF, 0x1F, 0x0F, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x0F, 0x1F, 0x3F, 0x7F, 0xFF, 0xFC, 0xF8, 0xF0, 0xF0, 0xE0, 0xE0, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x03, 0x03, 0x07,**

**0x07, 0x07, 0x0F, 0x0F, 0x1F, 0x3F, 0x7F, 0xFE, 0xFC, 0xFC, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0, 0xE0, 0xC0, 0xC0,**

**0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xF0, 0xF8, 0xFF, 0xFF, 0x7F, 0x3F, 0x0F, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"S",83\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07,**

**0x07, 0x07, 0x07, 0x07, 0x07, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 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, 0xFF, 0xFF, 0xFF,**

**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, 0xFF, 0xFF, 0xFF, 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, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"T",84\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x1F, 0x7F, 0xFF, 0xFF,**

**0xFC, 0xF0, 0xE0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xE0, 0xF8, 0xFE, 0xFF, 0x7F, 0x3F, 0x0F, 0x01, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"U",85\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x07, 0x3F, 0xFF, 0xFF, 0xFF, 0xFE, 0xF0,**

**0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF0, 0xFC, 0xFF, 0xFF, 0xFF, 0x1F, 0x03, 0x00,**

**0x00, 0x00, 0x00, 0x01, 0x07, 0x3F, 0xFF, 0xFF, 0xFF, 0xFC, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xFC, 0xFF,**

**0xFF, 0xFF, 0x3F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x0F, 0x7F, 0xFF, 0xFF, 0xFF, 0xF8, 0xC0, 0x00, 0x00,**

**0x00, 0x00, 0xC0, 0xF8, 0xFF, 0xFF, 0xFF, 0x3F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x03, 0x1F, 0x7F, 0xFF, 0xFF, 0xFE, 0xF0, 0xF0, 0xFE, 0xFF, 0xFF, 0x3F, 0x0F, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"V",86\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x03, 0xFF, 0xFF, 0xFF, 0xFF, 0x80, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF8, 0xF8, 0xF8, 0xF8, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFC, 0xFF, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x80, 0x00, 0x00, 0xE0, 0xFC, 0xFF, 0xFF, 0x1F, 0x03,**

**0x0F, 0x7F, 0xFF, 0xFF, 0xFC, 0xE0, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1F, 0xFF, 0xFF, 0xFF, 0xFF,**

**0xF8, 0xFF, 0xFF, 0x3F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x7F, 0xFF, 0xFF, 0xF8, 0xE0, 0xFF, 0xFF, 0xFF, 0x1F, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"W",87\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0x1F, 0x3F, 0xFF,**

**0xFF, 0xFC, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0xFC, 0xFF, 0xFF, 0x3F, 0x1F, 0x07, 0x03, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0x1F, 0x3F, 0xFF, 0xFF, 0xFC, 0xF8, 0xF8, 0xFC, 0xFF, 0x7F, 0x1F, 0x0F, 0x07, 0x01,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xF8, 0xFC, 0xFE, 0x7F, 0x3F, 0x0F,**

**0x1F, 0x3F, 0xFF, 0xFE, 0xFC, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF8, 0xFE, 0xFF,**

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

**0x00, 0x02, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x02, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"X",88\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x01, 0x07, 0x1F, 0x3F, 0xFF, 0xFF, 0xFE,**

**0xF8, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0xFC, 0xFF, 0xFF, 0x3F, 0x1F, 0x07, 0x01, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0x1F, 0x7F, 0xFF, 0xFF, 0xFC, 0xF0, 0xC0, 0x00, 0x00, 0xC0, 0xE0, 0xF8, 0xFE, 0xFF, 0x7F, 0x1F,**

**0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x0F, 0xFF, 0xFF, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x0F, 0x03, 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, 0x00, 0x00, 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, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"Y",89\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x07,**

**0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x87, 0xC7, 0xF7, 0xFF, 0xFF, 0xFF, 0x7F, 0x1F, 0x0F, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xF8, 0xFC, 0xFF, 0x7F, 0x3F, 0x0F, 0x07, 0x03,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0xFC, 0xFE, 0xFF, 0x3F, 0x1F,**

**0x0F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xF8, 0xFC, 0xFF,**

**0xFF, 0xDF, 0xCF, 0xC7, 0xC1, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"Z",90\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF8, 0xF8, 0xF8, 0xF8, 0x78, 0x78,**

**0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 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, 0x00, 0xFF, 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, 0x00, 0xFF, 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, 0x00, 0xFF, 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, 0x00, 0xFF, 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, 0x00, 0x1F, 0x1F, 0x1F, 0x1F, 0x1E, 0x1E,**

**0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"[",91\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0xF0, 0xF0, 0xF0, 0xE0, 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, 0x00, 0x00, 0x07,**

**0x1F, 0x7F, 0xFF, 0xFE, 0xF8, 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, 0x00, 0x00, 0x01, 0x0F, 0x3F, 0xFF, 0xFF, 0xFC, 0xF0, 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, 0x00, 0x00, 0x03, 0x1F,**

**0x7F, 0xFF, 0xFE, 0xF8, 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, 0x01, 0x0F, 0x3F, 0xFF, 0xFF, 0xFC, 0xF0, 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, 0x00, 0x00, 0x03, 0x1F, 0x7F,**

**0xFF, 0xFE, 0xF8, 0xE0, 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, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"\",92\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78,**

**0x78, 0xF8, 0xF8, 0xF8, 0xF8, 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, 0xFF, 0xFF, 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, 0xFF, 0xFF, 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, 0xFF, 0xFF, 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, 0xFF, 0xFF, 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, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E,**

**0x1E, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"]",93\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xC0, 0xF0, 0xF8, 0xFE, 0x3F, 0x1F, 0x07, 0x0F, 0x3F, 0x7F, 0xFC, 0xF8, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xF8, 0xFE, 0x7F, 0x1F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x0F, 0x3F, 0xFF,**

**0xFE, 0xF8, 0xE0, 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, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"^",94\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E,**

**0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E}, /\*"\_",95\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10, 0x30, 0x70, 0x70, 0xF0, 0xF0, 0xF0, 0xE0, 0xC0, 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, 0x00, 0x01, 0x01, 0x03, 0x07, 0x07, 0x07, 0x06, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"`",96\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x0F, 0x1F, 0xFF, 0xFF,**

**0xFE, 0xFC, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF0, 0xF0, 0xF8, 0x78, 0x78, 0x3C, 0x3C, 0x3C, 0x3C,**

**0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0xFF, 0xFF, 0xFF,**

**0xFF, 0xE0, 0xC0, 0xC0, 0x80, 0x80, 0x80, 0xC0, 0xC0, 0xC0, 0xE0, 0xE0, 0xF0, 0x78, 0x7F, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x00, 0x00, 0x00, 0x03,**

**0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"a",97\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 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, 0xFF, 0xFF, 0xFF,**

**0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xF8, 0x7C, 0x3E, 0x1F, 0x0F, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x0F, 0x1F, 0xFF,**

**0xFF, 0xFE, 0xFC, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF,**

**0xFF, 0xE0, 0xC0, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xF0, 0xF8, 0xFF, 0x7F, 0x3F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"b",98\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0xFC, 0xFC, 0xFE, 0x3F, 0x1F, 0x0F, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07,**

**0x07, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0xFF, 0xFF, 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, 0x07, 0x1F, 0x3F,**

**0x7F, 0xFF, 0xFC, 0xF8, 0xF0, 0xE0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"c",99\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xF8, 0xFC, 0xFE, 0x3F, 0x1F, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0xFF, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFE, 0xFF, 0xFF, 0xFF, 0xFF, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x1F, 0x7F, 0xFF, 0xFF,**

**0xFE, 0xF0, 0xE0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xF0, 0xF8, 0x7C, 0x3E, 0x1F, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x03,**

**0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"d",100\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xF8, 0xFC, 0xFE, 0x3F, 0x1F, 0x0F, 0x07, 0x07, 0x03, 0x03, 0x03, 0x03, 0x07, 0x07, 0x0F, 0x0F, 0x3F, 0xFF,**

**0xFE, 0xFC, 0xF8, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFF, 0xFF, 0xFF, 0xFF, 0xBF, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C,**

**0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3F, 0x3F, 0x3F, 0x3F, 0x3F, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x7F, 0xFF,**

**0xFF, 0xF8, 0xF0, 0xE0, 0xE0, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"e",101\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xC0,**

**0xE0, 0xE0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xE0, 0xFE, 0xFF, 0xFF, 0xFF, 0x1F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78,**

**0x78, 0x78, 0x78, 0x78, 0x78, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 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, 0xFF, 0xFF, 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, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"f",102\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x80, 0xF0, 0xFC, 0xFE, 0xFF, 0x7F, 0x0F, 0x07, 0x07, 0x03, 0x03, 0x03, 0x03, 0x03, 0x07, 0x07, 0x0F, 0x3F, 0xFF, 0xFF,**

**0xFF, 0xF7, 0xC7, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x8F, 0xFF, 0xFF, 0xFF, 0xFE, 0xF8, 0xF0, 0xE0, 0xE0, 0xC0, 0xC0,**

**0xC0, 0xE0, 0xE0, 0xF0, 0xF8, 0xFE, 0x7F, 0x7F, 0x3F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3E, 0x7F, 0xFF, 0xFF,**

**0xFB, 0xF0, 0xE0, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xF8, 0xFC, 0xFE, 0xFF, 0xFF, 0x0F, 0x03, 0x03, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x87,**

**0xFF, 0xFF, 0xFF, 0xFF, 0xFE, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x0F, 0x0F, 0x0F, 0x1F, 0x1E, 0x1E, 0x1E, 0x1E, 0x1C, 0x1C,**

**0x1C, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x0F, 0x0F, 0x0F, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00}, /\*"g",103\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 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, 0xFF, 0xFF, 0xFF,**

**0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xF8, 0x7C, 0x3E, 0x1F, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x1F, 0xFF, 0xFF,**

**0xFF, 0xFC, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF,**

**0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"h",104\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xF0, 0xF0,**

**0xF0, 0xF0, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x81, 0x83, 0x87, 0x87, 0x87, 0x87, 0x83, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0xFF, 0xFF, 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, 0xFF, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0,**

**0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"i",105\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 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, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x81, 0x83, 0x87, 0x87, 0x87, 0x87, 0x83, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0xFF, 0xFF, 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, 0xFF, 0xFF, 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, 0xFF, 0xFF, 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, 0x80, 0xF0, 0xFF, 0xFF, 0xFF, 0xFF, 0x1F,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1F,**

**0x1F, 0x0F, 0x0F, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"j",106\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 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, 0xFF, 0xFF, 0xFF,**

**0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF0, 0xF8, 0xFC, 0x7E, 0x3F, 0x1F, 0x1F,**

**0x0F, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x08, 0x1C, 0x3E, 0x7F, 0xFF, 0xFF,**

**0xF7, 0xE3, 0xC1, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF,**

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

**0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x02, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"k",107\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0,**

**0xF0, 0xF0, 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, 0xFF, 0xFF, 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, 0xFF, 0xFF, 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, 0xFF, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0,**

**0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"l",108\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x00, 0x00,**

**0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xF8, 0xFC, 0x3E, 0x0F, 0x07, 0x03, 0x07, 0x0F, 0xFF, 0xFF, 0xFF, 0xF0, 0xFC, 0x3E, 0x1F, 0x07, 0x07, 0x07,**

**0x0F, 0xFF, 0xFF, 0xFF, 0xFE, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"m",109\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80,**

**0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xF8, 0x7C, 0x3E, 0x1F, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x0F, 0xFF, 0xFF,**

**0xFF, 0xFE, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF,**

**0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"n",110\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xE0, 0xF8, 0xFC, 0xFE, 0xFE, 0x3F, 0x1F, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x0F, 0x1F, 0x3F, 0xFF,**

**0xFE, 0xFC, 0xF8, 0xF0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x3F, 0x7F, 0xFF,**

**0xFC, 0xF0, 0xE0, 0xE0, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0xC0, 0xC0, 0xC0, 0xE0, 0xF0, 0xF8, 0xFE, 0xFF, 0x7F, 0x3F, 0x0F, 0x03, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"o",111\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80,**

**0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xF8, 0x7C, 0x3E, 0x1F, 0x0F, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x0F, 0x1F, 0xFF,**

**0xFF, 0xFE, 0xFC, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF,**

**0xFF, 0xE0, 0xC0, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xF0, 0xF8, 0xFF, 0x7F, 0x3F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x01, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1F, 0x1F, 0x1F, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"p",112\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xF8, 0xFC, 0xFE, 0x3F, 0x1F, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0xFF, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFE, 0xFF, 0xFF, 0xFF, 0xFF, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x1F, 0x7F, 0xFF, 0xFF,**

**0xFE, 0xF0, 0xE0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xF0, 0xF8, 0x7C, 0x3E, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**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, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"q",113\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80,**

**0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0xF8, 0x7C, 0x3E, 0x1F, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x0F, 0x1F,**

**0xFF, 0xFF, 0xFE, 0xFC, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**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, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"r",114\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF8, 0xFC, 0xFE, 0xFF, 0xFF, 0x0F, 0x07, 0x07, 0x07, 0x07, 0x03, 0x03, 0x03, 0x07, 0x07, 0x07, 0x07, 0x07,**

**0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0x0F, 0x0F, 0x1F, 0x1F, 0x3E, 0x3E, 0x3C, 0x7C,**

**0x7C, 0x78, 0xF8, 0xF8, 0xF8, 0xF0, 0xF0, 0xE0, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xC0, 0xC0,**

**0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xE1, 0xFF, 0xFF, 0xFF, 0x7F, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"s",115\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0xC0, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07,**

**0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 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, 0x1F, 0x7F, 0xFF, 0xFF, 0xFF, 0xF0, 0xE0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"t",116\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80,**

**0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0xFF, 0xFF,**

**0xFF, 0xFE, 0xE0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xE0, 0xF0, 0xF8, 0x7C, 0x3F, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x00, 0x00, 0x00, 0x03,**

**0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"u",117\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00,**

**0x00, 0x00, 0x01, 0x07, 0x1F, 0xFF, 0xFF, 0xFF, 0xFC, 0xF0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xFE,**

**0xFF, 0xFF, 0x7F, 0x1F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x1F, 0x7F, 0xFF, 0xFF, 0xFE, 0xF0, 0xC0, 0x00, 0x00,**

**0x00, 0x00, 0xC0, 0xF0, 0xFE, 0xFF, 0xFF, 0x3F, 0x0F, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x03, 0x0F, 0x3F, 0xFF, 0xFF, 0xFE, 0xF0, 0xF0, 0xFE, 0xFF, 0xFF, 0x3F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"v",118\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00,**

**0x00, 0x03, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xF8, 0xFF, 0xFF, 0xFF, 0xFF, 0x01, 0x00, 0x00, 0x00, 0x01, 0x7F, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0xE0, 0xFC, 0xFF, 0xFF, 0x1F,**

**0x3F, 0xFF, 0xFF, 0xF8, 0xE0, 0x00, 0x00, 0x00, 0xF0, 0xFF, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0xFF, 0xFF, 0xFF,**

**0xF8, 0xF8, 0xFF, 0xFF, 0x3F, 0x07, 0x00, 0x00, 0x00, 0x01, 0x0F, 0x7F, 0xFF, 0xFF, 0xF8, 0xF0, 0xFF, 0xFF, 0xFF, 0x1F, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"w",119\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00,**

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

**0x3F, 0x0F, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0xC7, 0xEF, 0xFF, 0xFF, 0xFF,**

**0xFE, 0xFF, 0xFF, 0xEF, 0xC7, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF8,**

**0xFC, 0xFE, 0x7F, 0x3F, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x03, 0x07, 0x0F, 0x3F, 0x7F, 0xFE, 0xFC, 0xF8, 0xF0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x02, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x02, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"x",120\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00,**

**0x00, 0x00, 0x01, 0x07, 0x1F, 0xFF, 0xFF, 0xFF, 0xFC, 0xF0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF0, 0xFE,**

**0xFF, 0xFF, 0x7F, 0x1F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x7F, 0xFF, 0xFF, 0xFE, 0xF8, 0xC0, 0x00, 0x00,**

**0x00, 0x00, 0x80, 0xF0, 0xFE, 0xFF, 0xFF, 0x7F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x01, 0x0F, 0x3F, 0xFF, 0xFF, 0xFF, 0xF8, 0xF0, 0xFE, 0xFF, 0xFF, 0x3F, 0x0F, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF8, 0xFF, 0xFF, 0xFF, 0x3F, 0x1F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1F, 0x0F, 0x0F, 0x07, 0x07, 0x03, 0x01, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"y",121\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x87, 0xC7, 0xF7, 0xFF, 0xFF, 0xFF, 0x7F,**

**0x1F, 0x0F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF0, 0xF8, 0xFE,**

**0x7F, 0x3F, 0x1F, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xF8,**

**0xFC, 0xFE, 0xFF, 0xDF, 0xCF, 0xC7, 0xC3, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"z",122\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0,**

**0xE0, 0xF0, 0xF0, 0xF8, 0x78, 0x78, 0x78, 0x78, 0x78, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF0, 0xFF, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0E, 0x0E, 0x0E, 0x1F, 0x1F, 0x1F, 0x3F, 0xFB, 0xFB, 0xF1, 0xF0,**

**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, 0x00, 0x01, 0xFF, 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, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xE0, 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, 0x00, 0x03, 0x07,**

**0x0F, 0x0F, 0x0F, 0x1F, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"{",123\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 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, 0x00, 0x00, 0xFF, 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, 0x00, 0xFF, 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, 0x00, 0xFF, 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, 0x00, 0xFF, 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, 0x00, 0xFF, 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, 0x00, 0xFF, 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, 0x00, 0x1F, 0x1F,**

**0x1F, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"|",124\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0xF8, 0xF0, 0xF0, 0xE0,**

**0xE0, 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, 0x00, 0x01, 0x07, 0xFF, 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, 0x00, 0x7F, 0xFF, 0xFF, 0xFF, 0xE0, 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, 0xC0,**

**0xF1, 0xF3, 0xFB, 0xFB, 0x3F, 0x1F, 0x1F, 0x1F, 0x0E, 0x0E, 0x0E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 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, 0x80, 0xE0, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1F, 0x0F, 0x0F, 0x07,**

**0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"}",125\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x78, 0x7E, 0x7F, 0x7F, 0x3F, 0x07, 0x03, 0x03, 0x03, 0x03, 0x07, 0x0F, 0x1F, 0x3F,**

**0x7E, 0x7C, 0xF8, 0xF8, 0xF0, 0xF0, 0xE0, 0xF0, 0xF0, 0xFF, 0xFF, 0x7F, 0x3F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"~",126\*/**

**};**

**// 18\*36 ASCII字符集点阵(Consolas)**

**const unsigned char asc2\_3618[][90] PROGMEM = {**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*" ",32\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x0F, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0E, 0x0F, 0x0F, 0x0E, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"!",33\*/**

**{0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F,**

**0x3F, 0x3F, 0x03, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*""",34\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x60, 0x60, 0x60,**

**0xFC, 0xFF, 0x7F, 0x60, 0x60, 0x60, 0xE0, 0xFF, 0xFF, 0x61, 0x60, 0x60, 0x00, 0x00, 0x30, 0x30, 0x30, 0xF8, 0xFF, 0x7F, 0x30, 0x30, 0x30,**

**0x30, 0xFF, 0xFF, 0x33, 0x30, 0x30, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x07, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"#",35\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF8, 0xFC, 0xFE,**

**0x8E, 0x06, 0x07, 0xE7, 0xFF, 0x7F, 0x07, 0x07, 0x06, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0xF7, 0xFF, 0x3F,**

**0x0E, 0x0C, 0x1C, 0xFC, 0xF8, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x06, 0x0E, 0x0E, 0x0C, 0xFC, 0xFF, 0x1F, 0x0C, 0x0E, 0x0E, 0x07, 0x07, 0x03,**

**0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"$",36\*/**

**{0x00, 0x00, 0x80, 0x80, 0xC0, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x00, 0x00, 0x3F, 0x7F, 0x61, 0x60,**

**0x60, 0x61, 0x7F, 0x3F, 0x80, 0xE0, 0xF0, 0x3C, 0x1E, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0x78, 0x1C, 0x0F, 0xE7,**

**0xF1, 0x38, 0x18, 0x18, 0x38, 0xF0, 0xF0, 0x80, 0x00, 0x0C, 0x0E, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0x0E, 0x0C, 0x0C, 0x0C,**

**0x0F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"%",37\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7E, 0xFF,**

**0xFF, 0xC3, 0x81, 0x81, 0xC3, 0xE3, 0xFF, 0x7E, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF8, 0xFC, 0x0E, 0x07, 0x07, 0x0F, 0x1F, 0x3D,**

**0x78, 0xF0, 0xE0, 0xF0, 0xFC, 0x7C, 0x00, 0x00, 0x00, 0x01, 0x07, 0x07, 0x0E, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x07, 0x07, 0x07, 0x0F,**

**0x0E, 0x0C, 0x08, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"&",38\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x07, 0x3F, 0x3F, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"'",39\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x80, 0xE0, 0xF8, 0x7C, 0x1F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x1F, 0x3E, 0xF8, 0xE0, 0xC0, 0x80, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00}, /\*"(",40\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x03, 0x07, 0x0F, 0x7E, 0xF8, 0xF0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF8, 0x7E, 0x1F, 0x07, 0x01, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*")",41\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x86, 0xCE,**

**0xCC, 0x58, 0x78, 0xFF, 0xFF, 0x78, 0x78, 0xCC, 0xCE, 0x86, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0x07, 0x07,**

**0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"\*",42\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0xFF, 0xFF,**

**0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"+",43\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0xC7, 0xFF, 0x7F, 0x3F, 0x0C, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*",",44\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18,**

**0x18, 0x18, 0x18, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"-",45\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x06, 0x0F, 0x0F, 0x0F, 0x0F, 0x06, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*".",46\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xFC, 0x1F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF8, 0x3F, 0x0F,**

**0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x40, 0x70, 0x7E, 0x1F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"/",47\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF8, 0x7C,**

**0x0E, 0x07, 0x03, 0x03, 0x03, 0x03, 0x87, 0xCE, 0xFE, 0xFC, 0xF0, 0x80, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xF0, 0x38, 0x1C, 0x1C, 0x0E, 0x07,**

**0x03, 0x03, 0x01, 0xC0, 0xFF, 0xFF, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0x0F, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x07, 0x07, 0x03,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"0",48\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x08, 0x38, 0x1C,**

**0x1C, 0x0E, 0x06, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0F, 0x0F, 0x0F, 0x0C, 0x0C, 0x0C, 0x0C,**

**0x0C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"1",49\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x0E,**

**0x06, 0x07, 0x03, 0x03, 0x03, 0x07, 0x0F, 0xFE, 0xFC, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0x70, 0x38,**

**0x1C, 0x0F, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x0E, 0x0F, 0x0F, 0x0F, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E,**

**0x0E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"2",50\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x06, 0x07,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x07, 0x8F, 0xFE, 0xFC, 0x78, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x06, 0x06, 0x06, 0x06, 0x07,**

**0x07, 0x0D, 0x1D, 0xFC, 0xF8, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x0E, 0x07, 0x03, 0x03,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"3",51\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xC0, 0xE0, 0xF8, 0x3C, 0x1F, 0x07, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF8, 0xFE, 0xEF, 0xE3, 0xE1, 0xE0, 0xE0, 0xE0,**

**0xE0, 0xFF, 0xFF, 0xFF, 0xE0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"4",52\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x07,**

**0x07, 0x06, 0x9E, 0xFE, 0xFC, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x06, 0x07, 0x03, 0x01,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"5",53\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF8,**

**0x3C, 0x1C, 0x0E, 0x06, 0x06, 0x07, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x06, 0x07, 0x03, 0x03, 0x03,**

**0x03, 0x03, 0x07, 0x9E, 0xFE, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0x07, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x07, 0x03,**

**0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"6",54\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07,**

**0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0xC7, 0xF7, 0xFF, 0x3F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0xFC,**

**0x3F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x08, 0x0E, 0x0F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"7",55\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0xFC, 0xFE,**

**0xCE, 0x87, 0x03, 0x03, 0x03, 0x03, 0x87, 0xC7, 0xFE, 0xFC, 0x38, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF0, 0xF9, 0x1D, 0x0F, 0x0F, 0x07, 0x07,**

**0x0F, 0x0F, 0x1D, 0xF9, 0xF8, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0x07, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x07, 0x07,**

**0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"8",56\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF8, 0xFC, 0xFE,**

**0x06, 0x07, 0x03, 0x03, 0x03, 0x03, 0x07, 0x0E, 0xFE, 0xF8, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0x0F, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C,**

**0x0C, 0x0C, 0xCE, 0xFF, 0xFF, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x06, 0x07, 0x07, 0x03, 0x01, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"9",57\*/**

**{0x00, 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, 0xE0, 0xE0, 0xE0, 0x40, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01,**

**0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x0F, 0x02, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*":",58\*/**

**{0x00, 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, 0xE0, 0xE0, 0xE0, 0x40, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x81, 0x81,**

**0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0xC7, 0xFF, 0xFF, 0x7F, 0x1E, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*";",59\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0x70, 0x70, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x18, 0x3C, 0x3E, 0x77, 0xE7, 0xC3, 0xC1,**

**0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x07, 0x0E, 0x0C, 0x04,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"<",60\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63,**

**0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"=",61\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30,**

**0x70, 0x70, 0xE0, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x81, 0xC3,**

**0xE3, 0x77, 0x3E, 0x3C, 0x18, 0x08, 0x00, 0x00, 0x00, 0x00, 0x00, 0x04, 0x0C, 0x0E, 0x07, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*">",62\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x01, 0x01, 0x01, 0x01, 0x03, 0x07, 0x8F, 0xFE, 0xFC, 0x78, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x3F, 0x3F, 0x03,**

**0x03, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0E, 0x0F, 0x0F, 0x0E, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"?",63\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xFC, 0x1E,**

**0x07, 0x03, 0x81, 0xC1, 0xC0, 0xC0, 0xC0, 0xC1, 0xC1, 0x07, 0x7F, 0xFC, 0xE0, 0xFC, 0xFF, 0x1F, 0x00, 0x00, 0xFE, 0xFF, 0x07, 0x01, 0x00,**

**0xF8, 0xFF, 0x3F, 0x00, 0x00, 0xE0, 0xFF, 0x3F, 0x07, 0x3F, 0xFF, 0xE0, 0x80, 0x03, 0x07, 0x06, 0x06, 0x03, 0x01, 0x07, 0x07, 0x86, 0x07,**

**0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00}, /\*"@",64\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xE0, 0xFC, 0x7F, 0x0F, 0x0F, 0x7F, 0xFE, 0xF0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF8, 0xFF, 0xFF, 0xE3, 0xE0, 0xE0, 0xE0,**

**0xE0, 0xE3, 0xFF, 0xFF, 0xFC, 0xE0, 0x00, 0x00, 0x08, 0x0F, 0x0F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07,**

**0x0F, 0x0F, 0x08, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"A",65\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0x03, 0x03, 0x03, 0x03, 0x07, 0x07, 0x07, 0xCE, 0xFE, 0xFC, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x06, 0x06, 0x06, 0x06, 0x06,**

**0x07, 0x0F, 0x0D, 0xFD, 0xFC, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x06, 0x07, 0x03,**

**0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"B",66\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF8, 0xFC,**

**0x3E, 0x0E, 0x07, 0x07, 0x07, 0x03, 0x03, 0x03, 0x07, 0x07, 0x06, 0x00, 0x00, 0x00, 0x04, 0xFF, 0xFF, 0xFF, 0x80, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x07, 0x0E, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E,**

**0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"C",67\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF,**

**0x03, 0x03, 0x03, 0x07, 0x07, 0x06, 0x0E, 0x0E, 0x3C, 0xF8, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x80, 0xE0, 0xFF, 0xFF, 0x3F, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x06, 0x07, 0x07, 0x03, 0x01,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"D",68\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x06, 0x06, 0x06, 0x06,**

**0x06, 0x06, 0x06, 0x06, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"E",69\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x06, 0x06, 0x06, 0x06,**

**0x06, 0x06, 0x06, 0x06, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"F",70\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF0, 0xF8, 0x7C,**

**0x1E, 0x0E, 0x07, 0x07, 0x07, 0x03, 0x03, 0x03, 0x07, 0x07, 0x06, 0x00, 0x00, 0x00, 0x3F, 0xFF, 0xFF, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x06,**

**0x06, 0x06, 0x06, 0xFE, 0xFE, 0xFE, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0x07, 0x0E, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0F, 0x0F,**

**0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"G",71\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x06, 0x06, 0x06, 0x06, 0x06,**

**0x06, 0x06, 0x06, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F,**

**0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"H",72\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03,**

**0x03, 0x03, 0x03, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0F, 0x0F, 0x0F, 0x0C, 0x0C, 0x0C, 0x0C,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"I",73\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07,**

**0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x06, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x0E, 0x07, 0x03, 0x01, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"J",74\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0x00, 0x00, 0x80, 0xC0, 0xF0, 0x78, 0x3C, 0x1E, 0x0F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x02, 0x0F, 0x1F, 0x3D, 0x78,**

**0xF0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x0F, 0x0E,**

**0x0C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"K",75\*/**

**{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, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C,**

**0x0C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"L",76\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xFF, 0xFF, 0x1F,**

**0x7E, 0xF0, 0xC0, 0x00, 0x00, 0xC0, 0xF0, 0x7E, 0x0F, 0xFF, 0xFF, 0xF8, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x03, 0x1F, 0x3E, 0x3E,**

**0x0F, 0x01, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07,**

**0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"M",77\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF,**

**0x3F, 0xFC, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x03, 0x0F, 0x3F,**

**0xFC, 0xF0, 0xC0, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x0F, 0x0F,**

**0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"N",78\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xFC, 0x3E,**

**0x0E, 0x07, 0x07, 0x03, 0x03, 0x03, 0x07, 0x0E, 0x1E, 0xFC, 0xF8, 0xE0, 0x00, 0x00, 0x7F, 0xFF, 0xFF, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x0F, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x07, 0x07, 0x03,**

**0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"O",79\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0x03, 0x03, 0x03, 0x03, 0x07, 0x07, 0x06, 0x0E, 0xFE, 0xFC, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x18, 0x18, 0x18, 0x18, 0x18,**

**0x1C, 0x0C, 0x0E, 0x07, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"P",80\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xFC, 0x3E,**

**0x0E, 0x07, 0x07, 0x03, 0x03, 0x07, 0x07, 0x0E, 0x1E, 0xFC, 0xF8, 0xE0, 0x00, 0x00, 0x7F, 0xFF, 0xFF, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x0F, 0x0E, 0x0C, 0x7C, 0xFC, 0xEC, 0xCE, 0x87, 0x87, 0x83,**

**0x81, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00}, /\*"Q",81\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0x03, 0x03, 0x03, 0x07, 0x07, 0x06, 0xFE, 0xFE, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x06, 0x06, 0x06, 0x0E,**

**0x3E, 0xFB, 0xF3, 0xE1, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x0F,**

**0x0E, 0x08, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"R",82\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x78, 0xFC, 0xFE,**

**0xCE, 0x87, 0x07, 0x03, 0x03, 0x03, 0x03, 0x03, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x07, 0x07, 0x0E,**

**0x0E, 0x0C, 0x1C, 0xFC, 0xF8, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x0E, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x0E, 0x07, 0x03,**

**0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"S",83\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x07,**

**0x07, 0x07, 0x07, 0xFF, 0xFF, 0xFF, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"T",84\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xC0, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x07, 0x0E, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x07, 0x03,**

**0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"U",85\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x0F, 0x7F, 0xFE, 0xF0,**

**0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xFC, 0x7F, 0x1F, 0x03, 0x00, 0x00, 0x00, 0x03, 0x1F, 0xFF, 0xFC, 0xE0, 0x00, 0x00,**

**0xE0, 0xFC, 0xFF, 0x1F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x0F, 0x0F, 0x0F, 0x03, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"V",86\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFE, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xFF, 0xFF, 0x00, 0x00, 0x07, 0xFF, 0xFF, 0x00, 0xC0, 0xF8, 0x7E, 0x0F, 0x0F,**

**0x7F, 0xF8, 0xC0, 0x00, 0xFF, 0xFF, 0x1F, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0E, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x0F, 0x0F,**

**0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"W",87\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x0F, 0x3E,**

**0x7C, 0xF0, 0xE0, 0x80, 0x80, 0xE0, 0xF0, 0x7C, 0x3E, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF9, 0x3F, 0x1F, 0x1F,**

**0x3F, 0xF9, 0xF0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x0E, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x0F,**

**0x0E, 0x0C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"X",88\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x0F, 0x3F, 0xFC,**

**0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xF8, 0x3E, 0x1F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x1F, 0xFE, 0xFC,**

**0xFF, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"Y",89\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07,**

**0x07, 0x07, 0x07, 0x07, 0x87, 0xC7, 0xF7, 0x7F, 0x3F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xF8, 0x3C, 0x1F, 0x0F,**

**0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0E, 0x0F, 0x0F, 0x0F, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E,**

**0x0E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"Z",90\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00}, /\*"[",91\*/**

**{0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07,**

**0x1F, 0x7E, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x3F,**

**0xFC, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x7E, 0x78,**

**0x60, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"\",92\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"]",93\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0,**

**0xF0, 0x3C, 0x1E, 0x07, 0x07, 0x1E, 0x7C, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x02, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x03, 0x03, 0x02, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"^",94\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 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, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03}, /\*"\_",95\*/**

**{0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x01, 0x01, 0x03, 0x03, 0x02, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"`",96\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0,**

**0x60, 0x60, 0x60, 0x60, 0x60, 0x60, 0x60, 0xE0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xF0, 0x30, 0x38, 0x18, 0x18, 0x18,**

**0x18, 0x18, 0x19, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0x0F, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x06, 0x03, 0x0F, 0x0F,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"a",97\*/**

**{0x00, 0x00, 0x00, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0x80, 0xC0, 0xE0, 0x60, 0x60, 0x60, 0x60, 0xE0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x03, 0x01, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xE7, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x0F, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x07, 0x07, 0x03,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"b",98\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80,**

**0xC0, 0xC0, 0xE0, 0x60, 0x60, 0x60, 0x60, 0x60, 0x60, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFE, 0xFF, 0xEF, 0x01, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x07, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"c",99\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0,**

**0xC0, 0xE0, 0x60, 0x60, 0x60, 0x60, 0x60, 0x60, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFE, 0xFF, 0xFF, 0x01, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x80, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x07, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x06, 0x07, 0x03, 0x0F, 0x0F,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"d",100\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0,**

**0xE0, 0xE0, 0x60, 0x60, 0x60, 0x60, 0x60, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFE, 0xFF, 0xFF, 0x19, 0x18, 0x18, 0x18, 0x18,**

**0x18, 0x18, 0x18, 0x1F, 0x1F, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0x07, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"e",101\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0xF0, 0xFE, 0xFF, 0x87, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x01, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0xFF, 0xFF, 0xFF, 0x01,**

**0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"f",102\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0,**

**0xE0, 0x60, 0x60, 0x60, 0x60, 0x60, 0xE0, 0xE0, 0xE0, 0xE0, 0x60, 0x60, 0x00, 0x00, 0x00, 0x80, 0xFF, 0xFF, 0x39, 0x70, 0x60, 0x60, 0x60,**

**0x60, 0x70, 0x3F, 0x1F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF3, 0xFF, 0xFF, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0C, 0x9C, 0xFC,**

**0xF8, 0x70, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00}, /\*"g",103\*/**

**{0x00, 0x00, 0x00, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0x80, 0xC0, 0xE0, 0x60, 0x60, 0x60, 0x60, 0xE0, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x03, 0x01, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFE, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F,**

**0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"h",104\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x60, 0x60,**

**0x60, 0x60, 0x63, 0xE7, 0xE7, 0xE3, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0F, 0x0F, 0x0F, 0x0C, 0x0C, 0x0C, 0x0C,**

**0x0C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"i",105\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x60, 0x60,**

**0x60, 0x60, 0x60, 0x60, 0x60, 0x63, 0xE7, 0xE3, 0xE3, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xFF, 0xFF, 0x1F, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"j",106\*/**

**{0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xE0, 0x60, 0x20, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x18, 0x3C, 0x7E, 0xF7,**

**0xE7, 0xC3, 0x81, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x0F, 0x0E,**

**0x0C, 0x08, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"k",107\*/**

**{0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01,**

**0x01, 0x01, 0x01, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0F, 0x0F, 0x0F, 0x0C, 0x0C, 0x0C, 0x0C,**

**0x0C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"l",108\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0, 0xE0, 0xC0,**

**0xE0, 0x60, 0x60, 0xE0, 0xC0, 0x80, 0xE0, 0x60, 0x60, 0xE0, 0xE0, 0x80, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x01, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0x03, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x0F,**

**0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"m",109\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0,**

**0x80, 0xC0, 0xE0, 0x60, 0x60, 0x60, 0x60, 0xE0, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x03, 0x01, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFE, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F,**

**0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"n",110\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0,**

**0xE0, 0x60, 0x60, 0x60, 0x60, 0x60, 0x60, 0xE0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x10, 0xFF, 0xFF, 0xC7, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x83, 0xFF, 0xFF, 0x3C, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x0F, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x0E, 0x07, 0x03,**

**0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"o",111\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0,**

**0x80, 0xC0, 0xE0, 0x60, 0x60, 0x60, 0x60, 0xE0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x03, 0x01, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xE7, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x07, 0x07, 0x03,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"p",112\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0,**

**0xC0, 0xE0, 0x60, 0x60, 0x60, 0x60, 0x60, 0x60, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFE, 0xFF, 0xFF, 0x01, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x80, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x07, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x06, 0x07, 0x03, 0xFF, 0xFF,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x00, 0x00, 0x00}, /\*"q",113\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0,**

**0xE0, 0x80, 0xC0, 0xE0, 0x60, 0x60, 0x60, 0x60, 0xE0, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x03, 0x01, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x01, 0x07, 0x07, 0x06, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"r",114\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0,**

**0xE0, 0xE0, 0x60, 0x60, 0x60, 0x60, 0x60, 0x60, 0x60, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x0F, 0x1C, 0x1C, 0x18, 0x38,**

**0x38, 0x70, 0xF0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0F, 0x07, 0x03,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"s",115\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x60, 0x60, 0x60, 0x60,**

**0x60, 0xFF, 0xFF, 0x60, 0x60, 0x60, 0x60, 0x60, 0x60, 0x60, 0x60, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0x0F, 0x0E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C,**

**0x0C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"t",116\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x80, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0x0F, 0x0C, 0x0C, 0x0C, 0x0C, 0x0E, 0x07, 0x03, 0x0F, 0x0F,**

**0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"u",117\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x20, 0xE0, 0xE0, 0xE0,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xE0, 0x20, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x7F, 0xF8, 0xE0, 0x00, 0x00,**

**0xE0, 0xF8, 0xFF, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x0F, 0x0F, 0x0F, 0x03, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"v",118\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0, 0xC0, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xE0, 0x00, 0x00, 0x07, 0xFF, 0xFF, 0xC0, 0x00, 0xE0, 0xFC, 0x3E, 0x3E,**

**0xFC, 0xE0, 0x80, 0x80, 0xFF, 0xFF, 0x0F, 0x00, 0x00, 0x00, 0x01, 0x0F, 0x0F, 0x0F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x07, 0x0F, 0x0F, 0x0F,**

**0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"w",119\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x20, 0x60, 0xE0,**

**0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xE0, 0x60, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x83, 0xE7, 0xFF, 0x7C, 0x7C,**

**0xFE, 0xEF, 0xC3, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x0E, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x0F, 0x0E,**

**0x0C, 0x08, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"x",120\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x20, 0xE0, 0xE0, 0xE0,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xE0, 0x20, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x7F, 0xFC, 0xE0, 0x80, 0x00,**

**0xC0, 0xF8, 0xFF, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC3, 0xFF, 0xFF, 0x3F, 0x1F, 0x03, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"y",121\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x60, 0x60,**

**0x60, 0x60, 0x60, 0x60, 0x60, 0x60, 0xE0, 0xE0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0x78, 0x3C,**

**0x1E, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0E, 0x0F, 0x0F, 0x0D, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C,**

**0x0C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"z",122\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFE, 0xFF, 0xFF, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x0C, 0x1C, 0x1E, 0xF7, 0xF3, 0xE1,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7F, 0xFF, 0xFF, 0x80, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00}, /\*"{",123\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF8, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"|",124\*/**

**{0x00, 0x00, 0x00, 0x00, 0xC0, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x01, 0x01, 0xFF, 0xFF, 0xFE, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC1, 0xF3,**

**0xF7, 0x1E, 0x1C, 0x0C, 0x0C, 0x0C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"}",125\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x38, 0x3C, 0x3E, 0x06, 0x06, 0x06, 0x0E, 0x1C, 0x3C,**

**0x38, 0x70, 0x70, 0x70, 0x3C, 0x3E, 0x0E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"~",126\*/**

**};**

**// 24\*48 ASCII字符集点阵(Consolas)**

**const unsigned char asc2\_4824[][144] PROGMEM = {**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*" ",32\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x18, 0x3C, 0x3E, 0x3E, 0x3E, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"!",33\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x0C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*""",34\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x1C, 0x1C, 0x1C, 0x1C, 0xFC, 0xFF, 0xFF, 0x7F, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFC, 0xFF, 0xFF, 0x1F, 0x1C, 0x1C, 0x1C, 0x1C, 0x00, 0x00, 0x70, 0x70,**

**0x70, 0x70, 0xF0, 0xFF, 0xFF, 0xFF, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xFF, 0xFF, 0x7F, 0x70, 0x70, 0x70, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x20, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"#",35\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xE0, 0xF0, 0x70, 0x70, 0x70, 0xFC, 0xFF, 0xFF, 0x73, 0x70, 0x70, 0x70, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x3F, 0x7F, 0xFF, 0xFF, 0xE0, 0xE0, 0xC0, 0xC0, 0xFE, 0xFF, 0xFF, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x83, 0xFF, 0xFF, 0x7F, 0x07, 0x0F, 0x0F, 0x1E, 0x3E, 0xFC, 0xFC, 0xF8, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x1C, 0x1C,**

**0x3C, 0x38, 0x38, 0x38, 0xF8, 0xFF, 0xFF, 0x3F, 0x38, 0x38, 0x3C, 0x1C, 0x1E, 0x1F, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"$",36\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xF0, 0xF8, 0xFC, 0x1C, 0x0C, 0x0E, 0x0E, 0x1C, 0xFC, 0xF8, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF8, 0x3C, 0x1C, 0x0C, 0x00, 0x00, 0x07,**

**0x0F, 0x1F, 0x1C, 0x18, 0x18, 0x18, 0x1C, 0x1F, 0x0F, 0xC7, 0xE0, 0xF8, 0x7C, 0x3F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x80, 0xE0, 0xF0, 0xFC, 0x3E, 0x1F, 0x07, 0xE3, 0xF0, 0xF8, 0x38, 0x1C, 0x1C, 0x1C, 0x38, 0xF8, 0xF8, 0xE0, 0x00, 0x00, 0x30, 0x38, 0x3C,**

**0x1F, 0x0F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x3F, 0x38, 0x38, 0x30, 0x30, 0x38, 0x3F, 0x1F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"%",37\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xC0, 0xF0, 0xF8, 0xF8, 0x38, 0x1C, 0x1C, 0x1C, 0x1C, 0x38, 0xF8, 0xF8, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x0F, 0x3F, 0xFF, 0xFF, 0xF0, 0xE0, 0xE0, 0xE0, 0x70, 0x78, 0x3F, 0x1F, 0x0F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF8,**

**0xFE, 0xFF, 0x1F, 0x07, 0x03, 0x01, 0x07, 0x0F, 0x1F, 0x3F, 0x7C, 0xF8, 0xF0, 0xE0, 0xC0, 0xFF, 0xFF, 0xFF, 0x1F, 0x00, 0x00, 0x00, 0x01, 0x07, 0x0F,**

**0x1F, 0x1E, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x1F, 0x0F, 0x0F, 0x1F, 0x3F, 0x3C, 0x38, 0x30, 0x20, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"&",38\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"'",39\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0xF8, 0x7C, 0x3E, 0x0F, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xFC, 0xFF, 0xFF, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x1C, 0xFF, 0xFF, 0xFF, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x03, 0x1F, 0x3F, 0xFF, 0xFC, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x0F, 0x1F, 0x3E, 0x3C, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"(",40\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x06, 0x0E, 0x1E, 0x7C, 0xF8, 0xF0, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x7F, 0xFF, 0xFC, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xFF, 0xFF, 0xFF, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF8, 0xFF, 0x7F, 0x1F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x18, 0x3C, 0x3E, 0x1F, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*")",41\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xC0, 0xE0, 0xC0, 0xC0, 0x80, 0x00, 0x00, 0xFC, 0xFC, 0x0C, 0x00, 0x80, 0xC0, 0xC0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x60, 0x70, 0x71, 0x39, 0x19, 0x1F, 0x0F, 0xFF, 0xFF, 0x0F, 0x1F, 0x1B, 0x39, 0x71, 0x70, 0x60, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"\*",42\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFE, 0xFE, 0xFE, 0xFE, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1C,**

**0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFF, 0xFF, 0xFF, 0xFF, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1F, 0x1F, 0x1F, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"+",43\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x1F, 0xFF, 0xFF, 0xFF, 0xFE, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x1C, 0x1C, 0x1C, 0x0C, 0x0E, 0x0F, 0x07, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*",",44\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"-",45\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x0C, 0x1E, 0x3F, 0x3F, 0x3F, 0x3F, 0x1E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*".",46\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 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, 0xF8, 0xFC, 0x7C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xFC, 0xFF, 0x1F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF8, 0xFE, 0x3F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xC0, 0xF0, 0xFE, 0x7F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03,**

**0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"/",47\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xE0, 0xF0, 0x70, 0x70, 0x38, 0x38, 0x70, 0x70, 0xF0, 0xF0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xF0, 0xFE, 0xFF, 0xFF, 0x07, 0x01, 0x00, 0x00, 0x00, 0x80, 0x80, 0xC0, 0xE0, 0xE0, 0xF1, 0x7F, 0xFF, 0xFF, 0xFF, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x7F,**

**0xFF, 0xFF, 0xFF, 0xF8, 0x3C, 0x1E, 0x1E, 0x0F, 0x07, 0x03, 0x03, 0x01, 0x00, 0x00, 0x80, 0xFC, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01,**

**0x07, 0x0F, 0x1F, 0x1E, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1E, 0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"0",48\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0x80, 0xC0, 0xC0, 0xE0, 0xE0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x01, 0x07, 0x03, 0x03, 0x01, 0x01, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3F, 0x3F, 0x3F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"1",49\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xC0, 0xE0, 0xE0, 0xF0, 0x70, 0x70, 0x70, 0x78, 0x78, 0x70, 0xF0, 0xF0, 0xE0, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC1, 0xFF, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF0, 0xF8, 0x7C, 0x3E, 0x1F, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x38,**

**0x3C, 0x3E, 0x3F, 0x3F, 0x3B, 0x39, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"2",50\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x78, 0x78, 0x70, 0x70, 0x70, 0xF0, 0xE0, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xC0, 0xF1, 0x7F, 0x7F, 0x3F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x07, 0x07, 0x0F, 0xFE, 0xFE, 0xFC, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1C, 0x1E, 0x0F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"3",51\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0xFC, 0x3E, 0x1F, 0x07, 0x03, 0x01, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF8,**

**0xFE, 0xFF, 0xEF, 0xE7, 0xE1, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xFF, 0xFF, 0xFF, 0xFF, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"4",52\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x03, 0x03, 0x07, 0x0F, 0xFF, 0xFF, 0xFE, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1E, 0x1F, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"5",53\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xC0, 0xE0, 0xE0, 0xE0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xF0, 0xFC, 0xFF, 0xFF, 0x87, 0x83, 0xC1, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x03, 0x07, 0xFF, 0xFF, 0xFE, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01,**

**0x07, 0x0F, 0x1F, 0x1E, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1E, 0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"6",54\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF8, 0xFE, 0x7F, 0x1F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xF8, 0xFE, 0x7F, 0x1F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x30, 0x3C, 0x3F, 0x3F, 0x1F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"7",55\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xC0, 0xE0, 0xE0, 0xF0, 0x70, 0x70, 0x30, 0x38, 0x38, 0x30, 0x70, 0x70, 0xF0, 0xF0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x0F, 0x3F, 0x7F, 0xFF, 0xF0, 0xE0, 0xE0, 0xC0, 0xC0, 0x80, 0xC0, 0xC0, 0xE0, 0xF0, 0x7F, 0x7F, 0x3F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xF0, 0xF8, 0xFC, 0xFC, 0x1E, 0x0F, 0x07, 0x07, 0x03, 0x03, 0x07, 0x07, 0x0F, 0x1E, 0x7E, 0xFC, 0xF8, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07,**

**0x0F, 0x1F, 0x1F, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1E, 0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"8",56\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xC0, 0xE0, 0xE0, 0xF0, 0x70, 0x70, 0x30, 0x38, 0x38, 0x70, 0x70, 0xF0, 0xF0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x7C, 0xFF, 0xFF, 0xFF, 0xC3, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x9F, 0xFF, 0xFF, 0xFE, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x01, 0x03, 0x07, 0x07, 0x0F, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x07, 0x87, 0xF3, 0xFF, 0xFF, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x1C, 0x1C, 0x1E, 0x0F, 0x07, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"9",57\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x38, 0x7C, 0xFC, 0xFC, 0x7C, 0x78, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x1E, 0x3E, 0x3F, 0x3F, 0x3E, 0x1E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*":",58\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x38, 0x7C, 0xFC, 0xFC, 0x7C, 0x78, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x0E, 0x1F, 0xFF, 0xFF, 0xFF, 0xFE, 0x3C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x1C, 0x1C, 0x1C, 0x1C, 0x0E, 0x0F, 0x07, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*";",59\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF0, 0xF0, 0x78, 0x3C, 0x1E, 0x1E, 0x0F, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x08, 0x0C, 0x1E, 0x3F, 0x7F, 0x7F, 0xF3, 0xE1, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x07, 0x0F, 0x1E, 0x3C, 0x3C, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"<",60\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"=",61\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x06, 0x0F, 0x0F, 0x1E, 0x3C, 0x78, 0x70, 0xF0, 0xE0, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE1, 0xF3, 0xF7, 0x7F, 0x3F, 0x1E, 0x0C, 0x08, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x18, 0x38, 0x3C, 0x1E, 0x0F, 0x07, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*">",62\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1C, 0x1C, 0x1C, 0x1C, 0x3C, 0x3C, 0x78, 0x78, 0xF0, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xC0, 0xE1, 0xFF, 0xFF, 0x7F, 0x1E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x3C, 0x3E, 0x3E, 0x3E, 0x3C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"?",63\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF0, 0x38, 0x38, 0x1C, 0x0C, 0x0C, 0x0E, 0x0E, 0x0C, 0x0C, 0x1C, 0x3C, 0xF8, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0xC0,**

**0xF8, 0xFE, 0x3F, 0x07, 0x01, 0x00, 0xC0, 0xE0, 0xF0, 0x70, 0x30, 0x30, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x07, 0xFF, 0xFF, 0xF8, 0xFC, 0xFF, 0xFF,**

**0x07, 0x00, 0x00, 0xF8, 0xFF, 0xFF, 0x1F, 0x00, 0x00, 0x00, 0xE0, 0xFF, 0xFF, 0xFF, 0x01, 0x00, 0x00, 0xE0, 0xFF, 0xFF, 0x1F, 0x07, 0xFF, 0xFF, 0xF8,**

**0x80, 0x00, 0x03, 0x0F, 0x0F, 0x1F, 0x1C, 0x0C, 0x0E, 0x07, 0x07, 0x0F, 0x1F, 0x1C, 0x1C, 0x0E, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x03, 0x07, 0x0F,**

**0x1E, 0x1C, 0x38, 0x38, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x38, 0x18, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"@",64\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xF0, 0xFC, 0xFF, 0x7F, 0x0F, 0x01, 0x01, 0x07, 0x3F, 0xFF, 0xFE, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xE0, 0xFC, 0xFF, 0xFF, 0xEF, 0xE1, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE1, 0xEF, 0xFF, 0xFF, 0xFE, 0xF0, 0x80, 0x00, 0x00, 0x20, 0x3C, 0x3F, 0x3F,**

**0x1F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x1F, 0x3F, 0x3F, 0x3C, 0x20, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"A",65\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xF0, 0xE0, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xC0, 0xF1, 0xFF, 0x7F, 0x3F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x07, 0x0F, 0x1F, 0xFE, 0xFE, 0xFC, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F,**

**0x3F, 0x3F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x1C, 0x1F, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"B",66\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xE0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xE0, 0xFC, 0xFF, 0xFF, 0x1F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F,**

**0xFF, 0xFF, 0xFF, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01,**

**0x03, 0x07, 0x0F, 0x1F, 0x1E, 0x3C, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"C",67\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xE0, 0xE0, 0xC0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x7F, 0xFF, 0xFF, 0xFC, 0xC0, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF8, 0xFF, 0xFF, 0x7F, 0x07, 0x00, 0x00, 0x00, 0x3F, 0x3F,**

**0x3F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x1C, 0x1E, 0x0F, 0x0F, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"D",68\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x3F, 0x3F, 0x3F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"E",69\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"F",70\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xE0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x78, 0x78, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x80,**

**0xF8, 0xFE, 0xFF, 0x7F, 0x0F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x07, 0x7F,**

**0xFF, 0xFF, 0xF8, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01,**

**0x07, 0x0F, 0x0F, 0x1F, 0x1E, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3F, 0x3F, 0x1F, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"G",71\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F,**

**0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"H",72\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3F, 0x3F, 0x3F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"I",73\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x1C, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1E, 0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"J",74\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF0, 0xF0, 0x70, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0xC0, 0xE0, 0xF0, 0xF8, 0x3C, 0x1E, 0x0F, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x07, 0x0F, 0x1F, 0x3E, 0xFC, 0xF8, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F,**

**0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x1F, 0x3F, 0x3E, 0x3C, 0x30, 0x20, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"K",75\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"L",76\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0x00, 0x07, 0x3F, 0xFF, 0xFC, 0xE0, 0x00, 0x00, 0xE0, 0xFC, 0xFF, 0x1F, 0x03, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xE0, 0xFF,**

**0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x1F, 0x1F, 0x1F, 0x03, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFC, 0x00, 0x00, 0x3F, 0x3F, 0x3F,**

**0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"M",77\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0x01, 0x07, 0x3F, 0xFF, 0xFC, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x3F, 0xFF, 0xFC, 0xF0, 0xC0, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F,**

**0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x3F, 0x3F, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"N",78\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xE0, 0xF0, 0x70, 0x70, 0x78, 0x78, 0x70, 0x70, 0x70, 0xF0, 0xE0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0xE0,**

**0xFC, 0xFF, 0xFF, 0x1F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0xFF, 0xFF, 0xFE, 0xF0, 0x00, 0x00, 0x1F, 0xFF,**

**0xFF, 0xFF, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xFF, 0xFF, 0xFF, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x03,**

**0x0F, 0x0F, 0x1F, 0x3E, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1E, 0x1F, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"O",79\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xE0, 0xE0, 0xC0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC3, 0xFF, 0xFF, 0xFF, 0x3E, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1E, 0x0E, 0x0E, 0x0E, 0x0F, 0x07, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F,**

**0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"P",80\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xE0, 0xF0, 0x70, 0x70, 0x78, 0x78, 0x70, 0x70, 0xF0, 0xF0, 0xE0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0xC0,**

**0xFC, 0xFF, 0xFF, 0x1F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0xFF, 0xFF, 0xFE, 0xF0, 0x00, 0x00, 0x1F, 0xFF,**

**0xFF, 0xFF, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xFF, 0xFF, 0xFF, 0x1F, 0x00, 0x00, 0x00, 0x01, 0x03,**

**0x0F, 0x0F, 0x1F, 0x3E, 0x3C, 0x38, 0xF8, 0xF8, 0xF8, 0xF8, 0x38, 0x3C, 0x1E, 0x1F, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x0F, 0x1F, 0x1E, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1E, 0x0E, 0x0C}, /\*"Q",81\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xF0, 0xE0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xFF, 0xFF, 0x7F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x07, 0x07, 0x1F, 0x3F, 0xFD, 0xF9, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x3F, 0x3F, 0x3C, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"R",82\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xC0, 0xE0, 0xE0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x78, 0x78, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x3F, 0x7F, 0xFF, 0xFF, 0xF0, 0xE0, 0xC0, 0xC0, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x07, 0x07, 0x0F, 0x0F, 0x1E, 0xFE, 0xFC, 0xF8, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1C, 0x3C,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1C, 0x1F, 0x0F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"S",83\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"T",84\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07,**

**0x0F, 0x1F, 0x1F, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1F, 0x0F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"U",85\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10,**

**0xF0, 0xF0, 0xF0, 0xF0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF0, 0xF0, 0xF0, 0x10, 0x00, 0x00,**

**0x07, 0x1F, 0xFF, 0xFF, 0xFC, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF8, 0xFF, 0xFF, 0x3F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x07, 0x3F, 0xFF, 0xFF, 0xF8, 0xC0, 0x00, 0x00, 0x80, 0xF0, 0xFE, 0xFF, 0x3F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x01, 0x07, 0x3F, 0x3F, 0x3E, 0x3C, 0x3F, 0x3F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"V",86\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x07,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFE, 0xFF, 0xFF, 0x3F, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0xF8, 0x00, 0xC0, 0xF8, 0xFF, 0x3F, 0x07, 0x07, 0x3F, 0xFF, 0xFC, 0xE0, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x01, 0x3F,**

**0x3F, 0x3F, 0x3E, 0x3F, 0x0F, 0x01, 0x00, 0x00, 0x00, 0x00, 0x01, 0x0F, 0x3F, 0x3F, 0x3C, 0x3F, 0x3F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"W",87\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x30, 0x70, 0xF0, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0xF0, 0x70, 0x30, 0x10, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x01, 0x03, 0x0F, 0x1F, 0x7F, 0xFC, 0xF8, 0xE0, 0xE0, 0xF0, 0xFC, 0x7F, 0x1F, 0x0F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x80, 0xE0, 0xF0, 0xFC, 0x7E, 0x3F, 0x0F, 0x0F, 0x3F, 0x7E, 0xFC, 0xF0, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0x38, 0x3C,**

**0x3F, 0x1F, 0x0F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x0F, 0x1F, 0x3F, 0x3E, 0x38, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"X",88\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10,**

**0x30, 0xF0, 0xF0, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xF0, 0xF0, 0x70, 0x10, 0x00, 0x00,**

**0x00, 0x03, 0x07, 0x1F, 0x7F, 0xFE, 0xFC, 0xF0, 0xC0, 0x00, 0x00, 0x80, 0xE0, 0xF8, 0xFC, 0x7F, 0x1F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0xFF, 0xFF, 0xFE, 0xFF, 0x0F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"Y",89\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0xFC, 0x7E, 0x3F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF8, 0x7E, 0x3F, 0x1F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x38, 0x3C,**

**0x3F, 0x3F, 0x3F, 0x3B, 0x39, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"Z",90\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFE, 0xFE, 0xFE, 0xFE, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"[",91\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x0C, 0x3C, 0xFC, 0xF8, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x7F, 0xFE, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x0F, 0x3F, 0xFF, 0xFC, 0xF0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x7F, 0xFE, 0xF8, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00}, /\*"\",92\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0xFE, 0xFE, 0xFE, 0xFE, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"]",93\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0x70, 0xF0, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x80, 0xC0, 0xF0, 0xFC, 0x3E, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x01, 0x07, 0x1F, 0x3F, 0xFC, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"^",94\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x38, 0x38, 0x38, 0x38, 0x38,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38}, /\*"\_",95\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x04, 0x0C, 0x0C, 0x1C, 0x3C, 0x38, 0x70, 0x60, 0x40, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"`",96\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x38, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x7C, 0xF8, 0xF8, 0xF0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xC0, 0xE0, 0xF0, 0xF8, 0x38, 0x38, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07,**

**0x1F, 0x1F, 0x3F, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x1C, 0x0E, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"a",97\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xE0, 0x70, 0x38, 0x3C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x3C, 0xFC, 0xF8, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xFF, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1F,**

**0x1F, 0x1F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1E, 0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"b",98\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF0, 0xF8, 0x38, 0x3C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x3C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x3C, 0xFF, 0xFF, 0xFF, 0xC3, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x01, 0x07, 0x0F, 0x1F, 0x1E, 0x3C, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"c",99\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x80, 0xE0, 0xF0, 0xF8, 0x78, 0x3C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7C,**

**0xFF, 0xFF, 0xFF, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03,**

**0x0F, 0x1F, 0x3F, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x1E, 0x0F, 0x07, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"d",100\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x80, 0xE0, 0xF0, 0xF8, 0x78, 0x3C, 0x1C, 0x1C, 0x0C, 0x0C, 0x0C, 0x1C, 0x1C, 0x3C, 0xF8, 0xF8, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xDF, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1F, 0x1F, 0x1F, 0x1E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01,**

**0x07, 0x0F, 0x1F, 0x1E, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"e",101\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 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, 0xF0, 0xF8, 0xFC, 0x3C, 0x1C, 0x1C, 0x0C, 0x0E, 0x0E, 0x0C, 0x1C, 0x1C, 0x1C, 0x00, 0x00, 0x00,**

**0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xFF, 0xFF, 0xFF, 0xE1, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"f",102\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x80, 0xE0, 0xF0, 0xF8, 0x7C, 0x1C, 0x1C, 0x0C, 0x0C, 0x0C, 0x0C, 0x1C, 0x3C, 0xFC, 0xFC, 0xFC, 0xFC, 0x1C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x83, 0xCF, 0xFF, 0xFF, 0x7C, 0x70, 0x70, 0x60, 0x60, 0x60, 0x60, 0x70, 0x78, 0x3F, 0x3F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC7,**

**0xEF, 0xFF, 0x7E, 0x3C, 0x1C, 0x1C, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0xF8, 0xF0, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x07, 0x0F, 0x1F,**

**0x3F, 0x3C, 0x38, 0x38, 0x38, 0x30, 0x30, 0x30, 0x30, 0x38, 0x38, 0x38, 0x3C, 0x1F, 0x1F, 0x0F, 0x07, 0x00, 0x00}, /\*"g",103\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xE0, 0x70, 0x38, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x3C, 0xFC, 0xF8, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F,**

**0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"h",104\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x38, 0x7C, 0x7C, 0x7E, 0x7C, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3F, 0x3F, 0x3F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"i",105\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3C, 0x7C, 0x7E, 0x7C, 0x3C, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3C, 0x38,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1E, 0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"j",106\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xC0, 0xE0, 0xF0, 0x78, 0x3C, 0x1C, 0x0C, 0x0C, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x0C, 0x1E, 0x3F, 0x7F, 0xF7, 0xE3, 0xC1, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x0F, 0x1F, 0x3E, 0x3C, 0x38, 0x30, 0x20, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"k",107\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3F, 0x3F, 0x3F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"l",108\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFC, 0xFC, 0xFC, 0xE0, 0x78, 0x1C, 0x1C, 0x1C, 0xFC, 0xFC, 0xF8, 0xE0, 0xF0, 0x38, 0x1C, 0x0C, 0x1C, 0xFC, 0xFC, 0xF8, 0x00, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0x01, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F,**

**0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"m",109\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0xE0, 0x70, 0x38, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x3C, 0xFC, 0xF8, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F,**

**0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"n",110\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xC0, 0xE0, 0xF0, 0xF8, 0x38, 0x3C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x3C, 0xF8, 0xF8, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03,**

**0x0F, 0x0F, 0x1F, 0x3C, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x1F, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"o",111\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0xE0, 0x70, 0x38, 0x3C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x3C, 0xFC, 0xF8, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xFF, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1E, 0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F,**

**0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"p",112\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x80, 0xE0, 0xF0, 0xF8, 0x78, 0x3C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7C,**

**0xFF, 0xFF, 0xFF, 0x81, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03,**

**0x0F, 0x1F, 0x3F, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x1E, 0x0F, 0x07, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00}, /\*"q",113\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0xE0, 0xF0, 0x78, 0x38, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x3C, 0xFC, 0xF8, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"r",114\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xC0, 0xF0, 0xF8, 0xF8, 0x3C, 0x1C, 0x1C, 0x1C, 0x0C, 0x0C, 0x0C, 0x1C, 0x1C, 0x1C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x01, 0x07, 0x07, 0x0F, 0x0F, 0x1E, 0x1C, 0x1C, 0x3C, 0x38, 0x38, 0x78, 0xF0, 0xF0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"s",115\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1C,**

**0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFF, 0xFF, 0xFF, 0xFF, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x01, 0x0F, 0x1F, 0x1F, 0x3E, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"t",116\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x0F, 0x1F, 0x3F, 0x3E, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x0E, 0x0F, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"u",117\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x0C, 0x7C, 0xFC, 0xFC, 0xF0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF0, 0xFC, 0xFC, 0x7C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x01, 0x0F, 0x3F, 0xFF, 0xFE, 0xF0, 0xC0, 0x00, 0x00, 0x80, 0xF0, 0xFC, 0xFF, 0x3F, 0x0F, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x3F, 0x3E, 0x3E, 0x3F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"v",118\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3C,**

**0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x1F,**

**0xFF, 0xFF, 0xFC, 0x00, 0x00, 0xE0, 0xFC, 0xFF, 0x1F, 0x1F, 0xFF, 0xFC, 0xF0, 0x80, 0x00, 0xF0, 0xFF, 0xFF, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07,**

**0x3F, 0x3F, 0x3E, 0x3F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x3F, 0x3C, 0x3F, 0x3F, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"w",119\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x04, 0x0C, 0x1C, 0x3C, 0xFC, 0xF8, 0xF0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0xFC, 0x7C, 0x3C, 0x0C, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x81, 0xC3, 0xF7, 0xFF, 0x7F, 0x7E, 0xFF, 0xFF, 0xE7, 0xC1, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x20, 0x30,**

**0x3C, 0x3E, 0x3F, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x0F, 0x1F, 0x3E, 0x3C, 0x38, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"x",120\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x0C, 0x7C, 0xFC, 0xFC, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xFC, 0xFC, 0x7C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x01, 0x07, 0x3F, 0xFF, 0xFE, 0xF8, 0xC0, 0x00, 0x00, 0x80, 0xF0, 0xFC, 0xFF, 0x3F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x03, 0x9F, 0xFF, 0xFF, 0xFC, 0xFF, 0x3F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x38, 0x38, 0x38, 0x38,**

**0x38, 0x3C, 0x3C, 0x1E, 0x0F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"y",121\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x9C, 0xDC, 0xFC, 0xFC, 0xFC, 0x7C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF8, 0x7C, 0x3E, 0x1F, 0x0F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x3C, 0x3E, 0x3F, 0x3F, 0x3F, 0x3B, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"z",122\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF8, 0xFC, 0x3C, 0x1E, 0x0E, 0x0E, 0x0E, 0x0E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x1F, 0xFB, 0xFB, 0xF1, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x7F, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x0F, 0x1F, 0x3E, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"{",123\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"|",124\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x0E, 0x0E, 0x0E, 0x0E, 0x1E, 0x3C, 0xFC, 0xF8, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7F, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF1, 0xFB, 0xFB, 0x1B, 0x0F, 0x0E, 0x0E, 0x0E, 0x0E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1F, 0x1F, 0x0F, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"}",125\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0x3E,**

**0x3F, 0x3F, 0x07, 0x03, 0x03, 0x03, 0x07, 0x0F, 0x1F, 0x3E, 0x3C, 0x78, 0x70, 0x70, 0x70, 0x78, 0x7F, 0x3F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"~",126\*/**

**};**

**// 32\*32 消防预警(思源宋体Heavy)**

**const unsigned char FireWarning\_32x32[][128] PROGMEM = {**

**{**

**0x00, 0x00, 0x00, 0x00, 0x02, 0x06, 0x1E, 0x7E, 0x7C, 0x7C, 0x38, 0x00, 0xC0, 0x58, 0x78, 0xF8, 0xF0, 0xE0, 0x00, 0x00, 0xFE, 0xFE,**

**0xFE, 0xFE, 0x02, 0x00, 0xF0, 0xFC, 0x78, 0x30, 0x10, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x3F, 0x3E, 0x3E, 0x1C, 0xC0, 0xF0, 0x38, 0x06,**

**0x01, 0xFE, 0xFC, 0xFD, 0xFB, 0x09, 0x08, 0x08, 0x0F, 0x0F, 0x0F, 0x0F, 0x08, 0x0B, 0xFD, 0xFE, 0xFC, 0xFC, 0x08, 0x00, 0x00, 0x00,**

**0x00, 0x20, 0x60, 0xF0, 0xFC, 0xFF, 0x0F, 0x01, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21,**

**0x21, 0x21, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7F, 0x7F, 0x7F, 0x7F, 0x3C, 0x00, 0x00, 0x00, 0x7F,**

**0x7F, 0x7F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x04, 0x04, 0x0C, 0x7C, 0x7F, 0x7F, 0x3F, 0x1F, 0x00, 0x00, /\*"消",0\*/**

**},**

**{**

**0x00, 0x00, 0xFE, 0xFC, 0xFC, 0xF8, 0x08, 0x08, 0x88, 0xFC, 0xFE, 0xFE, 0xFC, 0x98, 0x90, 0x80, 0x80, 0x82, 0xBE, 0xFE, 0xFC, 0xFC,**

**0xB8, 0x80, 0x80, 0xC0, 0xE0, 0xF0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x10, 0x2F, 0xE7, 0xC3, 0x80,**

**0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x27, 0x20, 0x20, 0x20, 0x20, 0xF0, 0xF8, 0xF8, 0xF0, 0x60, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x20, 0x60, 0xE0, 0xE1, 0xFF, 0xFF, 0xFF, 0x7C, 0xC0, 0xFE, 0xFF, 0xFF, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00,**

**0xF0, 0xFF, 0xFF, 0xFF, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x7F, 0x7F, 0x00, 0x00, 0x03, 0x03, 0x43, 0x33, 0x19, 0x0E,**

**0x0F, 0x03, 0x01, 0x00, 0x04, 0x04, 0x0C, 0x7C, 0x7C, 0x7E, 0x3F, 0x3F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, /\*"防",0\*/**

**},**

**{**

**0x00, 0x00, 0x10, 0x90, 0x10, 0x10, 0x10, 0x10, 0xF0, 0xF8, 0xFC, 0x7C, 0x78, 0x30, 0x30, 0x10, 0x10, 0x10, 0x10, 0xF0, 0xF0, 0xF0,**

**0x70, 0x30, 0x10, 0x18, 0x1C, 0x1E, 0x1C, 0x18, 0x10, 0x00, 0x00, 0x20, 0x20, 0x21, 0x23, 0xEF, 0xFF, 0xFF, 0xFF, 0x21, 0x30, 0xF8,**

**0xF0, 0x60, 0xFF, 0xFE, 0xFE, 0xFC, 0x04, 0x07, 0xE7, 0xE5, 0xE4, 0xE4, 0x24, 0x24, 0xFE, 0xFF, 0xFE, 0xFC, 0x04, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x06, 0x03, 0x01, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0xC0, 0xFF, 0xFF, 0xFF, 0x1F,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x08, 0x08, 0x18, 0xF8, 0xFF, 0x7F, 0x7F, 0x1F, 0x00, 0x00, 0x80, 0x80, 0x40,**

**0x43, 0x61, 0x31, 0x39, 0x1C, 0x1F, 0x0F, 0x07, 0x03, 0x06, 0x0E, 0x1E, 0x3F, 0x7C, 0xFC, 0xF8, 0x70, 0x00, /\*"预",0\*/**

**},**

**{**

**0x00, 0x10, 0x10, 0xD0, 0xFC, 0xFC, 0xFC, 0xD4, 0x90, 0x90, 0xBC, 0xBC, 0xBC, 0xD4, 0xD8, 0x9C, 0x98, 0x90, 0xF0, 0xFE, 0x7E, 0xFC,**

**0xBC, 0x2C, 0xE4, 0xE0, 0xF0, 0xFC, 0x78, 0x30, 0x20, 0x00, 0x00, 0x02, 0x03, 0x3F, 0x3F, 0x3E, 0x0A, 0x0A, 0x1F, 0x1F, 0x0E, 0x3A,**

**0x38, 0x7F, 0xFF, 0xFF, 0x91, 0x91, 0x10, 0x08, 0x08, 0x0D, 0x0F, 0x07, 0x0F, 0x8F, 0xDF, 0xDC, 0xBC, 0xB8, 0x18, 0x08, 0x00, 0x01,**

**0x01, 0x01, 0x01, 0x01, 0x81, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x6D, 0x7F, 0xFF,**

**0x6D, 0x49, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0x7F, 0x7F, 0x7E, 0x22, 0x22, 0x22, 0x22,**

**0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0xFF, 0x7F, 0x7F, 0x7E, 0x02, 0x00, 0x00, 0x00, 0x00, 0x00, /\*"警",0\*/**

**},**

**};**

**// 32\*64 消防预警(思源宋体Heavy)**

**const unsigned char FireWarning\_32x64[][256] PROGMEM = {**

**{0x00, 0x00, 0x00, 0x00, 0x04, 0x3C, 0xF8, 0xF8, 0xF0, 0xE0, 0x80, 0x00, 0x20, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x0C,**

**0x04, 0x00, 0x00, 0xF0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x1F, 0x3F, 0x3F, 0x0F, 0xE0, 0x18, 0x01, 0x7F, 0xFF,**

**0xFE, 0xF8, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x3F, 0x0F, 0x07, 0x00, 0x00, 0x02, 0x0E, 0xFE, 0xFC, 0xF8, 0xF8, 0xE0,**

**0x00, 0x80, 0xFC, 0x07, 0xFC, 0xF0, 0xE0, 0xE7, 0xC7, 0xC3, 0xC0, 0xFF, 0xFF, 0xFF, 0xFF, 0xC0, 0xC0, 0xCF, 0xF3, 0xFC, 0xF8, 0xE0, 0x80, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x07, 0x0F, 0x0F, 0x03, 0xF8, 0xFF, 0x01, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF,**

**0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x04, 0x04, 0x1C, 0xFF, 0xFF, 0xFF,**

**0x80, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10, 0x30, 0x70,**

**0xF0, 0xF0, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1F, 0x3F, 0x3F, 0x1F, 0x07, 0x00, 0x00, 0x3F, 0x3F, 0x1F, 0x0F,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x3F, 0x1F, 0x1F, 0x0F, 0x01, 0x00, 0x00}, /\*"消",0\*/**

**{0x00, 0xF0, 0xE0, 0xE0, 0xC0, 0x80, 0x80, 0x80, 0xE0, 0xF8, 0xF0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x1C, 0xF8, 0xF8, 0xF0, 0xC0, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x01, 0x01, 0x81, 0xFF, 0xFF, 0xFF, 0xFF, 0x8F, 0x86, 0x80, 0x80,**

**0x80, 0x80, 0xBF, 0xFF, 0xFF, 0xFF, 0x9F, 0x80, 0x80, 0xC0, 0xF8, 0xFE, 0xF8, 0xE0, 0xC0, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0xFF,**

**0x7F, 0x1F, 0x03, 0x01, 0x01, 0x01, 0x01, 0x01, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x01, 0x01, 0x01, 0x01, 0x01, 0xC1, 0x01, 0x01, 0x01, 0x01, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x06, 0x1F, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x3F, 0x18, 0x18, 0x18,**

**0x18, 0xFE, 0xFF, 0xFF, 0xFC, 0x30, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x03, 0xFF, 0xFF, 0xFE, 0xE0, 0x00, 0x00, 0xFE,**

**0xFF, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x08, 0x18, 0xF8,**

**0xFC, 0xFF, 0xFF, 0xFF, 0x7F, 0x00, 0xF8, 0xFF, 0xFF, 0xFF, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFE, 0xFF, 0xFF, 0xFF, 0x3F, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x1F, 0x0F, 0x0F, 0x07, 0x03, 0xE0, 0xFF, 0xFF, 0x3F, 0x07, 0x00, 0x40, 0xC0, 0xC0, 0xC0, 0xC0, 0xF8,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7F, 0x7F, 0x7F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x60, 0x18, 0x0F, 0x07, 0x01, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x01, 0x7F, 0x7F, 0x3F, 0x3F, 0x1F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"防",1\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xF0, 0xC0, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80,**

**0x80, 0x80, 0xE0, 0xF8, 0xFE, 0xF8, 0xE0, 0x80, 0x00, 0x00, 0x03, 0xC3, 0x83, 0x83, 0x03, 0x03, 0xFB, 0xFF, 0xFF, 0x7F, 0x3F, 0x0F, 0x07, 0x05,**

**0x01, 0x01, 0x01, 0x01, 0xFF, 0xFF, 0xFF, 0x3F, 0x03, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x01, 0x07, 0x7F, 0xFF, 0xFF,**

**0xFF, 0x01, 0x00, 0xC0, 0xC0, 0x80, 0x00, 0xFF, 0xFE, 0xFC, 0xFC, 0x18, 0x1F, 0x1F, 0x19, 0x18, 0x18, 0x18, 0x18, 0xFE, 0xFF, 0xFE, 0xF8, 0x10,**

**0x00, 0x06, 0x06, 0x06, 0x06, 0xFE, 0xFF, 0xFF, 0xFF, 0x06, 0x07, 0xFF, 0xFF, 0xFF, 0x1E, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0xFE, 0xFE, 0xFE,**

**0xFE, 0x06, 0x02, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x0E, 0x07, 0x03, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF,**

**0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0xF0, 0xFF, 0xFF, 0xFF, 0x03, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x7F, 0x00,**

**0x00, 0x40, 0xC0, 0xC0, 0xC0, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x83, 0xE1, 0xFC, 0xFF, 0x7F, 0x1F, 0x07,**

**0x1E, 0x7E, 0xFC, 0xFC, 0xF8, 0xF8, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x03, 0x7F, 0x3F, 0x3F, 0x1F, 0x03, 0x00, 0x00, 0x40, 0x60, 0x20, 0x30, 0x1C,**

**0x1E, 0x0F, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x1F, 0x3F, 0x7F, 0x1F}, /\*"预",2\*/**

**{0x00, 0x80, 0x80, 0x80, 0xF8, 0xF8, 0xF8, 0x98, 0x80, 0x80, 0xF8, 0xF8, 0xF8, 0x98, 0xE0, 0xF0, 0xC0, 0x80, 0x00, 0xC0, 0xFC, 0xF8, 0xF0, 0xF0,**

**0x20, 0x00, 0x00, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0xF9, 0xFF, 0xFF, 0xDF, 0xD1, 0xC1, 0xC1, 0xC7, 0xC7, 0xC7, 0xF1, 0xFD, 0xE1,**

**0x81, 0x01, 0xC0, 0xFF, 0xFF, 0xFF, 0x8F, 0x0C, 0xFC, 0xFC, 0xFF, 0xFF, 0x1F, 0x0F, 0x0E, 0x00, 0x00, 0x70, 0x1F, 0xFF, 0xFF, 0xF9, 0x18, 0x18,**

**0xFE, 0xFF, 0xFC, 0x10, 0x80, 0xFF, 0xFF, 0xFF, 0x01, 0x0C, 0x07, 0x03, 0x00, 0xC3, 0xFF, 0xFE, 0xFF, 0xFF, 0xFF, 0xF1, 0xF0, 0xF0, 0xE0, 0x20,**

**0x00, 0x00, 0x00, 0x3F, 0x3F, 0x1F, 0x03, 0x03, 0x0F, 0x0F, 0x07, 0x07, 0x3F, 0x3F, 0xFF, 0xE3, 0xF0, 0xE8, 0xCC, 0x06, 0x07, 0x03, 0x01, 0x00,**

**0x01, 0x07, 0x8F, 0xDF, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x08, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x18, 0x1F,**

**0x1F, 0x1F, 0x1B, 0x18, 0x18, 0x98, 0xF8, 0xF8, 0x98, 0x1C, 0x1F, 0x1F, 0x1F, 0x1E, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x63,**

**0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x7F, 0xFF, 0x7B, 0x63, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFE, 0xFE, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C, 0xFF, 0xFF,**

**0xFF, 0xFC, 0x08, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x1F, 0x06, 0x06, 0x06, 0x06, 0x06, 0x06,**

**0x06, 0x06, 0x06, 0x06, 0x06, 0x06, 0x3F, 0x3F, 0x1F, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"警",3\*/**

**};**

**// 24\*48 数码管字体(DS-Digital)[下方空出高16的任务栏空间]**

**const unsigned char asc2\_Digital\_2448[][144] PROGMEM = {**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*" ",32\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x18, 0x3C, 0x3E, 0x3E, 0x3E, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"!",33\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x0C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*""",34\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x1C, 0x1C, 0x1C, 0x1C, 0xFC, 0xFF, 0xFF, 0x7F, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFC, 0xFF, 0xFF, 0x1F, 0x1C, 0x1C, 0x1C, 0x1C, 0x00, 0x00, 0x70, 0x70,**

**0x70, 0x70, 0xF0, 0xFF, 0xFF, 0xFF, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xFF, 0xFF, 0x7F, 0x70, 0x70, 0x70, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x20, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"#",35\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xE0, 0xF0, 0x70, 0x70, 0x70, 0xFC, 0xFF, 0xFF, 0x73, 0x70, 0x70, 0x70, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x3F, 0x7F, 0xFF, 0xFF, 0xE0, 0xE0, 0xC0, 0xC0, 0xFE, 0xFF, 0xFF, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x83, 0xFF, 0xFF, 0x7F, 0x07, 0x0F, 0x0F, 0x1E, 0x3E, 0xFC, 0xFC, 0xF8, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x1C, 0x1C,**

**0x3C, 0x38, 0x38, 0x38, 0xF8, 0xFF, 0xFF, 0x3F, 0x38, 0x38, 0x3C, 0x1C, 0x1E, 0x1F, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"$",36\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xF0, 0xF8, 0xFC, 0x1C, 0x0C, 0x0E, 0x0E, 0x1C, 0xFC, 0xF8, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF8, 0x3C, 0x1C, 0x0C, 0x00, 0x00, 0x07,**

**0x0F, 0x1F, 0x1C, 0x18, 0x18, 0x18, 0x1C, 0x1F, 0x0F, 0xC7, 0xE0, 0xF8, 0x7C, 0x3F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x80, 0xE0, 0xF0, 0xFC, 0x3E, 0x1F, 0x07, 0xE3, 0xF0, 0xF8, 0x38, 0x1C, 0x1C, 0x1C, 0x38, 0xF8, 0xF8, 0xE0, 0x00, 0x00, 0x30, 0x38, 0x3C,**

**0x1F, 0x0F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x3F, 0x38, 0x38, 0x30, 0x30, 0x38, 0x3F, 0x1F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"%",37\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xC0, 0xF0, 0xF8, 0xF8, 0x38, 0x1C, 0x1C, 0x1C, 0x1C, 0x38, 0xF8, 0xF8, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x0F, 0x3F, 0xFF, 0xFF, 0xF0, 0xE0, 0xE0, 0xE0, 0x70, 0x78, 0x3F, 0x1F, 0x0F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF8,**

**0xFE, 0xFF, 0x1F, 0x07, 0x03, 0x01, 0x07, 0x0F, 0x1F, 0x3F, 0x7C, 0xF8, 0xF0, 0xE0, 0xC0, 0xFF, 0xFF, 0xFF, 0x1F, 0x00, 0x00, 0x00, 0x01, 0x07, 0x0F,**

**0x1F, 0x1E, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x1F, 0x0F, 0x0F, 0x1F, 0x3F, 0x3C, 0x38, 0x30, 0x20, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"&",38\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"'",39\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0xF8, 0x7C, 0x3E, 0x0F, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xFC, 0xFF, 0xFF, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x1C, 0xFF, 0xFF, 0xFF, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x03, 0x1F, 0x3F, 0xFF, 0xFC, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x0F, 0x1F, 0x3E, 0x3C, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"(",40\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x06, 0x0E, 0x1E, 0x7C, 0xF8, 0xF0, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x7F, 0xFF, 0xFC, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xFF, 0xFF, 0xFF, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF8, 0xFF, 0x7F, 0x1F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x18, 0x3C, 0x3E, 0x1F, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*")",41\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xC0, 0xE0, 0xC0, 0xC0, 0x80, 0x00, 0x00, 0xFC, 0xFC, 0x0C, 0x00, 0x80, 0xC0, 0xC0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x60, 0x70, 0x71, 0x39, 0x19, 0x1F, 0x0F, 0xFF, 0xFF, 0x0F, 0x1F, 0x1B, 0x39, 0x71, 0x70, 0x60, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"\*",42\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFE, 0xFE, 0xFE, 0xFE, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1C,**

**0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFF, 0xFF, 0xFF, 0xFF, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1F, 0x1F, 0x1F, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"+",43\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x1F, 0xFF, 0xFF, 0xFF, 0xFE, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x1C, 0x1C, 0x1C, 0x0C, 0x0E, 0x0F, 0x07, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*",",44\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x1E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"-",45\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x0C, 0x1E, 0x3F, 0x3F, 0x3F, 0x3F, 0x1E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*".",46\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 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, 0xF8, 0xFC, 0x7C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xFC, 0xFF, 0x1F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF8, 0xFE, 0x3F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xC0, 0xF0, 0xFE, 0x7F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03,**

**0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"/",47\*/**

**{**

**0x00, 0x00, 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, 0xD0, 0xB0, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0xB0, 0xD0, 0xE0, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0xC7, 0xEF, 0xC7, 0x83, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x83, 0xC7, 0xEF, 0xC7, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x1F, 0x2F, 0x37, 0x3B, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3B, 0x37, 0x2F, 0x1F, 0x00, 0x00,**

**}, /\*"0",0\*/**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x83, 0xC7, 0xEF, 0xC7, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x0F, 0x1F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**}, /\*"1",1\*/**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x10, 0x30, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0xB0, 0xD0, 0xE0, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x80, 0xC0, 0x80, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x7F, 0x3F, 0x0F, 0x0F, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x1F, 0x2F, 0x37, 0x3B, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x38, 0x30, 0x20, 0x00, 0x00, 0x00,**

**}, /\*"2",2\*/**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x10, 0x30, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0xB0, 0xC0, 0xE0, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0xB3, 0xC7, 0xEF, 0xC7, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x20, 0x30, 0x38, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3B, 0x37, 0x0F, 0x1F, 0x00, 0x00,**

**}, /\*"3",3\*/**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xF0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x07, 0x0F, 0x07, 0x7B, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0xFB, 0xF7, 0xEF, 0xC7, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x0F, 0x1F, 0x3F, 0x00, 0x00,**

**}, /\*"4",4\*/**

**{**

**0x00, 0x00, 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, 0xD0, 0xB0, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x30, 0x10, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x7F, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0xB0, 0xC0, 0x80, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x20, 0x30, 0x38, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3B, 0x37, 0x2F, 0x1F, 0x00, 0x00,**

**}, /\*"5",5\*/**

**{**

**0x00, 0x00, 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, 0xD0, 0xB0, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x30, 0x10, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xC7, 0xEF, 0xE7, 0xFB, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0xF8, 0xF0, 0xE0, 0xC0, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x1F, 0x2F, 0x37, 0x3B, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3B, 0x37, 0x2F, 0x1F, 0x00, 0x00,**

**}, /\*"6",6\*/**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x10, 0x30, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0xB0, 0xC0, 0xE0, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x83, 0xC7, 0xEF, 0xC7, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x0F, 0x1F, 0x3F, 0x00, 0x00,**

**}, /\*"7",7\*/**

**{**

**0x00, 0x00, 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, 0xD0, 0xB0, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0xB0, 0xD0, 0xE0, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0xC7, 0xEF, 0xE7, 0xFB, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0xFB, 0xF7, 0xEF, 0xC7, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x1F, 0x2F, 0x37, 0x3B, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3B, 0x37, 0x2F, 0x1F, 0x00, 0x00,**

**}, /\*"8",8\*/**

**{**

**0x00, 0x00, 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, 0xD0, 0xB0, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0xB0, 0xD0, 0xF0, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x07, 0x0F, 0x07, 0x7B, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0x78, 0xFB, 0xF7, 0xEF, 0xC7, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x20, 0x30, 0x38, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3C, 0x3B, 0x37, 0x2F, 0x1F, 0x00, 0x00,**

**}, /\*"9",9\*/**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**}, /\*":",10\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x38, 0x7C, 0xFC, 0xFC, 0x7C, 0x78, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x0E, 0x1F, 0xFF, 0xFF, 0xFF, 0xFE, 0x3C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x1C, 0x1C, 0x1C, 0x1C, 0x0E, 0x0F, 0x07, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*";",59\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF0, 0xF0, 0x78, 0x3C, 0x1E, 0x1E, 0x0F, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x08, 0x0C, 0x1E, 0x3F, 0x7F, 0x7F, 0xF3, 0xE1, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x07, 0x0F, 0x1E, 0x3C, 0x3C, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"<",60\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0xE1, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"=",61\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x06, 0x0F, 0x0F, 0x1E, 0x3C, 0x78, 0x70, 0xF0, 0xE0, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE1, 0xF3, 0xF7, 0x7F, 0x3F, 0x1E, 0x0C, 0x08, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x18, 0x38, 0x3C, 0x1E, 0x0F, 0x07, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*">",62\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1C, 0x1C, 0x1C, 0x1C, 0x3C, 0x3C, 0x78, 0x78, 0xF0, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xC0, 0xE1, 0xFF, 0xFF, 0x7F, 0x1E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x3C, 0x3E, 0x3E, 0x3E, 0x3C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"?",63\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF0, 0x38, 0x38, 0x1C, 0x0C, 0x0C, 0x0E, 0x0E, 0x0C, 0x0C, 0x1C, 0x3C, 0xF8, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0xC0,**

**0xF8, 0xFE, 0x3F, 0x07, 0x01, 0x00, 0xC0, 0xE0, 0xF0, 0x70, 0x30, 0x30, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x07, 0xFF, 0xFF, 0xF8, 0xFC, 0xFF, 0xFF,**

**0x07, 0x00, 0x00, 0xF8, 0xFF, 0xFF, 0x1F, 0x00, 0x00, 0x00, 0xE0, 0xFF, 0xFF, 0xFF, 0x01, 0x00, 0x00, 0xE0, 0xFF, 0xFF, 0x1F, 0x07, 0xFF, 0xFF, 0xF8,**

**0x80, 0x00, 0x03, 0x0F, 0x0F, 0x1F, 0x1C, 0x0C, 0x0E, 0x07, 0x07, 0x0F, 0x1F, 0x1C, 0x1C, 0x0E, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x03, 0x07, 0x0F,**

**0x1E, 0x1C, 0x38, 0x38, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x38, 0x18, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"@",64\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xF0, 0xFC, 0xFF, 0x7F, 0x0F, 0x01, 0x01, 0x07, 0x3F, 0xFF, 0xFE, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xE0, 0xFC, 0xFF, 0xFF, 0xEF, 0xE1, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE1, 0xEF, 0xFF, 0xFF, 0xFE, 0xF0, 0x80, 0x00, 0x00, 0x20, 0x3C, 0x3F, 0x3F,**

**0x1F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x1F, 0x3F, 0x3F, 0x3C, 0x20, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"A",65\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xF0, 0xE0, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xC0, 0xF1, 0xFF, 0x7F, 0x3F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x07, 0x0F, 0x1F, 0xFE, 0xFE, 0xFC, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F,**

**0x3F, 0x3F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x1C, 0x1F, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"B",66\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xE0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xE0, 0xFC, 0xFF, 0xFF, 0x1F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F,**

**0xFF, 0xFF, 0xFF, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01,**

**0x03, 0x07, 0x0F, 0x1F, 0x1E, 0x3C, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"C",67\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xE0, 0xE0, 0xC0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x7F, 0xFF, 0xFF, 0xFC, 0xC0, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF8, 0xFF, 0xFF, 0x7F, 0x07, 0x00, 0x00, 0x00, 0x3F, 0x3F,**

**0x3F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x1C, 0x1E, 0x0F, 0x0F, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"D",68\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x3F, 0x3F, 0x3F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"E",69\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"F",70\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xE0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x78, 0x78, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x80,**

**0xF8, 0xFE, 0xFF, 0x7F, 0x0F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x07, 0x7F,**

**0xFF, 0xFF, 0xF8, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01,**

**0x07, 0x0F, 0x0F, 0x1F, 0x1E, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3F, 0x3F, 0x1F, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"G",71\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0x03, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F,**

**0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"H",72\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3F, 0x3F, 0x3F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"I",73\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x1C, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1E, 0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"J",74\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF0, 0xF0, 0x70, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0xC0, 0xE0, 0xF0, 0xF8, 0x3C, 0x1E, 0x0F, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x03, 0x07, 0x0F, 0x1F, 0x3E, 0xFC, 0xF8, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F,**

**0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x1F, 0x3F, 0x3E, 0x3C, 0x30, 0x20, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"K",75\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"L",76\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0x00, 0x07, 0x3F, 0xFF, 0xFC, 0xE0, 0x00, 0x00, 0xE0, 0xFC, 0xFF, 0x1F, 0x03, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xE0, 0xFF,**

**0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x1F, 0x1F, 0x1F, 0x03, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFC, 0x00, 0x00, 0x3F, 0x3F, 0x3F,**

**0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"M",77\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0x01, 0x07, 0x3F, 0xFF, 0xFC, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x3F, 0xFF, 0xFC, 0xF0, 0xC0, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F,**

**0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x3F, 0x3F, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"N",78\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xE0, 0xF0, 0x70, 0x70, 0x78, 0x78, 0x70, 0x70, 0x70, 0xF0, 0xE0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0xE0,**

**0xFC, 0xFF, 0xFF, 0x1F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0xFF, 0xFF, 0xFE, 0xF0, 0x00, 0x00, 0x1F, 0xFF,**

**0xFF, 0xFF, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xFF, 0xFF, 0xFF, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x03,**

**0x0F, 0x0F, 0x1F, 0x3E, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1E, 0x1F, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"O",79\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xE0, 0xE0, 0xC0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC3, 0xFF, 0xFF, 0xFF, 0x3E, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1E, 0x0E, 0x0E, 0x0E, 0x0F, 0x07, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F,**

**0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"P",80\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xE0, 0xF0, 0x70, 0x70, 0x78, 0x78, 0x70, 0x70, 0xF0, 0xF0, 0xE0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0xC0,**

**0xFC, 0xFF, 0xFF, 0x1F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0xFF, 0xFF, 0xFE, 0xF0, 0x00, 0x00, 0x1F, 0xFF,**

**0xFF, 0xFF, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xFF, 0xFF, 0xFF, 0x1F, 0x00, 0x00, 0x00, 0x01, 0x03,**

**0x0F, 0x0F, 0x1F, 0x3E, 0x3C, 0x38, 0xF8, 0xF8, 0xF8, 0xF8, 0x38, 0x3C, 0x1E, 0x1F, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x0F, 0x1F, 0x1E, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1E, 0x0E, 0x0C}, /\*"Q",81\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xF0, 0xE0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xFF, 0xFF, 0x7F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0x03, 0x03, 0x03, 0x03, 0x07, 0x07, 0x1F, 0x3F, 0xFD, 0xF9, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x3F, 0x3F, 0x3C, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"R",82\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xC0, 0xE0, 0xE0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x78, 0x78, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x3F, 0x7F, 0xFF, 0xFF, 0xF0, 0xE0, 0xC0, 0xC0, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x03, 0x03, 0x03, 0x07, 0x07, 0x0F, 0x0F, 0x1E, 0xFE, 0xFC, 0xF8, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1C, 0x3C,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1C, 0x1F, 0x0F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"S",83\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"T",84\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07,**

**0x0F, 0x1F, 0x1F, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1F, 0x0F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"U",85\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10,**

**0xF0, 0xF0, 0xF0, 0xF0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF0, 0xF0, 0xF0, 0x10, 0x00, 0x00,**

**0x07, 0x1F, 0xFF, 0xFF, 0xFC, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF8, 0xFF, 0xFF, 0x3F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x07, 0x3F, 0xFF, 0xFF, 0xF8, 0xC0, 0x00, 0x00, 0x80, 0xF0, 0xFE, 0xFF, 0x3F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x01, 0x07, 0x3F, 0x3F, 0x3E, 0x3C, 0x3F, 0x3F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"V",86\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x07,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFE, 0xFF, 0xFF, 0x3F, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0xF8, 0x00, 0xC0, 0xF8, 0xFF, 0x3F, 0x07, 0x07, 0x3F, 0xFF, 0xFC, 0xE0, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x01, 0x3F,**

**0x3F, 0x3F, 0x3E, 0x3F, 0x0F, 0x01, 0x00, 0x00, 0x00, 0x00, 0x01, 0x0F, 0x3F, 0x3F, 0x3C, 0x3F, 0x3F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"W",87\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x30, 0x70, 0xF0, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0xF0, 0x70, 0x30, 0x10, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x01, 0x03, 0x0F, 0x1F, 0x7F, 0xFC, 0xF8, 0xE0, 0xE0, 0xF0, 0xFC, 0x7F, 0x1F, 0x0F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x80, 0xE0, 0xF0, 0xFC, 0x7E, 0x3F, 0x0F, 0x0F, 0x3F, 0x7E, 0xFC, 0xF0, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0x38, 0x3C,**

**0x3F, 0x1F, 0x0F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x0F, 0x1F, 0x3F, 0x3E, 0x38, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"X",88\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10,**

**0x30, 0xF0, 0xF0, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xF0, 0xF0, 0x70, 0x10, 0x00, 0x00,**

**0x00, 0x03, 0x07, 0x1F, 0x7F, 0xFE, 0xFC, 0xF0, 0xC0, 0x00, 0x00, 0x80, 0xE0, 0xF8, 0xFC, 0x7F, 0x1F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x07, 0xFF, 0xFF, 0xFE, 0xFF, 0x0F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"Y",89\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x70, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0xFC, 0x7E, 0x3F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF8, 0x7E, 0x3F, 0x1F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x38, 0x3C,**

**0x3F, 0x3F, 0x3F, 0x3B, 0x39, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"Z",90\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFE, 0xFE, 0xFE, 0xFE, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"[",91\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x0C, 0x3C, 0xFC, 0xF8, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x7F, 0xFE, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x0F, 0x3F, 0xFF, 0xFC, 0xF0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x0F, 0x7F, 0xFE, 0xF8, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00}, /\*"\",92\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0xFE, 0xFE, 0xFE, 0xFE, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"]",93\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0x70, 0xF0, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x80, 0xC0, 0xF0, 0xFC, 0x3E, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x01, 0x07, 0x1F, 0x3F, 0xFC, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"^",94\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x38, 0x38, 0x38, 0x38, 0x38,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38}, /\*"\_",95\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x04, 0x0C, 0x0C, 0x1C, 0x3C, 0x38, 0x70, 0x60, 0x40, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"`",96\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x38, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x7C, 0xF8, 0xF8, 0xF0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xC0, 0xE0, 0xF0, 0xF8, 0x38, 0x38, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07,**

**0x1F, 0x1F, 0x3F, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x1C, 0x0E, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"a",97\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xE0, 0x70, 0x38, 0x3C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x3C, 0xFC, 0xF8, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xFF, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1F,**

**0x1F, 0x1F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1E, 0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"b",98\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF0, 0xF8, 0x38, 0x3C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x3C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x3C, 0xFF, 0xFF, 0xFF, 0xC3, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x01, 0x07, 0x0F, 0x1F, 0x1E, 0x3C, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"c",99\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x80, 0xE0, 0xF0, 0xF8, 0x78, 0x3C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7C,**

**0xFF, 0xFF, 0xFF, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03,**

**0x0F, 0x1F, 0x3F, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x1E, 0x0F, 0x07, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"d",100\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x80, 0xE0, 0xF0, 0xF8, 0x78, 0x3C, 0x1C, 0x1C, 0x0C, 0x0C, 0x0C, 0x1C, 0x1C, 0x3C, 0xF8, 0xF8, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xDF, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1F, 0x1F, 0x1F, 0x1E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01,**

**0x07, 0x0F, 0x1F, 0x1E, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"e",101\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 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, 0xF0, 0xF8, 0xFC, 0x3C, 0x1C, 0x1C, 0x0C, 0x0E, 0x0E, 0x0C, 0x1C, 0x1C, 0x1C, 0x00, 0x00, 0x00,**

**0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xFF, 0xFF, 0xFF, 0xE1, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"f",102\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x80, 0xE0, 0xF0, 0xF8, 0x7C, 0x1C, 0x1C, 0x0C, 0x0C, 0x0C, 0x0C, 0x1C, 0x3C, 0xFC, 0xFC, 0xFC, 0xFC, 0x1C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x83, 0xCF, 0xFF, 0xFF, 0x7C, 0x70, 0x70, 0x60, 0x60, 0x60, 0x60, 0x70, 0x78, 0x3F, 0x3F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC7,**

**0xEF, 0xFF, 0x7E, 0x3C, 0x1C, 0x1C, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0xF8, 0xF0, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x07, 0x0F, 0x1F,**

**0x3F, 0x3C, 0x38, 0x38, 0x38, 0x30, 0x30, 0x30, 0x30, 0x38, 0x38, 0x38, 0x3C, 0x1F, 0x1F, 0x0F, 0x07, 0x00, 0x00}, /\*"g",103\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xE0, 0x70, 0x38, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x3C, 0xFC, 0xF8, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F,**

**0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"h",104\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x38, 0x7C, 0x7C, 0x7E, 0x7C, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3F, 0x3F, 0x3F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"i",105\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3C, 0x7C, 0x7E, 0x7C, 0x3C, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3C, 0x38,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1E, 0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"j",106\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xC0, 0xE0, 0xF0, 0x78, 0x3C, 0x1C, 0x0C, 0x0C, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x0C, 0x1E, 0x3F, 0x7F, 0xF7, 0xE3, 0xC1, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x0F, 0x1F, 0x3E, 0x3C, 0x38, 0x30, 0x20, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"k",107\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3F, 0x3F, 0x3F, 0x3F, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"l",108\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFC, 0xFC, 0xFC, 0xE0, 0x78, 0x1C, 0x1C, 0x1C, 0xFC, 0xFC, 0xF8, 0xE0, 0xF0, 0x38, 0x1C, 0x0C, 0x1C, 0xFC, 0xFC, 0xF8, 0x00, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0x01, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F,**

**0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"m",109\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0xE0, 0x70, 0x38, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x3C, 0xFC, 0xF8, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F,**

**0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"n",110\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xC0, 0xE0, 0xF0, 0xF8, 0x38, 0x3C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x3C, 0xF8, 0xF8, 0xF0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03,**

**0x0F, 0x0F, 0x1F, 0x3C, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x1F, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"o",111\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0xE0, 0x70, 0x38, 0x3C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x3C, 0xFC, 0xF8, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xFF, 0xFF, 0xFF, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1E, 0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F,**

**0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"p",112\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x80, 0xE0, 0xF0, 0xF8, 0x78, 0x3C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7C,**

**0xFF, 0xFF, 0xFF, 0x81, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03,**

**0x0F, 0x1F, 0x3F, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x1E, 0x0F, 0x07, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00}, /\*"q",113\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0xE0, 0xF0, 0x78, 0x38, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x3C, 0xFC, 0xF8, 0xF0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x03, 0x03, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"r",114\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xC0, 0xF0, 0xF8, 0xF8, 0x3C, 0x1C, 0x1C, 0x1C, 0x0C, 0x0C, 0x0C, 0x1C, 0x1C, 0x1C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x01, 0x07, 0x07, 0x0F, 0x0F, 0x1E, 0x1C, 0x1C, 0x3C, 0x38, 0x38, 0x78, 0xF0, 0xF0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"s",115\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1C,**

**0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0xFF, 0xFF, 0xFF, 0xFF, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x01, 0x0F, 0x1F, 0x1F, 0x3E, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"t",116\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x0F, 0x1F, 0x3F, 0x3E, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x1C, 0x0E, 0x0F, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"u",117\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x0C, 0x7C, 0xFC, 0xFC, 0xF0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF0, 0xFC, 0xFC, 0x7C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x01, 0x0F, 0x3F, 0xFF, 0xFE, 0xF0, 0xC0, 0x00, 0x00, 0x80, 0xF0, 0xFC, 0xFF, 0x3F, 0x0F, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x3F, 0x3E, 0x3E, 0x3F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"v",118\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3C,**

**0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xFC, 0xFC, 0xFC, 0x00, 0x00, 0x00, 0x1F,**

**0xFF, 0xFF, 0xFC, 0x00, 0x00, 0xE0, 0xFC, 0xFF, 0x1F, 0x1F, 0xFF, 0xFC, 0xF0, 0x80, 0x00, 0xF0, 0xFF, 0xFF, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07,**

**0x3F, 0x3F, 0x3E, 0x3F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x07, 0x1F, 0x3F, 0x3C, 0x3F, 0x3F, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"w",119\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x04, 0x0C, 0x1C, 0x3C, 0xFC, 0xF8, 0xF0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF0, 0xFC, 0x7C, 0x3C, 0x0C, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x81, 0xC3, 0xF7, 0xFF, 0x7F, 0x7E, 0xFF, 0xFF, 0xE7, 0xC1, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x20, 0x30,**

**0x3C, 0x3E, 0x3F, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x0F, 0x1F, 0x3E, 0x3C, 0x38, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"x",120\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x0C, 0x7C, 0xFC, 0xFC, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xFC, 0xFC, 0x7C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x01, 0x07, 0x3F, 0xFF, 0xFE, 0xF8, 0xC0, 0x00, 0x00, 0x80, 0xF0, 0xFC, 0xFF, 0x3F, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x03, 0x9F, 0xFF, 0xFF, 0xFC, 0xFF, 0x3F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x38, 0x38, 0x38, 0x38,**

**0x38, 0x3C, 0x3C, 0x1E, 0x0F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"y",121\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x9C, 0xDC, 0xFC, 0xFC, 0xFC, 0x7C, 0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF8, 0x7C, 0x3E, 0x1F, 0x0F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x3C, 0x3E, 0x3F, 0x3F, 0x3F, 0x3B, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"z",122\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF8, 0xFC, 0x3C, 0x1E, 0x0E, 0x0E, 0x0E, 0x0E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x1F, 0xFB, 0xFB, 0xF1, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x7F, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x0F, 0x1F, 0x3E, 0x3C, 0x38, 0x38, 0x38, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"{",123\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0, 0xE0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x3F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"|",124\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x0E, 0x0E, 0x0E, 0x0E, 0x1E, 0x3C, 0xFC, 0xF8, 0xF0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7F, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF1, 0xFB, 0xFB, 0x1B, 0x0F, 0x0E, 0x0E, 0x0E, 0x0E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x38, 0x38, 0x38, 0x38, 0x38, 0x3C, 0x1F, 0x1F, 0x0F, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"}",125\*/**

**{0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0x3E,**

**0x3F, 0x3F, 0x07, 0x03, 0x03, 0x03, 0x07, 0x0F, 0x1F, 0x3E, 0x3C, 0x78, 0x70, 0x70, 0x70, 0x78, 0x7F, 0x3F, 0x1F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00}, /\*"~",126\*/**

**};**

**// 低电量.png**

**const unsigned char Low\_Power\_IMG[] PROGMEM = {**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0xF0, 0x70, 0x38, 0x38, 0x38, 0x38, 0x70, 0xF0,**

**0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xFE, 0xFC,**

**0xFC, 0xFC, 0x78, 0x18, 0x08, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFC, 0xFC, 0xFC, 0xFC, 0xFC, 0xF8, 0x08, 0x00,**

**0x00, 0x00, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0xFE, 0xFE, 0xFC, 0xFC, 0xFC, 0xFC, 0x04, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0xF8, 0x7C, 0x1F, 0x0F, 0x03, 0x00, 0x60, 0xF8, 0xFC, 0xFC,**

**0xF8, 0x60, 0x00, 0x03, 0x0F, 0x1F, 0x7E, 0xF8, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF8,**

**0x7F, 0x1F, 0x07, 0x03, 0xFE, 0xFE, 0xFE, 0xFE, 0xFE, 0xFE, 0x03, 0x03, 0x03, 0x83, 0x03, 0x02, 0x04, 0x04, 0x04, 0x04, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,**

**0xFF, 0x04, 0x04, 0xFE, 0xFF, 0xFF, 0xFF, 0xFE, 0xFE, 0x0C, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFE, 0x02, 0x02,**

**0x02, 0x02, 0x02, 0x02, 0x02, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x02, 0x02, 0x02, 0x02, 0x02, 0x02, 0x02, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFE, 0x06,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF0, 0xFC, 0x3E, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x0F, 0xFF, 0xFF, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x0F, 0x3F, 0xFC, 0xF0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x09, 0x88, 0x88, 0x88, 0x88, 0x88, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x08, 0x88, 0x8E, 0x8F, 0x8F, 0x8E, 0x8C, 0x88, 0x80, 0x80, 0xFF, 0xFF,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x80, 0x80, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,**

**0xFF, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0xFF, 0xFF, 0xFF, 0xFF,**

**0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xF8, 0x7E, 0x1F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x40, 0xF0, 0xF9, 0xF9, 0xF0, 0x40, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x1F, 0x7E, 0xF8, 0xE0, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x80,**

**0xFC, 0xFF, 0xFF, 0xFF, 0xFF, 0x1F, 0x01, 0x3E, 0xF8, 0xF0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x7F, 0x08, 0x08, 0x08, 0x08, 0x08, 0x08, 0x08, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x08, 0x08, 0x08, 0x08, 0x08, 0x08, 0x08, 0xFF,**

**0x7F, 0x7F, 0x7F, 0x7F, 0x3F, 0x00, 0xE0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x02, 0x1F, 0x3F, 0x38, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70,**

**0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x71, 0x71, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x70, 0x38,**

**0x3F, 0x1F, 0x02, 0x00, 0x00, 0x00, 0x00, 0x18, 0x7F, 0xFF, 0x7F, 0x3F, 0x3C, 0x3F, 0x1F, 0x1F, 0x1F, 0x0F, 0x0F, 0x04, 0x7F, 0x7F, 0x7F, 0xBF, 0xE0,**

**0xF0, 0xFC, 0x3F, 0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x01, 0x07, 0x1F, 0x7F, 0xFF, 0xFF, 0xFE, 0xFE, 0x3C, 0x18, 0x18, 0x08, 0x00, 0x00,**

**0x00, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x7F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFC, 0xF8, 0xF8, 0xF8, 0xF8,**

**0xF8, 0xF8, 0xF8, 0xF8, 0xF8, 0xF8, 0xF8, 0xFC, 0xFF, 0xFF, 0xFF, 0xF8, 0x78, 0x20, 0x00, 0x00, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20,**

**0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20,**

**0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x26,**

**0x23, 0x21, 0x21, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x21, 0x23, 0x20, 0x20, 0x20, 0x20,**

**0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x21, 0x21, 0x21, 0x21,**

**0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x20, 0x20, 0x20, 0x20, 0x20, 0x00, 0x00, 0x02, 0x02, 0xFE, 0xFE, 0xFE, 0x02,**

**0x02, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF8, 0xFC, 0x02, 0x01, 0x01, 0x01, 0x01, 0x03, 0x06, 0xFC, 0xF8, 0xC0, 0x02, 0x02, 0x0E, 0x7E,**

**0xFA, 0xC2, 0x00, 0x00, 0xE2, 0x1E, 0x3E, 0xFE, 0xE2, 0x00, 0x00, 0xC2, 0x3A, 0x06, 0x02, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0xFF,**

**0xFF, 0xFF, 0x01, 0x01, 0x01, 0x83, 0xFE, 0xFE, 0x10, 0x00, 0x00, 0xC0, 0xF8, 0xFC, 0x02, 0x01, 0x01, 0x01, 0x01, 0x03, 0x06, 0xFC, 0xF8, 0xC0, 0x02,**

**0x02, 0x0E, 0x7E, 0xFA, 0xC2, 0x00, 0x00, 0xE2, 0x1E, 0x3E, 0xFE, 0xE2, 0x00, 0x00, 0xC2, 0x3A, 0x06, 0x02, 0x00, 0x01, 0xFF, 0xFF, 0xFF, 0x81, 0x81,**

**0x81, 0xF1, 0x03, 0x0F, 0x00, 0x00, 0x01, 0x01, 0xFF, 0xFF, 0xFF, 0x41, 0x41, 0xC1, 0xC3, 0x7E, 0x3E, 0x18, 0x00, 0x00, 0x00, 0x00, 0x20, 0x20, 0x3F,**

**0x3F, 0x3F, 0x20, 0x20, 0x20, 0x20, 0x30, 0x38, 0x04, 0x00, 0x00, 0x00, 0x07, 0x0F, 0x18, 0x30, 0x20, 0x20, 0x20, 0x20, 0x10, 0x0F, 0x07, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x01, 0x0F, 0x7E, 0x06, 0x01, 0x00, 0x00, 0x01, 0x0F, 0x7F, 0x0C, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x20, 0x3F, 0x3F, 0x3F, 0x21, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x0F, 0x18, 0x30, 0x20, 0x20, 0x20, 0x20, 0x10, 0x0F,**

**0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x0F, 0x7E, 0x06, 0x01, 0x00, 0x00, 0x01, 0x0F, 0x7F, 0x0C, 0x03, 0x00, 0x00, 0x00, 0x00, 0x20, 0x3F, 0x3F,**

**0x3F, 0x20, 0x20, 0x20, 0x23, 0x20, 0x30, 0x0C, 0x00, 0x00, 0x20, 0x3F, 0x3F, 0x3F, 0x20, 0x00, 0x01, 0x07, 0x0F, 0x1C, 0x30, 0x20, 0x20, 0x00,**

**};**

**// 正在充电.png**

**const unsigned char Charging\_IMG[] PROGMEM = {**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFE, 0xFE, 0xFE, 0x00, 0x00, 0x00, 0xFE, 0xFE, 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, 0x00, 0x00, 0x00, 0x00, 0x00, 0x02, 0x06, 0x1E, 0x7E, 0xFE, 0xFE, 0xFE, 0xFC, 0x78, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0,**

**0xE0, 0xF0, 0xE0, 0xE0, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0xFE, 0xFE, 0xFE, 0xFE, 0xFE, 0xFC, 0xFC, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x3E, 0xFE, 0xFE, 0xFE, 0xFE, 0xFE, 0xFE, 0xFE, 0xFE, 0xFE, 0xFE, 0xFE, 0xFE, 0xFE, 0xFE, 0xFE, 0x3E, 0x00, 0x00, 0x00, 0x80,**

**0xE0, 0xF8, 0xF8, 0x7C, 0x3E, 0x1E, 0x0E, 0x0E, 0x0E, 0x1E, 0x3E, 0x7C, 0xF8, 0xF8, 0xE0, 0x80, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,**

**0x01, 0x01, 0x01, 0x01, 0x81, 0xC1, 0xE1, 0xF1, 0xF9, 0xFD, 0xFF, 0xFF, 0x7F, 0x7D, 0x3D, 0x39, 0x19, 0x11, 0x01, 0x01, 0x01, 0x11, 0x21, 0xE1, 0xE1,**

**0xE1, 0xC1, 0xC1, 0x81, 0x81, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFE, 0x02, 0x02,**

**0x02, 0x02, 0x02, 0x02, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x02, 0x02, 0x02, 0x02, 0x02, 0x02, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFE, 0x06,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x1F, 0x3F, 0x7F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x7F, 0x3F, 0x1F, 0x03, 0x00, 0x00,**

**0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x1C, 0xFC, 0xFE, 0xFE, 0xFF, 0xFF, 0xFB, 0xFB, 0xFB, 0xF9, 0xF9, 0xF9, 0xF8, 0xF8, 0xF8, 0x78, 0x78, 0x38, 0x38, 0xF8, 0xF8, 0xF8, 0xF8, 0xF8,**

**0xF8, 0x08, 0x09, 0x0F, 0x1F, 0x7F, 0xFF, 0xFF, 0xFF, 0xFE, 0x78, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,**

**0xFF, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,**

**0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0xFF, 0xFF, 0xFF, 0x07, 0x07, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0xE0, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x0F, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,**

**0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF,**

**0xFF, 0xFF, 0xFF, 0x7F, 0x08, 0x08, 0x08, 0x08, 0x08, 0x08, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x08, 0x08, 0x08, 0x08, 0x08, 0x08, 0xFF, 0xFF,**

**0x7F, 0x7F, 0x7F, 0x7F, 0x3F, 0x00, 0xE0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1F, 0x7F, 0xFF, 0xF8, 0xE0,**

**0xC0, 0x80, 0x80, 0x80, 0x80, 0x80, 0xC0, 0xF0, 0xFF, 0xFF, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF0, 0xF8, 0x7C, 0x7F, 0x3F, 0x1F, 0x1F, 0x0F, 0x07, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x7F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF8, 0xF8, 0xF8, 0xF8, 0xF8, 0xF8, 0xF8, 0xF8, 0xFE, 0xFF, 0xFF, 0xF8, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x7F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF8, 0xF8, 0xF8, 0xF8, 0xF8,**

**0xF8, 0xF8, 0xF8, 0xF8, 0xF8, 0xF8, 0xF8, 0xFC, 0xFF, 0xFF, 0xFF, 0xF8, 0x78, 0x20, 0x00, 0x00, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20,**

**0x21, 0x21, 0x23, 0x23, 0x27, 0x27, 0x27, 0x27, 0x27, 0x23, 0x23, 0x21, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20,**

**0x27, 0x27, 0x27, 0x20, 0x20, 0x20, 0x24, 0x22, 0x22, 0x21, 0x21, 0x21, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20,**

**0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20,**

**0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x21, 0x21, 0x21, 0x21, 0x21,**

**0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x21, 0x20, 0x20, 0x20, 0x20, 0x20, 0x00, 0x00, 0xC0, 0xF8, 0xFC, 0x06, 0x03, 0x01,**

**0x01, 0x01, 0x01, 0x02, 0x06, 0x08, 0x00, 0x00, 0x00, 0x00, 0x00, 0x02, 0xFE, 0xFE, 0xFE, 0x82, 0x80, 0x80, 0x80, 0x82, 0xFE, 0xFE, 0xFE, 0x02, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0x30, 0x0C, 0x3F, 0xF8, 0xE0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0xFF, 0xFF, 0xFF, 0x41,**

**0x41, 0xC1, 0xC3, 0x7E, 0x3E, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF8, 0xFC, 0x02, 0x01, 0x01, 0x01, 0x01, 0x81, 0x82, 0x8E, 0x80, 0x80, 0x00,**

**0x00, 0x00, 0x00, 0x02, 0x02, 0xFE, 0xFE, 0xFE, 0x02, 0x02, 0x00, 0x00, 0x00, 0x00, 0x02, 0x02, 0xFE, 0x06, 0x1E, 0x3C, 0x70, 0xE0, 0xC0, 0x02, 0x02,**

**0xFE, 0x02, 0x02, 0x00, 0x00, 0x00, 0x00, 0xC0, 0xF8, 0xFC, 0x02, 0x01, 0x01, 0x01, 0x01, 0x81, 0x82, 0x8E, 0x80, 0x80, 0x00, 0x00, 0x00, 0x07, 0x0F,**

**0x1C, 0x30, 0x20, 0x20, 0x20, 0x20, 0x20, 0x10, 0x1E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x20, 0x3F, 0x3F, 0x3F, 0x20, 0x00, 0x00, 0x00, 0x20, 0x3F, 0x3F,**

**0x3F, 0x20, 0x00, 0x00, 0x00, 0x20, 0x20, 0x38, 0x27, 0x22, 0x02, 0x02, 0x02, 0x23, 0x27, 0x3F, 0x3C, 0x30, 0x20, 0x20, 0x00, 0x00, 0x00, 0x20, 0x3F,**

**0x3F, 0x3F, 0x20, 0x00, 0x01, 0x07, 0x0F, 0x1C, 0x30, 0x20, 0x20, 0x00, 0x00, 0x00, 0x00, 0x07, 0x0F, 0x1C, 0x30, 0x20, 0x20, 0x20, 0x20, 0x1F, 0x1F,**

**0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x20, 0x20, 0x3F, 0x3F, 0x3F, 0x20, 0x20, 0x00, 0x00, 0x00, 0x00, 0x20, 0x20, 0x3F, 0x20, 0x20, 0x00, 0x00, 0x01,**

**0x03, 0x0F, 0x1E, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x0F, 0x1C, 0x30, 0x20, 0x20, 0x20, 0x20, 0x1F, 0x1F, 0x1F, 0x00, 0x00,**

**};**

**// 程序下载\_128\*48.png(即将复位进入下载模式);**

**const unsigned char DownloadMode\_IMG[] PROGMEM = {**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF8, 0xF8, 0xF8, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40,**

**0x40, 0x40, 0x40, 0x40, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0xC0, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0x60, 0x78, 0x7C, 0x7E, 0x7C,**

**0x78, 0x70, 0x70, 0x60, 0x40, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF8, 0xF8, 0xF8, 0xF8, 0xF8, 0xF8, 0x08, 0x08, 0x80,**

**0xE0, 0xF0, 0xC0, 0x80, 0x00, 0x00, 0xF8, 0xF8, 0xF8, 0xF8, 0xF8, 0xF8, 0x08, 0x08, 0x08, 0x18, 0x38, 0x78, 0xF8, 0xF0, 0xF0, 0xF0, 0xE0, 0x40, 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,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 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, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x40, 0x40, 0x41, 0x41, 0x41, 0x41, 0x41, 0x41, 0x41, 0xFF, 0xFF, 0x7F, 0x7F, 0x7F, 0x7F,**

**0x41, 0x41, 0x41, 0x41, 0x41, 0x41, 0x41, 0x41, 0x40, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x40, 0x40, 0x40, 0x40, 0x40, 0x60, 0x78, 0x7D, 0x7D, 0x79,**

**0x71, 0x60, 0x40, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x60, 0xF0, 0xF0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF,**

**0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xF0, 0xF0, 0x60, 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, 0x00, 0x01, 0x03, 0x07, 0x0F, 0x1F, 0x7F, 0xFE,**

**0xFE, 0xFE, 0xFC, 0xFC, 0xF8, 0xF0, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x20, 0x20, 0x20, 0x20, 0xE0, 0xE0, 0xF0, 0xFE, 0xFF, 0xFF, 0xFF,**

**0x3F, 0xAF, 0xA2, 0xA2, 0xA0, 0xB0, 0xBC, 0x3E, 0x38, 0x30, 0x30, 0x20, 0x07, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xE0, 0x00, 0x00, 0xC0, 0xFC, 0xFF, 0xFE,**

**0xFE, 0xFC, 0xFC, 0x38, 0x18, 0x10, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x0F, 0x1F, 0x3E, 0x7C, 0xF8, 0xF0, 0xE0,**

**0xC0, 0x80, 0xFF, 0xFF, 0xFF, 0x80, 0xC0, 0xE0, 0xF0, 0xF8, 0x7C, 0x3E, 0x1F, 0x0F, 0x07, 0x03, 0x01, 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, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x01, 0x03, 0x0F, 0x1F, 0x3F, 0x3F, 0x1F, 0x1F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10, 0x18, 0x3C, 0x3F, 0x7F, 0x7F, 0x3F, 0x1F,**

**0x1F, 0x13, 0xF0, 0xF0, 0xFF, 0xFF, 0xFF, 0xFF, 0x11, 0x18, 0x1E, 0x1F, 0x1C, 0x1C, 0x18, 0x10, 0x03, 0x3F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFC, 0xFE, 0xFF,**

**0xFF, 0x7F, 0x1F, 0x0F, 0x03, 0x00, 0x00, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x01, 0x03, 0x07, 0x0F, 0x1F, 0x1F, 0x1F, 0x0F, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 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, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x3E, 0xFE, 0x7E, 0x7E,**

**0x3E, 0x3E, 0x3E, 0x3E, 0x1E, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x06, 0x06, 0x02, 0x02, 0x03, 0x81, 0xC1, 0xE0, 0x70, 0x78, 0x3C, 0x1F, 0x0F, 0x1F,**

**0x3F, 0x7F, 0xFF, 0xFF, 0xFC, 0xF8, 0xF0, 0xF0, 0xF8, 0xFC, 0xFF, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E,**

**0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x0E, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x0F, 0x07, 0x07,**

**0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x0F, 0x07, 0x07, 0x00, 0x00, 0x04, 0x02, 0x03, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x03, 0x07, 0x07, 0x0F, 0x0F, 0x0F, 0x0F, 0x07, 0x00, 0x00, 0x00, /\*"程序下载.png",0\*/**

**};**

**// 状态栏图标;**

**const unsigned char StatusBars[][32] PROGMEM = {**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF0, 0xFE, 0xFE, 0xBE, 0x8E, 0x02, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x41, 0x79, 0x3F, 0x0F, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00,**

**}, // 正在充电图标-[0];**

**{**

**0x00, 0xCE, 0xCE, 0x8E, 0x8C, 0x1C, 0x1C, 0x38, 0x78, 0xF0, 0xE0, 0xC0, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x31, 0x79, 0x79, 0x33, 0x03, 0x0F, 0x7E, 0x78, 0x60, 0x01, 0x0F, 0x7F, 0x7E, 0x70, 0x00,**

**}, // WIFI连接-[1];**

**{**

**0x00, 0x00, 0xC0, 0x80, 0x00, 0x00, 0x00, 0xFE, 0xFE, 0x00, 0x00, 0x00, 0x80, 0xC0, 0x00, 0x00,**

**0x00, 0x00, 0x40, 0x41, 0x43, 0x46, 0x4C, 0x5F, 0x5F, 0x4C, 0x46, 0x43, 0x41, 0x40, 0x00, 0x00,**

**}, // 程序下载标志-[2];**

**{**

**0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0x60, 0x68, 0x6C, 0x2E, 0x0E, 0x86, 0xCE, 0xFC, 0x38, 0x00,**

**0x00, 0x1C, 0x3E, 0x73, 0x61, 0x70, 0x74, 0x36, 0x16, 0x06, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00,**

**}, // 断网\_任务栏图标-[3]**

**{**

**0x00, 0x00, 0x00, 0xFC, 0x04, 0xF6, 0xF6, 0xF6, 0xF6, 0xF6, 0xF6, 0x04, 0xFC, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x7F, 0x40, 0x5F, 0x5F, 0x5F, 0x5F, 0x5F, 0x5F, 0x40, 0x7F, 0x00, 0x00, 0x00,**

**}, // 电池供电\_任务栏图标-[4]**

**{**

**0x00, 0xFE, 0x12, 0x12, 0x92, 0x12, 0x12, 0x12, 0x12, 0x12, 0x12, 0x12, 0x12, 0x12, 0xFE, 0x00,**

**0x00, 0x7F, 0x40, 0x40, 0x48, 0x45, 0x42, 0x48, 0x48, 0x48, 0x40, 0x40, 0x40, 0x40, 0x7F, 0x00,**

**}, //"CMDCP\_任务栏图标-[5]"**

**{**

**0x00, 0xC0, 0xF0, 0xF8, 0xFC, 0xFC, 0xFE, 0xFE, 0xFE, 0xFE, 0xFC, 0xFC, 0xF8, 0xF0, 0xC0, 0x00,**

**0x00, 0x03, 0x0F, 0x1F, 0x3F, 0x3F, 0x7F, 0x7F, 0x7F, 0x7F, 0x3F, 0x3F, 0x1F, 0x0F, 0x03, 0x00, //"Sunny\_Day\_0.png",[6];**

**},**

**{**

**0x00, 0xE0, 0xF0, 0xFC, 0xFC, 0xFE, 0xC2, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x03, 0x0F, 0x1F, 0x3F, 0x3F, 0x7F, 0x7F, 0x7E, 0x7E, 0x7C, 0x3C, 0x1C, 0x1C, 0x06, 0x00, //"Clear\_Night\_1.png",[7];**

**},**

**{**

**0x00, 0xC0, 0xF0, 0xF8, 0xFC, 0xFC, 0xFE, 0xFE, 0xFE, 0xFE, 0xFC, 0xFC, 0xF8, 0xF0, 0xC0, 0x00,**

**0x00, 0x03, 0x0F, 0x1F, 0x3F, 0x3F, 0x7F, 0x7F, 0x7F, 0x7F, 0x3F, 0x3F, 0x1F, 0x0F, 0x03, 0x00, //"Sunny\_Day\_2.png",[8];**

**},**

**{**

**0x00, 0xE0, 0xF0, 0xFC, 0xFC, 0xFE, 0xC2, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x03, 0x0F, 0x1F, 0x3F, 0x3F, 0x7F, 0x7F, 0x7E, 0x7E, 0x7C, 0x3C, 0x1C, 0x1C, 0x06, 0x00, //"Clear\_Night\_3.png",[9];**

**},**

**{**

**0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xE0, 0xF0, 0xF8, 0x78, 0x78, 0x70, 0xC0, 0x80, 0x00, 0x00,**

**0x00, 0x10, 0x18, 0x13, 0x07, 0x07, 0x03, 0x0D, 0x1E, 0x1F, 0x1F, 0x1F, 0x1E, 0x1D, 0x0D, 0x00, //"Cloudy\_4.png",[10];**

**},**

**{**

**0x00, 0x00, 0x00, 0x80, 0xC0, 0xC0, 0xC0, 0xC0, 0xA0, 0xB0, 0x78, 0xF8, 0xF8, 0xF0, 0xE0, 0x00,**

**0x00, 0x0E, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x1E, 0x1E, 0x0D, 0x00, 0x00, //"Partly\_Cloudy\_Day\_5.png",[11];**

**},**

**{**

**0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xC0, 0xC0, 0xB8, 0x7C, 0xE0, 0xC0, 0x80, 0x80, 0x80, 0x00,**

**0x00, 0x18, 0x3C, 0x3F, 0x3F, 0x3F, 0x3F, 0x3F, 0x3F, 0x3F, 0x3E, 0x3E, 0x1D, 0x0D, 0x00, 0x00, //"Partly Cloudy\_6.png",[12];**

**},**

**{**

**0x00, 0x00, 0x00, 0x80, 0xC0, 0xC0, 0xC0, 0xC0, 0xA0, 0xB0, 0x78, 0xF8, 0xF8, 0xF0, 0xE0, 0x00,**

**0x00, 0x0E, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x1E, 0x1E, 0x0D, 0x00, 0x00, //"Mostly\_Cloudy\_Day\_7.png",[13];**

**},**

**{**

**0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xC0, 0xC0, 0xB8, 0x7C, 0xE0, 0xC0, 0x80, 0x80, 0x80, 0x00,**

**0x00, 0x18, 0x3C, 0x3F, 0x3F, 0x3F, 0x3F, 0x3F, 0x3F, 0x3F, 0x3E, 0x3E, 0x1D, 0x0D, 0x00, 0x00, //"Mostly\_Cloudy\_Night\_8.png",[14];**

**},**

**{**

**0x00, 0x00, 0x00, 0xC0, 0xE0, 0xE0, 0xE0, 0xF0, 0xF8, 0xF8, 0xF8, 0xF0, 0xE0, 0x80, 0x00, 0x00,**

**0x00, 0x06, 0x0F, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x1F, 0x0F, 0x07, 0x00, //"Overcast\_9.png",[15];**

**},**

**{**

**0x00, 0x00, 0x80, 0xE0, 0xF0, 0xF0, 0xF8, 0xF8, 0xF0, 0xEC, 0xDE, 0xBE, 0xBE, 0x7C, 0x38, 0x00,**

**0x00, 0x03, 0x07, 0x67, 0x37, 0x07, 0x67, 0x37, 0x07, 0x67, 0x37, 0x07, 0x07, 0x03, 0x00, 0x00, //"Shower\_10.png",[16];**

**},**

**{**

**0x00, 0xC0, 0xE0, 0xF8, 0xFC, 0xFC, 0xFC, 0xFC, 0xFE, 0xFE, 0xFE, 0xFC, 0xF8, 0xE0, 0xC0, 0x00,**

**0x00, 0x01, 0x03, 0x33, 0x1B, 0x03, 0x73, 0x3D, 0x0D, 0x03, 0x33, 0x1B, 0x03, 0x01, 0x00, 0x00, //"Thundershower\_11.png",[17];**

**},**

**{**

**0x00, 0xC0, 0xE0, 0xF8, 0xFC, 0xFC, 0xFC, 0xFC, 0xFE, 0xFE, 0xFE, 0xFC, 0xF8, 0xE0, 0xC0, 0x00,**

**0x00, 0x01, 0x1B, 0x1B, 0x03, 0x03, 0x73, 0x3D, 0x0D, 0x03, 0x03, 0x1B, 0x1B, 0x01, 0x00, 0x00, //"Thundershower\_with\_Hail\_12.png",[18];**

**},**

**{**

**0x00, 0xC0, 0xE0, 0xF8, 0xFC, 0xFC, 0xFC, 0xFC, 0xFE, 0xFE, 0xFE, 0xFC, 0xF8, 0xE0, 0xC0, 0x00,**

**0x00, 0x01, 0x63, 0x33, 0x03, 0x03, 0x63, 0x33, 0x03, 0x03, 0x63, 0x33, 0x03, 0x01, 0x00, 0x00, //"Light\_Rain\_13.png",[19];**

**},**

**{**

**0x00, 0xC0, 0xE0, 0xF8, 0xFC, 0xFC, 0xFC, 0xFC, 0xFE, 0xFE, 0xFE, 0xFC, 0xF8, 0xE0, 0xC0, 0x00,**

**0x00, 0x01, 0x63, 0x33, 0x03, 0x63, 0x33, 0x03, 0x63, 0x33, 0x03, 0x63, 0x33, 0x01, 0x00, 0x00, //"Moderate\_Rain\_14.png",[20];**

**},**

**{**

**0x00, 0xC0, 0xE0, 0xF8, 0xFC, 0xFC, 0xFC, 0xFC, 0xFE, 0xFE, 0xFE, 0xFC, 0xF8, 0xE0, 0xC0, 0x00,**

**0x00, 0x61, 0x33, 0x03, 0x63, 0x33, 0x03, 0x63, 0x33, 0x03, 0x63, 0x33, 0x03, 0x61, 0x30, 0x00, //"Heavy\_Rain\_15.png",[21];**

**},**

**{**

**0x00, 0xC0, 0xE0, 0xF8, 0xFC, 0xFC, 0xFC, 0xFC, 0xFE, 0xFE, 0xFE, 0xFC, 0xF8, 0xE0, 0xC0, 0x00,**

**0x00, 0x61, 0x1B, 0x03, 0x63, 0x1B, 0x03, 0x63, 0x1B, 0x03, 0x63, 0x1B, 0x03, 0x61, 0x18, 0x00, //"Storm\_16.png",[22];**

**},**

**{**

**0x00, 0xC0, 0xE0, 0xF8, 0xFC, 0xFC, 0xFC, 0xFC, 0xFE, 0xFE, 0xFE, 0xFC, 0xF8, 0xE0, 0xC0, 0x00,**

**0x20, 0x41, 0x03, 0x63, 0x1B, 0x63, 0x1B, 0x63, 0x1B, 0x63, 0x1B, 0x63, 0x1B, 0x41, 0x20, 0x00, //"Heavy\_Storm\_17.png",[23];**

**},**

**{**

**0x00, 0xC0, 0xE0, 0x38, 0x1C, 0x0C, 0x0C, 0x0C, 0x06, 0x06, 0x06, 0x1C, 0x38, 0xE0, 0xC0, 0x00,**

**0x20, 0x41, 0x03, 0x63, 0x1B, 0x63, 0x1B, 0x63, 0x1B, 0x63, 0x1B, 0x63, 0x1B, 0x41, 0x20, 0x00, //"Severe\_Storm\_18.png",[24];**

**},**

**{**

**0x00, 0xC0, 0xE0, 0xF8, 0xFC, 0xFC, 0xFC, 0xFC, 0xFE, 0xFE, 0xFE, 0xFC, 0xF8, 0xE0, 0xC0, 0x00,**

**0x00, 0x01, 0x03, 0x1B, 0x1B, 0x63, 0x63, 0x1B, 0x1B, 0x63, 0x63, 0x1B, 0x1B, 0x01, 0x00, 0x00, //"Ice\_Rain\_19.png",[25];**

**},**

**{**

**0x00, 0xC0, 0xE0, 0xF8, 0xFC, 0xFC, 0xFC, 0xFC, 0xFE, 0xFE, 0xFE, 0xFC, 0xF8, 0xE0, 0xC0, 0x00,**

**0x00, 0x01, 0x03, 0x63, 0x33, 0x03, 0x63, 0x33, 0x03, 0x03, 0x53, 0x23, 0x53, 0x01, 0x00, 0x00, //"Sleet\_20.png",[26];**

**},**

**{**

**0x00, 0x00, 0x80, 0xE0, 0xF0, 0xF0, 0xF8, 0xF8, 0xF0, 0xEC, 0xDE, 0xBE, 0xBE, 0x7C, 0x38, 0x00,**

**0x00, 0x03, 0x07, 0x07, 0x27, 0x57, 0x27, 0x07, 0x07, 0x27, 0x57, 0x27, 0x07, 0x03, 0x00, 0x00, //"Snow\_Flurry\_21.png",[27];**

**},**

**{**

**0x00, 0xC0, 0xE0, 0xF8, 0xFC, 0xFC, 0xFC, 0xFC, 0xFE, 0xFE, 0xFE, 0xFC, 0xF8, 0xE0, 0xC0, 0x00,**

**0x00, 0x01, 0x03, 0x03, 0x23, 0x53, 0x23, 0x03, 0x03, 0x23, 0x53, 0x23, 0x03, 0x01, 0x00, 0x00, //"Light\_Snow\_22.png",[28];**

**},**

**{**

**0x00, 0xC0, 0xE0, 0xF8, 0xFC, 0xFC, 0xFC, 0xFC, 0xFE, 0xFE, 0xFE, 0xFC, 0xF8, 0xE0, 0xC0, 0x00,**

**0x00, 0x01, 0x23, 0x53, 0x23, 0x03, 0x53, 0x23, 0x53, 0x03, 0x23, 0x53, 0x23, 0x01, 0x00, 0x00, //"Moderate\_Snow\_23.png",[29];**

**},**

**{**

**0x00, 0xC0, 0xE0, 0xF8, 0xFC, 0xFC, 0xFC, 0xFC, 0xFE, 0xFE, 0xFE, 0xFC, 0xF8, 0xE0, 0xC0, 0x00,**

**0x20, 0x51, 0x23, 0x03, 0x53, 0x23, 0x53, 0x03, 0x23, 0x53, 0x23, 0x03, 0x53, 0x21, 0x50, 0x00, //"Heavy\_Snow\_24.png",[30];**

**},**

**{**

**0x00, 0xC0, 0xE0, 0x38, 0x1C, 0x0C, 0x0C, 0x0C, 0x06, 0x06, 0x06, 0x1C, 0x38, 0xE0, 0xC0, 0x00,**

**0x20, 0x51, 0x23, 0x03, 0x53, 0x23, 0x53, 0x03, 0x23, 0x53, 0x23, 0x03, 0x53, 0x21, 0x50, 0x00, //"Snowstorm\_25.png",[31];**

**},**

**{**

**0x00, 0x40, 0x40, 0x10, 0x50, 0x40, 0x40, 0x10, 0x50, 0x40, 0x10, 0x50, 0x40, 0x10, 0x10, 0x00,**

**0x00, 0x44, 0x44, 0x11, 0x55, 0x44, 0x51, 0x15, 0x55, 0x44, 0x11, 0x55, 0x44, 0x11, 0x11, 0x00, //"Dust\_26.png",[32];**

**},**

**{**

**0x00, 0x44, 0x44, 0x10, 0x54, 0x54, 0x40, 0x10, 0x54, 0x44, 0x10, 0x54, 0x44, 0x10, 0x10, 0x00,**

**0x00, 0x44, 0x44, 0x11, 0x55, 0x44, 0x51, 0x15, 0x55, 0x44, 0x11, 0x55, 0x44, 0x11, 0x11, 0x00, //"Sand\_27.png",[33];**

**},**

**{**

**0x00, 0x40, 0x00, 0x40, 0x08, 0x40, 0x48, 0x40, 0x48, 0x40, 0x50, 0x48, 0x48, 0x30, 0x00, 0x00,**

**0x00, 0x00, 0x02, 0x00, 0x02, 0x10, 0x02, 0x10, 0x02, 0x12, 0x02, 0x0A, 0x12, 0x12, 0x0C, 0x00, //"Duststorm\_28.png",[34];**

**},**

**{**

**0x00, 0x40, 0x40, 0x48, 0x08, 0x40, 0x48, 0x40, 0x48, 0x40, 0x50, 0x48, 0x48, 0x30, 0x00, 0x00,**

**0x00, 0x00, 0x02, 0x02, 0x12, 0x10, 0x02, 0x10, 0x02, 0x12, 0x02, 0x0A, 0x12, 0x12, 0x0C, 0x00, //"Sandstorm\_29.png",[35];**

**},**

**{**

**0x00, 0x20, 0x38, 0x3C, 0x3C, 0x3E, 0x3E, 0x3E, 0x3C, 0x38, 0x30, 0x30, 0x30, 0x20, 0x20, 0x00,**

**0x00, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0x00, //"Foggy\_30.png",[36];**

**},**

**{**

**0x00, 0x8C, 0x4C, 0x20, 0x20, 0x20, 0x40, 0x8C, 0x8C, 0x40, 0x20, 0x20, 0x20, 0x4C, 0x8C, 0x00,**

**0x00, 0x63, 0x64, 0x08, 0x08, 0x08, 0x04, 0x63, 0x63, 0x04, 0x08, 0x08, 0x08, 0x64, 0x63, 0x00, //"Haze\_31.png",[37];**

**},**

**{**

**0x00, 0x90, 0x90, 0x90, 0x90, 0x90, 0x90, 0x94, 0x88, 0x80, 0x80, 0xA0, 0x90, 0x90, 0x60, 0x00,**

**0x00, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0x14, 0x24, 0x24, 0x18, 0x00, 0x00, //"Windy\_32.png",[38];**

**},**

**{**

**0x00, 0x90, 0x90, 0x90, 0x90, 0x94, 0x92, 0x92, 0x8C, 0x80, 0x88, 0x84, 0x84, 0x48, 0x30, 0x00,**

**0x00, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0x04, 0x24, 0x44, 0x44, 0x44, 0x38, 0x00, 0x00, //"Blustery\_33.png",[39];**

**},**

**{**

**0x00, 0x04, 0x44, 0x54, 0x54, 0x54, 0x54, 0x54, 0x54, 0x54, 0x54, 0x54, 0x14, 0x14, 0x14, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x05, 0x55, 0x55, 0x15, 0x15, 0x05, 0x05, 0x01, 0x01, 0x00, //"Hurricane\_34.png",[40];**

**},**

**{**

**0x00, 0x04, 0x44, 0x54, 0x54, 0x54, 0x50, 0x14, 0x54, 0x54, 0x10, 0x54, 0x14, 0x10, 0x14, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x05, 0x55, 0x51, 0x15, 0x15, 0x01, 0x05, 0x01, 0x01, 0x00, //"Tropical\_Storm\_35.png",[41];**

**},**

**{**

**0x00, 0x18, 0xA4, 0x24, 0x52, 0x4A, 0x4A, 0x4A, 0x4A, 0x4A, 0x4A, 0x52, 0x24, 0xA4, 0x18, 0x00,**

**0x00, 0x00, 0x00, 0x01, 0x05, 0x09, 0x2A, 0x4A, 0x4A, 0x2A, 0x09, 0x05, 0x01, 0x00, 0x00, 0x00, //"Tornado\_36.png",[42];**

**},**

**{**

**0x00, 0x00, 0x50, 0x20, 0x30, 0x44, 0x88, 0x54, 0xB2, 0x88, 0x44, 0x30, 0x20, 0x50, 0x00, 0x00,**

**0x00, 0x00, 0x0A, 0x04, 0x0C, 0x22, 0x11, 0x4D, 0x2A, 0x11, 0x22, 0x0C, 0x04, 0x0A, 0x00, 0x00, //"Cold\_37.png",[43];**

**},**

**{**

**0x00, 0x00, 0x80, 0x08, 0x90, 0xC0, 0xE0, 0xF2, 0xF4, 0xE0, 0xC0, 0x90, 0x08, 0x00, 0x80, 0x00,**

**0x00, 0x01, 0x00, 0x10, 0x09, 0x03, 0x07, 0x2F, 0x4F, 0x07, 0x03, 0x09, 0x10, 0x01, 0x00, 0x00, //"Hot\_38.png",[44];**

**},**

**{**

**0x00, 0x02, 0x02, 0xFE, 0x0C, 0x18, 0x20, 0xC0, 0x80, 0x00, 0x02, 0x02, 0xFE, 0x02, 0x02, 0x00,**

**0x00, 0x40, 0x40, 0x7F, 0x40, 0x40, 0x40, 0x00, 0x01, 0x06, 0x0C, 0x10, 0x7F, 0x00, 0x00, 0x00, //"Unknown\_99.png",[45];**

**},**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**}, // NULL空(用于清空任务栏图标)-[46];**

**};**

**// Loading动画(30FPS);**

**const unsigned char Loading\_X16\_30F[31][32] PROGMEM = {**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[4];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x11, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[5];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x21, 0x71, 0x21, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[6];**

**{**

**0xff, 0x01, 0x01, 0x61, 0xf1, 0xf9, 0xf9, 0xf1, 0x61, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x81, 0x81, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[7];**

**{**

**0xff, 0x01, 0x01, 0xe1, 0xf1, 0xf9, 0xf9, 0xf9, 0xf1, 0xe1, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x81, 0x83, 0x83, 0x83, 0x81, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[8];**

**{**

**0xff, 0x01, 0x01, 0xf1, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf1, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x87, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x87, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[9];**

**{**

**0xff, 0x01, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xf9, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x8f, 0x80, 0x80, 0xff,**

**}, // Loading-[10];**

**{**

**0xff, 0x01, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[11];**

**{**

**0xff, 0x01, 0x01, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x80, 0x80, 0xff,**

**}, // Loading-[12];**

**{**

**0xff, 0x01, 0x01, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x80, 0x80, 0xff,**

**}, // Loading-[13];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0xe1, 0xe1, 0xe1, 0xe1, 0xe1, 0xe1, 0xe1, 0xe1, 0xe1, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0x80, 0xff,**

**}, // Loading-[14];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0xc1, 0xc1, 0xc1, 0xc1, 0xc1, 0xc1, 0xc1, 0xc1, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0x80, 0xff,**

**}, // Loading-[15];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0x80, 0xff,**

**}, // Loading-[16];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xbc, 0xbc, 0xbc, 0xbc, 0x80, 0xff,**

**}, // Loading-[17];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xa0, 0x80, 0xff,**

**}, // Loading-[18];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[19];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[20];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[21];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xb0, 0xb8, 0xb8, 0x80, 0xff,**

**}, // Loading-[22];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xbe, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0x80, 0xff,**

**}, // Loading-[23];**

**{**

**0xff, 0x01, 0x01, 0x01, 0xc1, 0xe1, 0xf1, 0xf1, 0xf1, 0xf1, 0xf1, 0xf1, 0xe1, 0xc1, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x8f, 0x9f, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0x9f, 0x8f, 0x80, 0xff,**

**}, // Loading-[24];**

**{**

**0xff, 0x01, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xf9, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x83, 0x87, 0x87, 0x87, 0x87, 0x87, 0x83, 0x83, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[25];**

**{**

**0xff, 0x01, 0xfd, 0xfd, 0xfd, 0xfd, 0x7d, 0x7d, 0x09, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x81, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[26];**

**{**

**0xff, 0x01, 0x7d, 0x3d, 0x3d, 0x1d, 0x05, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[27];**

**{**

**0xff, 0x01, 0x0d, 0x05, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[28];**

**{**

**0xff, 0x01, 0x0d, 0x05, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[29];**

**{**

**0xff, 0x01, 0x0d, 0x0d, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[30];**

**{**

**0xff, 0x01, 0x1d, 0x0d, 0x05, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[31];**

**{**

**0xff, 0x01, 0x1d, 0x1d, 0x0d, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[32];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[33];**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**}, // Clear\_16x16;**

**};**

**// Loading动画(60FPS);**

**const unsigned char Loading\_X16\_60F[61][32] PROGMEM = {**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[11];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x21, 0x71, 0x21, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[12];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x61, 0xf1, 0xf1, 0x61, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[13];**

**{**

**0xff, 0x01, 0x01, 0x61, 0xf1, 0xf9, 0xf9, 0xf1, 0x61, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x81, 0x81, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[14];**

**{**

**0xff, 0x01, 0x01, 0x61, 0xf1, 0xf9, 0xf9, 0xf1, 0x61, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x81, 0x81, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[15];**

**{**

**0xff, 0x01, 0x01, 0xe1, 0xf1, 0xf9, 0xf9, 0xf9, 0xf1, 0xe1, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x81, 0x83, 0x83, 0x83, 0x81, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[16];**

**{**

**0xff, 0x01, 0x01, 0xe1, 0xf1, 0xf9, 0xf9, 0xf9, 0xf9, 0xf1, 0xe1, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x81, 0x83, 0x87, 0x87, 0x87, 0x87, 0x83, 0x81, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[17];**

**{**

**0xff, 0x01, 0x01, 0xf1, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf1, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x83, 0x87, 0x87, 0x87, 0x87, 0x87, 0x87, 0x83, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[18];**

**{**

**0xff, 0x01, 0x01, 0xf1, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf1, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x87, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x87, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[19];**

**{**

**0xff, 0x01, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xf9, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x87, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[20];**

**{**

**0xff, 0x01, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xf9, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x8f, 0x80, 0x80, 0xff,**

**}, // Loading-[21];**

**{**

**0xff, 0x01, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[22];**

**{**

**0xff, 0x01, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[23];**

**{**

**0xff, 0x01, 0x01, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x80, 0x80, 0xff,**

**}, // Loading-[24];**

**{**

**0xff, 0x01, 0x01, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x80, 0x80, 0xff,**

**}, // Loading-[25];**

**{**

**0xff, 0x01, 0x01, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0xf9, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x80, 0x80, 0xff,**

**}, // Loading-[26];**

**{**

**0xff, 0x01, 0x01, 0x01, 0xf1, 0xf1, 0xf1, 0xf1, 0xf1, 0xf1, 0xf1, 0xf1, 0xf1, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x9f, 0x80, 0x80, 0xff,**

**}, // Loading-[27];**

**{**

**0xff, 0x01, 0x01, 0x01, 0xf1, 0xf1, 0xf1, 0xf1, 0xf1, 0xf1, 0xf1, 0xf1, 0xf1, 0xf1, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0x80, 0xff,**

**}, // Loading-[28];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0xe1, 0xe1, 0xe1, 0xe1, 0xe1, 0xe1, 0xe1, 0xe1, 0xe1, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0x80, 0xff,**

**}, // Loading-[29];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0xe1, 0xe1, 0xe1, 0xe1, 0xe1, 0xe1, 0xe1, 0xe1, 0xe1, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0x80, 0xff,**

**}, // Loading-[30];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0xc1, 0xc1, 0xc1, 0xc1, 0xc1, 0xc1, 0xc1, 0xc1, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0x80, 0xff,**

**}, // Loading-[31];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0x80, 0xff,**

**}, // Loading-[32];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0x80, 0xff,**

**}, // Loading-[33];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xbe, 0xbe, 0xbe, 0xbe, 0xbe, 0x80, 0xff,**

**}, // Loading-[34];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xbc, 0xbc, 0xbc, 0xbc, 0x80, 0xff,**

**}, // Loading-[35];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xb0, 0xb0, 0x80, 0xff,**

**}, // Loading-[36];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xa0, 0x80, 0xff,**

**}, // Loading-[37];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[38];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[39];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[40];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[41];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[42];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[43];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xa0, 0xb0, 0x80, 0xff,**

**}, // Loading-[44];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xb0, 0xb8, 0xb8, 0x80, 0xff,**

**}, // Loading-[45];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xb8, 0xbc, 0xbc, 0xbc, 0x80, 0xff,**

**}, // Loading-[46];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xbe, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0x80, 0xff,**

**}, // Loading-[47];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x81, 0xc1, 0xc1, 0xc1, 0xc1, 0xc1, 0x81, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x9f, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0x80, 0xff,**

**}, // Loading-[48];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x81, 0xc1, 0xe1, 0xe1, 0xe1, 0xe1, 0xe1, 0xc1, 0x81, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x8f, 0x9f, 0xbf, 0xbf, 0xbf, 0xbf, 0xbf, 0x9f, 0x8f, 0x80, 0xff,**

**}, // Loading-[49];**

**{**

**0xff, 0x01, 0x01, 0xc1, 0xf1, 0xf1, 0xf9, 0xf9, 0xf9, 0xf9, 0xf1, 0xf1, 0xc1, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x83, 0x8f, 0x8f, 0x9f, 0x9f, 0x9f, 0x9f, 0x8f, 0x8f, 0x83, 0x80, 0x80, 0xff,**

**}, // Loading-[50];**

**{**

**0xff, 0x01, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xf1, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x87, 0x87, 0x8f, 0x8f, 0x8f, 0x8f, 0x8f, 0x87, 0x87, 0x81, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[51];**

**{**

**0xff, 0x01, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0x31, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x81, 0x81, 0x83, 0x83, 0x81, 0x81, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[52];**

**{**

**0xff, 0x01, 0xfd, 0xfd, 0xfd, 0xfd, 0xfd, 0x7d, 0x19, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x81, 0x81, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[53];**

**{**

**0xff, 0x01, 0xfd, 0xfd, 0x7d, 0x7d, 0x3d, 0x0d, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[54];**

**{**

**0xff, 0x01, 0x7d, 0x3d, 0x3d, 0x1d, 0x05, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[55];**

**{**

**0xff, 0x01, 0x1d, 0x1d, 0x0d, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[56];**

**{**

**0xff, 0x01, 0x1d, 0x1d, 0x0d, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[57];**

**{**

**0xff, 0x01, 0x0d, 0x05, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[58];**

**{**

**0xff, 0x01, 0x0d, 0x05, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[59];**

**{**

**0xff, 0x01, 0x0d, 0x0d, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[60];**

**{**

**0xff, 0x01, 0x0d, 0x0d, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[61];**

**{**

**0xff, 0x01, 0x0d, 0x0d, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[62];**

**{**

**0xff, 0x01, 0x1d, 0x0d, 0x05, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[63];**

**{**

**0xff, 0x01, 0x1d, 0x0d, 0x05, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[64];**

**{**

**0xff, 0x01, 0x1d, 0x0d, 0x05, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[65];**

**{**

**0xff, 0x01, 0x1d, 0x1d, 0x0d, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[66];**

**{**

**0xff, 0x01, 0x09, 0x1d, 0x09, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[67];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[68];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x11, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[69];**

**{**

**0xff, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xff,**

**0xff, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff,**

**}, // Loading-[70];**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**}, // Clear\_16x16;**

**};**

**// LoadingBar往复进度条动画(60FPS);**

**const unsigned char LoadingBackForthBar\_60x8\_60F[61][60] PROGMEM = {**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[4];**

**{**

**0xff, 0x81, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[5];**

**{**

**0xff, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[6];**

**{**

**0xff, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[7];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[8];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[9];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[10];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[11];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[12];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[13];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[14];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[15];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[16];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[17];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[18];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[19];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[20];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[21];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[22];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[23];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[24];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[25];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[26];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[27];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[28];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[29];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[30];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[31];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[32];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[33];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[34];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[35];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[36];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[37];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0x81, 0xff,**

**}, // Loading-[38];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0x81, 0xff,**

**}, // Loading-[39];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[40];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[41];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[42];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[43];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[44];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[45];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[46];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[47];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[48];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[49];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[50];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[51];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[52];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[53];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[54];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[55];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[56];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[57];**

**{**

**0xff, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[58];**

**{**

**0xff, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[59];**

**{**

**0xff, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[60];**

**{**

**0xff, 0x81, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[61];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[62];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[63];**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**}, // Clear\_60x8;**

**};**

**// LoadingBar进度条动画(30FPS);**

**const unsigned char LoadingBar\_60x8\_30F[31][60] PROGMEM = {**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[3];**

**{**

**0xff, 0x81, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[4];**

**{**

**0xff, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[5];**

**{**

**0xff, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[6];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[7];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[8];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[9];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[10];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[11];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[12];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[13];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[14];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[15];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[16];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[17];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[18];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[19];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[20];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[21];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[22];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[23];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[24];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[25];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[26];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[27];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[28];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[29];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0x81, 0xff,**

**}, // Loading-[30];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[31];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[32];**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**}, // Clear\_60x8;**

**};**

**// LoadingBar进度条动画(60FPS);**

**const unsigned char LoadingBar\_60x8\_60F[61][60] PROGMEM = {**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[4];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[5];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[6];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[7];**

**{**

**0xff, 0x81, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[8];**

**{**

**0xff, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[9];**

**{**

**0xff, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[10];**

**{**

**0xff, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[11];**

**{**

**0xff, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[12];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[13];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[14];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[15];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[16];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[17];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[18];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[19];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[20];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[21];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[22];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[23];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[24];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[25];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[26];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[27];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[28];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[29];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[30];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[31];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[32];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[33];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[34];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[35];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[36];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[37];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[38];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[39];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[40];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[41];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[42];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[43];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[44];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[45];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[46];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[47];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[48];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[49];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[50];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[51];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[52];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[53];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[54];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[55];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[56];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[57];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[58];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0xbd, 0x81, 0xff,**

**}, // Loading-[59];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0x81, 0xff,**

**}, // Loading-[60];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xbd, 0x81, 0xff,**

**}, // Loading-[61];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[62];**

**{**

**0xff, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81,**

**0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0x81, 0xff,**

**}, // Loading-[63];**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**}, // Clear\_60x8;**

**};**

**#endif**

**// OLED.h-----------------------------------------------------------------------------------------------------------**

**#pragma once**

**#include "Universal.h"**

**using namespace std;**

**// 输出电平的宏定义**

**#define I2C\_SCL\_L digitalWrite(SCL, LOW)  // SCL**

**#define I2C\_SCL\_H digitalWrite(SCL, HIGH)**

**#define I2C\_SDA\_L digitalWrite(SDA, LOW)  // SDA**

**#define I2C\_SDA\_H digitalWrite(SDA, HIGH)**

**// 管脚模式的宏定义**

**#define SCL\_IN pinMode(SCL, INPUT)**

**#define SCL\_OUT pinMode(SCL, OUTPUT)  // SCL**

**#define SDA\_IN pinMode(SDA, INPUT)**

**#define SDA\_OUT pinMode(SDA, OUTPUT)  // SDA**

**#define OLED\_CMD 0   // 写命令**

**#define OLED\_DATA 1  // 写数据**

**// 坐标点类型定义;**

**typedef struct POINT {**

**unsigned char x;**

**unsigned char y;**

**} POINT;**

**// 矩形区域类型定义;**

**typedef struct RECT {**

**unsigned char left;**

**unsigned char top;**

**unsigned char right;**

**unsigned char bottom;**

**} RECT;**

**class OLED {**

**private:**

**POINT OLED\_Pos = {0, 0};  // OLED像素指针位置;**

**unsigned int First\_Line = 0;                                  // 文本打印区域的首行在PrintBox中的位置;**

**RECT PrintRECT = {0, 0, 128, 64};                             // 文本打印区域;**

**unsigned char Width\_MaxNumChar, Hight\_MaxNumChar;             // 一行最多能显示多少字符, 一列最多能显示多少字符;**

**unsigned char sliderHeight;                                   // 滚动条高度;**

**unsigned char sliderPos[2] = {PrintRECT.top, PrintRECT.top};  // 滚动位置{当前位置, 上一个位置};**

**vector<String> PrintBox;                                      // 储存打印字符串的容器;**

**// "OLED\_GDDRAM\_CLONE"是"SSD1306 GDDRAM"的克隆数据(这里的数据总与OLED上显示的内容保持一致);**

**// 由于此OLED屏幕不能按位更新, 因此待显示字节通过与这里的数据进行位运算实现显示屏位更新;**

**unsigned char OLED\_GDDRAM\_CLONE[8][128] = {**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**},  // Page0;**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**},  // Page1;**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**},  // Page2;**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**},  // Page3;**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**},  // Page4;**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**},  // Page5;**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**},  // Page6;**

**{**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,**

**},  // Page7;**

**};**

**// 预渲染开关(true:启动预渲染待显示内容将会先缓存到OLED\_GDDRAM\_CLONE中,当设为false时将会把OLED\_GDDRAM\_CLONE的内容显示到屏幕上);**

**bool PreRendered = false;**

**// 按字节更新OLED\_GDDRAM\_CLONE中的信息;**

**void SET\_OLED\_GDDRAM\_CLONE(unsigned char BytesData);**

**// 渲染文本箱;**

**void drawPrintBox();**

**public:**

**// 字符串分割, 将一个给定的字符串 "input" 根据另一个字符串 "separator" 进行分割，并将分割后的每一段字符串存储在一个 vector 容器中。**

**vector<String> strsplit(String input, String separator);**

**// 将OLED\_GDDRAM\_CLONE中的内容刷新到屏幕上(可以指定刷新区域, 默认全屏刷新);**

**void OLED\_GDDRAM\_Refresh(u8 leftPixel = 0, u8 topPixel = 0, u8 rightPixel = 128, u8 bottomPixel = 64);**

**// 开始进行批量绘制(开始预渲染);**

**void BeginBatchDraw();**

**// 结束进行批量绘制(结束预渲染, 不会自动刷新预渲染缓存的内容);**

**void EndBatchDraw();**

**// 这个函数用于画点(点的坐标, enabl:true在屏幕上画一点, enabl:false清除屏幕上的一点);**

**void putpixel(u8 xPixel, u8 yPixel, bool enabl = true);**

**// 这个函数用于画直线;**

**void line(u8 x1Pixel, u8 y1Pixel, u8 x2Pixel, u8 y2Pixel, bool enable = true);**

**// 这个函数用于画无填充的矩形;**

**void rectangle(u8 leftPixel, u8 topPixel, u8 rightPixel, u8 bottomPixel);**

**// 这个函数用于画有边框的填充矩形;**

**void fillrectangle(u8 leftPixel, u8 topPixel, u8 rightPixel, u8 bottomPixel);**

**// 这个函数用于清空矩形区域;**

**void clearrectangle(u8 leftPixel, u8 topPixel, u8 rightPixel, u8 bottomPixel);**

**// 设置文本框(文本框左部 x 坐标, 文本框顶部 y 坐标, 文本框右部 x 坐标,文本框底部 y 坐标);**

**void setTextBox(u8 leftPixel, u8 topPixel, u8 rightPixel, u8 bottomPixel);**

**// 打印字符串(字符串, 自动滚动[设为true时会随着打印内容而自动向下滚动, 默认启用]);**

**void print(String text, bool autoScroll = true);**

**// 清空文本框并释放内存;**

**void clearTextBox();**

**// 获取文本框的全部文本;**

**vector<String> getPrintBox();**

**// 用一个新的PrintBox结构的数据替换掉现有的PrintBox(要求保证数据结构正确);**

**void replacePrintBox(vector<String> newPrintBox);**

**// 移动滚动条一行(true:向下滚动, falas:向上滚动);**

**void moveScrollBar(bool direction = true);**

**private:**

**// I2C初始化**

**void I2C\_Init();**

**// I2C开始**

**void I2C\_Start();**

**// I2C结束**

**void I2C\_Stop();**

**// 检查应答信号**

**void I2C\_Wait\_Ack();**

**// I2C写入一个字节**

**void Write\_I2C\_Byte(unsigned char dat);**

**// 发送一个字节**

**// 向SSD1306写入一个字节。**

**// mode:数据/命令标志 0,表示命令;1,表示数据;**

**void OLED\_WR\_Byte(u8 dat, u8 mode);**

**// 坐标设置**

**void OLED\_Set\_Pos(u8 x, u8 y);**

**// m^n函数**

**u32 oled\_pow(u8 m, u8 n);**

**public:**

**// 反显函数**

**void OLED\_ColorTurn(u8 i);**

**// 屏幕旋转180度**

**void OLED\_DisplayTurn(u8 i);**

**// 开启OLED显示**

**void OLED\_Display\_On(void);**

**// 关闭OLED显示**

**void OLED\_Display\_Off(void);**

**// 清屏函数;**

**void OLED\_Clear(void);**

**// 在指定位置显示一个字符**

**// x:0~127**

**// y:0~7**

**// sizey:选择字体 6x8  8x16**

**void OLED\_ShowChar(u8 x, u8 y, const u8 chr, u8 sizey);**

**// 显示数字**

**// x,y :起点坐标**

**// num:要显示的数字**

**// len :数字的位数**

**// sizey:字体大小**

**void OLED\_ShowNum(u8 x, u8 y, u32 num, u8 len, u8 sizey);**

**// 显示一个字符串**

**void OLED\_ShowString(u8 x, u8 y, const char \*chr, u8 sizey);**

**// 显示汉字**

**void OLED\_ShowChinese(u8 x, u8 y, const u8 no, u8 sizey);**

**// 显示图片**

**// x,y显示坐标**

**// sizex,sizey,图片长宽**

**// BMP：要显示的图片**

**void OLED\_DrawBMP(u8 x, u8 y, u8 sizex, u8 sizey, const u8 BMP[]);**

**// OLED的初始化**

**void OLED\_Init(void);**

**};**

**// OLED.cpp-----------------------------------------------------------------------------------------------------------**

**#include "OLED.h"**

**// I2C初始化**

**void OLED::I2C\_Init() {**

**SDA\_OUT;**

**SCL\_OUT;**

**I2C\_SDA\_H;**

**I2C\_SCL\_H;**

**}**

**// I2C开始**

**void OLED::I2C\_Start() {**

**I2C\_SCL\_H;**

**I2C\_SDA\_H;**

**I2C\_SDA\_L;**

**I2C\_SCL\_L;**

**}**

**// I2C结束**

**void OLED::I2C\_Stop() {**

**I2C\_SCL\_H;**

**I2C\_SDA\_L;**

**I2C\_SDA\_H;**

**}**

**// 检查应答信号**

**void OLED::I2C\_Wait\_Ack() {**

**I2C\_SDA\_H;**

**I2C\_SCL\_H;**

**I2C\_SCL\_L;**

**}**

**// I2C写入一个字节**

**void OLED::Write\_I2C\_Byte(unsigned char dat) {**

**unsigned char i;**

**for (i = 0; i < 8; i++) {**

**I2C\_SCL\_L;**

**if (dat & 0x80) {**

**I2C\_SDA\_H;**

**} else {**

**I2C\_SDA\_L;**

**}**

**dat <<= 1;**

**I2C\_SCL\_H;**

**}**

**I2C\_SCL\_L;**

**}**

**// 发送一个字节**

**// 向SSD1306写入一个字节。**

**// mode:数据/命令标志 0,表示命令;1,表示数据;**

**void OLED::OLED\_WR\_Byte(u8 dat, u8 mode) {**

**// 如果开启预渲染则图形数据将先缓存在OLED\_GDDRAM\_CLONE中, 不会立刻显示到屏幕上;**

**if (PreRendered == true && mode == OLED\_DATA) {**

**SET\_OLED\_GDDRAM\_CLONE(dat);  // 更新OLED\_GDDRAM\_CLONE中的数据;**

**} else if (PreRendered == false) {**

**I2C\_Start();**

**Write\_I2C\_Byte(0x78);**

**I2C\_Wait\_Ack();**

**if (mode == OLED\_DATA) {**

**SET\_OLED\_GDDRAM\_CLONE(dat);  // 更新OLED\_GDDRAM\_CLONE中的数据;**

**Write\_I2C\_Byte(0x40);**

**} else {**

**Write\_I2C\_Byte(0x00);**

**}**

**I2C\_Wait\_Ack();**

**Write\_I2C\_Byte(dat);**

**I2C\_Wait\_Ack();**

**I2C\_Stop();**

**}**

**}**

**// data: 为OLED某页中的某一列8位数据(1字节);**

**void OLED::SET\_OLED\_GDDRAM\_CLONE(unsigned char BytesData) {**

**OLED\_GDDRAM\_CLONE[OLED\_Pos.y][OLED\_Pos.x] = BytesData;  // [当前点在哪页][当前点的列位置]; 更新OLED\_GDDRAM\_CLONE中的数据;**

**}**

**// 反显函数**

**void OLED::OLED\_ColorTurn(u8 i) {**

**if (!i)**

**OLED\_WR\_Byte(0xA6, OLED\_CMD);  // 正常显示**

**else**

**OLED\_WR\_Byte(0xA7, OLED\_CMD);  // 反色显示**

**}**

**// 屏幕旋转180度**

**void OLED::OLED\_DisplayTurn(u8 i) {**

**if (i == 0) {**

**OLED\_WR\_Byte(0xC8, OLED\_CMD);  // 正常显示**

**OLED\_WR\_Byte(0xA1, OLED\_CMD);**

**} else {**

**OLED\_WR\_Byte(0xC0, OLED\_CMD);  // 反转显示**

**OLED\_WR\_Byte(0xA0, OLED\_CMD);**

**}**

**}**

**// 坐标设置**

**void OLED::OLED\_Set\_Pos(u8 x, u8 y) {**

**OLED\_Pos = {x, y};**

**OLED\_WR\_Byte(0xb0 + y, OLED\_CMD);**

**OLED\_WR\_Byte(((x & 0xf0) >> 4) | 0x10, OLED\_CMD);**

**OLED\_WR\_Byte((x & 0x0f), OLED\_CMD);**

**}**

**// 开启OLED显示**

**void OLED::OLED\_Display\_On(void) {**

**OLED\_WR\_Byte(0X8D, OLED\_CMD);  // SET DCDC命令**

**OLED\_WR\_Byte(0X14, OLED\_CMD);  // DCDC ON**

**OLED\_WR\_Byte(0XAF, OLED\_CMD);  // DISPLAY ON**

**}**

**// 关闭OLED显示**

**void OLED::OLED\_Display\_Off(void) {**

**OLED\_WR\_Byte(0X8D, OLED\_CMD);  // SET DCDC命令**

**OLED\_WR\_Byte(0X10, OLED\_CMD);  // DCDC OFF**

**OLED\_WR\_Byte(0XAE, OLED\_CMD);  // DISPLAY OFF**

**}**

**// 清屏函数;**

**void OLED::OLED\_Clear(void) {**

**for (u8 i = 0; i < 8; ++i) {**

**OLED\_WR\_Byte(0xb0 + i, OLED\_CMD);  // 设置页地址（0~7）**

**OLED\_WR\_Byte(0x00, OLED\_CMD);      // 设置显示位置—列低地址**

**OLED\_WR\_Byte(0x10, OLED\_CMD);      // 设置显示位置—列高地址**

**for (u8 n = 0; n < 128; ++n) OLED\_WR\_Byte(0, OLED\_DATA);**

**}  // 更新显示**

**}**

**// 在指定位置显示一个字符**

**// x:0~127**

**// y:0~7**

**// sizey:选择字体 6x8  8x16 16x32**

**void OLED::OLED\_ShowChar(u8 x, u8 y, const u8 chr, u8 sizey) {**

**u8 c = 0, sizex = sizey / 2, temp;**

**u16 i = 0, size1;**

**if (sizey == 8)**

**size1 = 5;**

**else**

**size1 = (sizey / 8 + ((sizey % 8) ? 1 : 0)) \* (sizey / 2);**

**c = chr - ' ';  // 得到偏移后的值**

**OLED\_Set\_Pos(x, y);**

**for (i = 0; i < size1; i++) {**

**if (i % sizex == 0 && sizey != 8) OLED\_Set\_Pos(x, y++);**

**if (sizey == 8) {**

**temp = pgm\_read\_byte(&GSOS\_ASCII\_0805[c][i]);**

**OLED\_WR\_Byte(temp, OLED\_DATA);  // 5\*8 GSOS\_ASCII字符集点阵字体;**

**} else if (sizey == 16) {**

**temp = pgm\_read\_byte(&asc2\_1608[c][i]);**

**OLED\_WR\_Byte(temp, OLED\_DATA);  // 8x16字号**

**} else if (sizey == 32) {**

**temp = pgm\_read\_byte(&asc2\_3216[c][i]);**

**OLED\_WR\_Byte(temp, OLED\_DATA);  // 16x32字号(Consolas)**

**} else if (sizey == 64) {**

**temp = pgm\_read\_byte(&asc2\_6432[c][i]);**

**OLED\_WR\_Byte(temp, OLED\_DATA);  // 32x64字号(Consolas)**

**} else if (sizey == 36) {**

**temp = pgm\_read\_byte(&asc2\_3618[c][i]);**

**OLED\_WR\_Byte(temp, OLED\_DATA);  // 18x36字号(Consolas)**

**} else if (sizey == 48) {**

**temp = pgm\_read\_byte(&asc2\_4824[c][i]);**

**OLED\_WR\_Byte(temp, OLED\_DATA);  // 24x48字号(Consolas)**

**} else if (sizey == 49) {**

**temp = pgm\_read\_byte(&asc2\_Digital\_2448[c][i]);**

**OLED\_WR\_Byte(temp, OLED\_DATA);  // 24x48字号(DS-Digital)**

**} else**

**return;**

**// 设置列坐标**

**OLED\_Pos = {static\_cast<unsigned char>(static\_cast<unsigned short>(x) + i), y};**

**}**

**}**

**// m^n函数**

**u32 OLED::oled\_pow(u8 m, u8 n) {**

**u32 result = 1;**

**while (n--) result \*= m;**

**return result;**

**}**

**// 显示数字**

**// x,y :起点坐标**

**// num:要显示的数字**

**// len :数字的位数**

**// sizey:字体大小**

**void OLED::OLED\_ShowNum(u8 x, u8 y, u32 num, u8 len, u8 sizey) {**

**u8 t, temp, m = 0;**

**u8 enshow = 0;**

**if (sizey == 8) m = 2;**

**for (t = 0; t < len; t++) {**

**temp = (num / oled\_pow(10, len - t - 1)) % 10;**

**if (enshow == 0 && t < (len - 1)) {**

**if (temp == 0) {**

**OLED\_ShowChar(x + (sizey / 2 + m) \* t, y, ' ', sizey);**

**continue;**

**} else**

**enshow = 1;**

**}**

**OLED\_ShowChar(x + (sizey / 2 + m) \* t, y, temp + '0', sizey);**

**}**

**}**

**// 显示一个字符串**

**void OLED::OLED\_ShowString(u8 x, u8 y, const char \*chr, u8 sizey) {**

**u8 j = 0;**

**while (chr[j] != '\0') {**

**OLED\_ShowChar(x, y, chr[j++], sizey);**

**if (sizey == 8)**

**x += 6;**

**else**

**x += sizey / 2;**

**}**

**}**

**// 显示汉字**

**void OLED::OLED\_ShowChinese(u8 x, u8 y, const u8 no, u8 sizey) {**

**u16 i, size1 = (sizey / 8 + ((sizey % 8) ? 1 : 0)) \* sizey;**

**u8 temp;**

**for (i = 0; i < size1; i++) {**

**if (i % sizey == 0) OLED\_Set\_Pos(x, y++);**

**if (sizey == 32) {**

**temp = pgm\_read\_byte(&FireWarning\_32x32[no][i]);**

**OLED\_WR\_Byte(temp, OLED\_DATA);  // 32x32字号**

**} else if (sizey == 64) {**

**temp = pgm\_read\_byte(&FireWarning\_32x64[no][i]);**

**OLED\_WR\_Byte(temp, OLED\_DATA);  // 32x64字号**

**} else**

**return;**

**}**

**}**

**// 显示图片**

**// x,y显示坐标**

**// sizex,sizey,图片长宽**

**// BMP：要显示的图片**

**void OLED::OLED\_DrawBMP(u8 x, u8 y, u8 sizex, u8 sizey, const u8 BMP[]) {**

**u16 j = 0;**

**u8 i, m, temp;**

**sizey = sizey / 8 + ((sizey % 8) ? 1 : 0);**

**for (i = 0; i < sizey; i++) {**

**OLED\_Set\_Pos(x, i + y);**

**for (m = 0; m < sizex; m++) {**

**temp = pgm\_read\_byte(&BMP[j++]);**

**OLED\_WR\_Byte(temp, OLED\_DATA);**

**}**

**}**

**}**

**// OLED的初始化**

**void OLED::OLED\_Init(void) {**

**I2C\_Init();  // 初始化I2C 管脚输出高电平**

**OLED\_WR\_Byte(0xAE, OLED\_CMD);  //--turn off oled panel**

**OLED\_WR\_Byte(0x00, OLED\_CMD);  //---set low column address**

**OLED\_WR\_Byte(0x10, OLED\_CMD);  //---set high column address**

**OLED\_WR\_Byte(0x40, OLED\_CMD);  //--set start line address  Set Mapping RAM Display Start Line (0x00~0x3F)**

**OLED\_WR\_Byte(0x81, OLED\_CMD);  //--set contrast control register**

**OLED\_WR\_Byte(0xCF, OLED\_CMD);  // Set SEG Output Current Brightness**

**OLED\_WR\_Byte(0xA1, OLED\_CMD);  //--Set SEG/Column Mapping     0xa0左右反置 0xa1正常**

**OLED\_WR\_Byte(0xC8, OLED\_CMD);  // Set COM/Row Scan Direction   0xc0上下反置 0xc8正常**

**OLED\_WR\_Byte(0xA6, OLED\_CMD);  //--set normal display**

**OLED\_WR\_Byte(0xA8, OLED\_CMD);  //--set multiplex ratio(1 to 64)**

**OLED\_WR\_Byte(0x3f, OLED\_CMD);  //--1/64 duty**

**OLED\_WR\_Byte(0xD3, OLED\_CMD);  //-set display offset Shift Mapping RAM Counter (0x00~0x3F)**

**OLED\_WR\_Byte(0x00, OLED\_CMD);  //-not offset**

**OLED\_WR\_Byte(0xd5, OLED\_CMD);  //--set display clock divide ratio/oscillator frequency**

**OLED\_WR\_Byte(0x80, OLED\_CMD);  //--set divide ratio, Set Clock as 100 Frames/Sec**

**OLED\_WR\_Byte(0xD9, OLED\_CMD);  //--set pre-charge period**

**OLED\_WR\_Byte(0xF1, OLED\_CMD);  // Set Pre-Charge as 15 Clocks & Discharge as 1 Clock**

**OLED\_WR\_Byte(0xDA, OLED\_CMD);  //--set com pins hardware configuration**

**OLED\_WR\_Byte(0x12, OLED\_CMD);**

**OLED\_WR\_Byte(0xDB, OLED\_CMD);  //--set vcomh**

**OLED\_WR\_Byte(0x40, OLED\_CMD);  // Set VCOM Deselect Level**

**OLED\_WR\_Byte(0x20, OLED\_CMD);  //-Set Page Addressing Mode (0x00/0x01/0x02)**

**OLED\_WR\_Byte(0x02, OLED\_CMD);  //**

**OLED\_WR\_Byte(0x8D, OLED\_CMD);  //--set Charge Pump enable/disable**

**OLED\_WR\_Byte(0x14, OLED\_CMD);  //--set(0x10) disable**

**OLED\_WR\_Byte(0xA4, OLED\_CMD);  // Disable Entire Display On (0xa4/0xa5)**

**OLED\_WR\_Byte(0xA6, OLED\_CMD);  // Disable Inverse Display On (0xa6/a7)**

**OLED\_Clear();**

**OLED\_WR\_Byte(0xAF, OLED\_CMD); /\*display ON\*/**

**}**

**/\*---------------------------------------------------------------------------------------------------------\*/**

**// 使用OLED\_GDDRAM\_CLONE刷新屏幕,可以将内容预渲染在OLED\_GDDRAM\_CLONE中然后再一次刷新(刷新区域, 默认全屏);**

**void OLED::OLED\_GDDRAM\_Refresh(u8 leftPixel, u8 topPixel, u8 rightPixel, u8 bottomPixel) {**

**//(Pixel >> 3): 计算"topPixel"与"bottomPixel"坐标对应的点位于哪一页;**

**unsigned char topPage = static\_cast<unsigned char>(topPixel >> 3);**

**unsigned char bottomPage = static\_cast<unsigned char>(bottomPixel >> 3);**

**for (u8 i = topPage; i < bottomPage; ++i) {**

**OLED\_WR\_Byte(0xb0 + i, OLED\_CMD);  // 设置页地址（0~7）**

**OLED\_WR\_Byte(0x00, OLED\_CMD);      // 设置显示位置—列低地址**

**OLED\_WR\_Byte(0x10, OLED\_CMD);      // 设置显示位置—列高地址**

**for (u8 n = leftPixel; n < rightPixel; ++n) {**

**OLED\_WR\_Byte(OLED\_GDDRAM\_CLONE[i][n], OLED\_DATA);**

**}**

**}  // 更新显示**

**}**

**// 开始进行批量绘制(开始预渲染);**

**void OLED::BeginBatchDraw() { PreRendered = true; }**

**// 结束进行批量绘制(结束预渲染, 不会自动刷新预渲染缓存的内容);**

**void OLED::EndBatchDraw() { PreRendered = false; }**

**// 这个函数用于画点(点的坐标, enabl:true在屏幕上画一点, enabl:false清除屏幕上的一点);**

**void OLED::putpixel(u8 xPixel, u8 yPixel, bool enable) {**

**if (xPixel > 127 || yPixel > 63) return;  // 非法参数(超出OLED显示区域);**

**// 每页的一列中有8个像素，计算待显示的点在当前列中处于第几点, 然后以此对字节0x01进行移位(得到一个字节的图形数据);**

**unsigned char Byte\_data = static\_cast<unsigned char>(0x01 << (yPixel % 8));**

**//(yPixel >> 3): 计算"yPixel"坐标对应的点位于哪一页;**

**unsigned char Page\_Pos = static\_cast<unsigned char>(yPixel >> 3);**

**// 因为OLED是字节操作的无法真正的进行位操作, 所有我们读取"OLED\_GDDRAM\_CLONE"中的OLED上已经显示的数据,**

**// 把同一位置将要显示的字节和已经显示进行位运算来达到只变更一位的目的;**

**if (enable == true) {**

**Byte\_data |= OLED\_GDDRAM\_CLONE[Page\_Pos][xPixel];  // 第Bit\_xPixel(yPixel % 8)位，置1，其他位值不变;**

**} else {**

**Byte\_data = OLED\_GDDRAM\_CLONE[Page\_Pos][xPixel] & (~Byte\_data);  // 第Bit\_xPixel(yPixel % 8)位，置0，其他位值不变;**

**}**

**if (Byte\_data == OLED\_GDDRAM\_CLONE[Page\_Pos][xPixel]) return;  // 检查该点是否已经被画过(避免重复画点);**

**OLED\_Set\_Pos(xPixel, Page\_Pos);**

**OLED\_WR\_Byte(static\_cast<unsigned char>(0xb0 + Page\_Pos), OLED\_CMD);**

**OLED\_WR\_Byte(static\_cast<unsigned char>(((xPixel & 0xf0) >> 4) | 0x10), OLED\_CMD);**

**OLED\_WR\_Byte(static\_cast<unsigned char>((xPixel & 0x0f) | 0x00), OLED\_CMD);**

**OLED\_WR\_Byte(Byte\_data, OLED\_DATA);**

**}**

**// 这个函数用于画直线;**

**void OLED::line(u8 x1Pixel, u8 y1Pixel, u8 x2Pixel, u8 y2Pixel, bool enable) {**

**unsigned char dx = abs(x2Pixel - x1Pixel);**

**unsigned char dy = abs(y2Pixel - y1Pixel);**

**short sx = (x1Pixel < x2Pixel) ? 1 : -1;**

**short sy = (y1Pixel < y2Pixel) ? 1 : -1;**

**short err = (dx > dy ? dx : -dy) / 2;**

**short e2;**

**while (true) {**

**putpixel(x1Pixel, y1Pixel, enable);**

**if (x1Pixel == x2Pixel && y1Pixel == y2Pixel) break;**

**e2 = err;**

**if (e2 > -dx) {**

**err -= dy;**

**x1Pixel += sx;**

**}**

**if (e2 < dy) {**

**err += dx;**

**y1Pixel += sy;**

**}**

**}**

**}**

**// 这个函数用于画无填充的矩形;**

**void OLED::rectangle(u8 leftPixel, u8 topPixel, u8 rightPixel, u8 bottomPixel) {**

**line(leftPixel, topPixel, rightPixel, topPixel);**

**line(leftPixel, topPixel, leftPixel, bottomPixel);**

**line(rightPixel, topPixel, rightPixel, bottomPixel);**

**line(leftPixel, bottomPixel, rightPixel, bottomPixel);**

**}**

**// 这个函数用于画有边框的填充矩形;**

**void OLED::fillrectangle(u8 leftPixel, u8 topPixel, u8 rightPixel, u8 bottomPixel) {**

**// Draw the four sides of the rectangle**

**line(leftPixel, topPixel, rightPixel, topPixel);**

**line(leftPixel, topPixel, leftPixel, bottomPixel);**

**line(rightPixel, topPixel, rightPixel, bottomPixel);**

**line(leftPixel, bottomPixel, rightPixel, bottomPixel);**

**// Fill the rectangle**

**for (unsigned char i = topPixel; i <= bottomPixel; i++) {**

**line(leftPixel, i, rightPixel, i);**

**}**

**}**

**// 这个函数用于清空矩形区域;**

**void OLED::clearrectangle(u8 leftPixel, u8 topPixel, u8 rightPixel, u8 bottomPixel) {**

**for (unsigned char y = topPixel; y <= bottomPixel; ++y) {**

**for (unsigned char x = leftPixel; x <= rightPixel; ++x) {**

**putpixel(x, y, false);**

**}**

**}**

**}**

**// 字符串分割, 将一个给定的字符串 "input" 根据另一个字符串 "separator" 进行分割，并将分割后的每一段字符串存储在一个 vector 容器中。**

**vector<String> OLED::strsplit(String input, String separator) {**

**vector<String> vecStr;**

**while (input.indexOf(separator) != -1) {**

**unsigned int splitIndex = input.indexOf(separator);**

**String segment = input.substring(0, splitIndex);**

**vecStr.push\_back(segment);**

**input = input.substring(splitIndex + separator.length());**

**}**

**vecStr.push\_back(input);**

**return vecStr;**

**/\***

**该函数的作用是将一个给定的字符串 "input" 根据另一个字符串 "separator" 进行分割，并将分割后的每一段字符串存储在一个 vector 容器中。**

**这个函数首先使用 while 循环来检查输入字符串中是否存在 "separator"，如果存在，则使用 "input.indexOf(separator)" 方法来找到 "separator"**

**第一次出现的位置，然后使用 "input.substring(0, splitIndex)" 方法来截取从 0 到 "splitIndex" 位置的字符串，将截取的字符串存储在 vector 容器 "vecStr"**

**中。接着，使用 "input = input.substring(splitIndex + 分割字符的长度);" 更新输入字符串，以便在下一次循环中继续查找 "separator"。**

**当循环结束后，将剩余的字符串 "input" 添加到 vector 容器 "vecStr" 中，并返回该容器。**

**\*/**

**}**

**// 渲染文本箱;**

**void OLED::drawPrintBox() {**

**BeginBatchDraw();  // 开始预渲染;**

**// 计算当前滚动条位置;**

**sliderPos[0] = static\_cast<unsigned char>(8 \* Hight\_MaxNumChar \* (static\_cast<float>(First\_Line) / static\_cast<float>(PrintBox.size())));**

**line(PrintRECT.right - 5, PrintRECT.top, PrintRECT.right - 5, PrintRECT.bottom);                  // 绘制滚动条栏;**

**clearrectangle(PrintRECT.right - 3, sliderPos[1], PrintRECT.right, sliderPos[1] + sliderHeight);  // 清空上一个滚动条区域;**

**rectangle(PrintRECT.right - 3, sliderPos[0], PrintRECT.right - 1, sliderPos[0] + sliderHeight);   // 绘制滚动条;**

**sliderPos[1] = sliderPos[0];  // 更新滚动条位置;**

**// 绘制字符;**

**unsigned char y = PrintRECT.top;  // 将输入指针移到文本框最顶端;**

**// 显示PrintBox中First\_Line到Hight\_MaxNumChar的内容;**

**for (unsigned int i = First\_Line; i < First\_Line + [](unsigned int PrintBox\_size, unsigned int Hight\_MaxNumChar) -> unsigned int {**

**// 这里要做一个比较, 如果PrintBox\_size小于Hight\_MaxNumChar则显示从首行到尾行的内容(否则会因为强制显示不存在的内容而发生错误!);**

**if (PrintBox\_size < Hight\_MaxNumChar)**

**return PrintBox\_size;**

**else**

**return Hight\_MaxNumChar;**

**}(PrintBox.size(), Hight\_MaxNumChar);**

**++i) {**

**unsigned char x = PrintRECT.left;**

**for (auto &j : PrintBox[i]) {**

**OLED\_ShowChar(x, y, j, 8);**

**x += 5;**

**}**

**++y;**

**}**

**EndBatchDraw();                                                                         // 结束预渲染;**

**OLED\_GDDRAM\_Refresh(PrintRECT.left, PrintRECT.top, PrintRECT.right, PrintRECT.bottom);  // 刷新预渲染内容到屏幕上;**

**/\***

**unsigned char x = PrintRECT.left;**

**unsigned char y = PrintRECT.top;**

**// 计算一行显示最多字符时的像素长度("-5"这个位置用于显示滚动条);**

**unsigned char PrintWidth = static\_cast<unsigned char>(0.2 \* (PrintRECT.right - PrintRECT.left)) \* 5 - 5;**

**unsigned char PrintHight = static\_cast<unsigned char>(0.125 \* (PrintRECT.bottom - PrintRECT.top));  // 计算一列显示最多字符时的像素长度;**

**unsigned int Newline\_Pos = text.indexOf("\n", 0);**

**for (auto &i : text) {**

**// 获取迭代器下标;**

**unsigned int Text\_Pos = static\_cast<unsigned int>(std::distance(text.begin(), &i));**

**// 如果右方超出显示区域或遇到换行符则换行;**

**if (x >= PrintWidth || Text\_Pos == Newline\_Pos) {**

**// 查找下一个换行符在字符串中的位置(从上一个换行符后开始查找);**

**Newline\_Pos = text.indexOf("\n", Newline\_Pos + 1);**

**x = PrintRECT.left;**

**++y;**

**// 如果显示行数超出显示区域则从开位置头显示剩余的字符;**

**if (y >= PrintHight) {**

**x = PrintRECT.left;**

**y = PrintRECT.top;**

**}**

**} else {**

**OLED\_ShowChar(x, y, i, 8);**

**x += 5;**

**}**

**}**

**\*/**

**}**

**// 设置文本框(文本框左部 x 坐标, 文本框顶部 y 坐标, 文本框右部 x 坐标,文本框底部 y 坐标);**

**void OLED::setTextBox(u8 leftPixel, u8 topPixel, u8 rightPixel, u8 bottomPixel) {**

**PrintRECT = {leftPixel, topPixel, rightPixel, bottomPixel};  // 设置文本框位置和大小;**

**First\_Line = 0;                                              // 首行位置设为0;**

**// 若PrintBox不为空, 则清空释放其内存;**

**if (PrintBox.empty() == false) {**

**PrintBox.clear();**

**PrintBox.shrink\_to\_fit();**

**}**

**}**

**// 打印字符串(字符串, 自动滚动[设为true时会随着打印内容而自动向下滚动, 默认启用]);**

**void OLED::print(String text, bool autoScroll) {**

**// 清空文本框区域(但是不要立刻显示出来);**

**BeginBatchDraw();**

**clearrectangle(PrintRECT.left, PrintRECT.top, PrintRECT.right, PrintRECT.bottom);**

**EndBatchDraw();**

**/\*-------------------------------------------------------------------------------------------------------------------------------**

**这里实现的是首先按照换行符换行, 换行符换行完成后检查是否有某行超出显示范围, 如果有则再对其换行.**

**这段代码主要的作用是将文本字符串根据换行符和一行最大字符数分割成若干行,**

**1.首先, 通过计算 PrintRECT 的宽度和高度，计算出最大字符数 Width\_MaxNumChar 和最大行数 Hight\_MaxNumChar。**

**2.然后，通过strsplit函数将文本按照'\n'分割成一个数组。**

**3.接下来，对于每一个分割后的字符串，使用 for 循环将其分割成若干段，每一段不会超过Width\_MaxNumChar个字符。**

**4.对于每一段，使用匿名函数计算出需要填充的空格数，并将这一段文本加上空格作为一个字符串插入到PrintBox中。**

**5.整个过程循环进行直到所有的文本都被分割完成。**

**'\n' 在代码中的作用是用来将文本分割成若干行, 这样就可以按照行来进行分割。**

**--------------------------------------------------------------------------------------------------------------------------------\*/**

**// 计算一行最多能显示多少字符("-1"是为了腾出滚动条的显示空间);**

**Width\_MaxNumChar = static\_cast<unsigned char>(0.2 \* (PrintRECT.right - PrintRECT.left)) - 1;**

**// 计算一列最多能显示多少字符;**

**Hight\_MaxNumChar = static\_cast<unsigned char>(0.125 \* (PrintRECT.bottom - PrintRECT.top));**

**// 通过换行符"\n"分割字符串成若干行;**

**for (auto &i : strsplit(text, "\n")) {**

**// 每行的字符串按照"Width\_MaxNumChar"的字符数分割成若干小字符串行;**

**for (unsigned int j = 0; j < i.length(); j += Width\_MaxNumChar) {**

**PrintBox.push\_back(i.substring(j, min(j + Width\_MaxNumChar, i.length())) + [](unsigned int length, unsigned char Width\_MaxNumChar) -> String {**

**// 每行的末尾可能会没有内容, 这样在文本滚动时造成显示错误, 我们对其添加空格来进行填充;**

**String SpaceChar = "";**

**for (unsigned int k = length; k < Width\_MaxNumChar; ++k) SpaceChar += " ";**

**return SpaceChar;**

**}(i.length(), Width\_MaxNumChar));**

**}**

**}**

**// 计算滚动条高度;**

**sliderHeight = static\_cast<unsigned char>(8 \* Hight\_MaxNumChar \* (static\_cast<float>(Hight\_MaxNumChar) / static\_cast<float>(PrintBox.size())));**

**// 如果启用自动滚动并且文本行数超过屏幕能够显示的行数则滚动到文本最底部;**

**if (autoScroll == true && PrintBox.size() > Hight\_MaxNumChar) First\_Line = PrintBox.size() - Hight\_MaxNumChar;**

**drawPrintBox();  // 渲染文本箱;**

**}**

**// 清空文本框并释放内存;**

**void OLED::clearTextBox() {**

**PrintBox.clear();**

**PrintBox.shrink\_to\_fit();**

**First\_Line = 0;  // 首行位置设为0;**

**};**

**// 获取文本框的全部文本;**

**vector<String> OLED::getPrintBox() { return PrintBox; }**

**// 用一个新的PrintBox结构的数据替换掉现有的PrintBox(要求保证数据结构正确);**

**void OLED::replacePrintBox(vector<String> newPrintBox) { PrintBox = newPrintBox; }**

**// 移动滚动条(true:向下滚动, falas:向上滚动);**

**void OLED::moveScrollBar(bool direction) {**

**if (direction == true && (First\_Line + Hight\_MaxNumChar) < PrintBox.size()) {**

**++First\_Line;**

**} else if (direction == false && First\_Line > 0) {**

**--First\_Line;**

**}**

**drawPrintBox();**

**delay(1);**

**}**

**// Alert.h------------------------------------------------------------------------------------------------------------**

**#pragma once**

**#include "Universal.h"**

**class ALERT {**

**private:**

**String alertFlashFile = "/alert\_log.txt";  // 被读取的文件位置和名称**

**public:**

**// 使能或失能蜂鸣器(Time = 使能/失能 时间[ms], SetState = 状态[true为使能 false为失能]);**

**void BUZZER\_Enable(unsigned short Time, bool SetState = true);**

**// RGBLED使能或失能;**

**void LED\_R\_Enable(unsigned short Time, bool SetState = true);**

**void LED\_G\_Enable(unsigned short Time, bool SetState = true);**

**void LED\_B\_Enable(unsigned short Time, bool SetState = true);**

**// 关闭所有声光警报;**

**void ALERT\_Disable();**

**// 初始化声光报警器;**

**void AlertInit();**

**// Flash写警报日志;**

**void flashWriteAlertLog(String alertLog);**

**// Flash读警报日志;**

**void flashReadAlertLog();**

**};**

**// Alert.cpp----------------------------------------------------------------------------------------------------------**

**#include "Alert.h"**

**void ALERT::BUZZER\_Enable(unsigned short Time, bool SetState) {**

**if (SetState == true) {**

**digitalWrite(0, LOW);**

**digitalWrite(2, LOW);**

**digitalWrite(15, HIGH);**

**} else {**

**digitalWrite(0, HIGH);**

**digitalWrite(2, HIGH);**

**digitalWrite(15, HIGH);**

**}**

**delay(Time);**

**}**

**void ALERT::LED\_R\_Enable(unsigned short Time, bool SetState) {**

**if (SetState == true) {**

**digitalWrite(0, LOW);**

**digitalWrite(2, HIGH);**

**digitalWrite(15, LOW);**

**} else {**

**digitalWrite(0, HIGH);**

**digitalWrite(2, HIGH);**

**digitalWrite(15, HIGH);**

**}**

**delay(Time);**

**}**

**void ALERT::LED\_G\_Enable(unsigned short Time, bool SetState) {**

**if (SetState == true) {**

**digitalWrite(0, LOW);**

**digitalWrite(2, HIGH);**

**digitalWrite(15, HIGH);**

**} else {**

**digitalWrite(0, HIGH);**

**digitalWrite(2, HIGH);**

**digitalWrite(15, HIGH);**

**}**

**delay(Time);**

**}**

**void ALERT::LED\_B\_Enable(unsigned short Time, bool SetState) {**

**if (SetState == true) {**

**digitalWrite(0, HIGH);**

**digitalWrite(2, LOW);**

**digitalWrite(15, LOW);**

**} else {**

**digitalWrite(0, HIGH);**

**digitalWrite(2, HIGH);**

**digitalWrite(15, HIGH);**

**}**

**delay(Time);**

**}**

**void ALERT::ALERT\_Disable() {**

**digitalWrite(0, HIGH);**

**digitalWrite(2, HIGH);**

**digitalWrite(15, HIGH);**

**}**

**void ALERT::AlertInit() {**

**// 138译码器输出引脚(控制声光报警);**

**pinMode(0, OUTPUT);**

**pinMode(2, OUTPUT);**

**pinMode(15, OUTPUT);**

**// 关闭所有声光警报;**

**ALERT\_Disable();**

**}**

**void ALERT::flashWriteAlertLog(String alertLog) {**

**LittleFS.begin();  // 启动LittleFS;**

**File dataFile;**

**// 确认闪存中是否有alertFlashFile文件**

**if (LittleFS.exists(alertFlashFile)) {**

**dataFile = LittleFS.open(alertFlashFile, "a");  // 建立File对象用于向LittleFS中的file对象追加信息(添加);**

**} else {**

**dataFile = LittleFS.open(alertFlashFile, "w");  // 建立File对象用于向LittleFS中的file对象写入信息(新建&覆盖);**

**}**

**dataFile.print(alertLog);  // 向dataFile写入字符串信息**

**dataFile.close();          // 完成文件写入后关闭文件**

**}**

**void ALERT::flashReadAlertLog() {**

**LittleFS.begin();  // 启动LittleFS;**

**File dataFile;**

**// 确认闪存中是否有alertFlashFile文件**

**if (LittleFS.exists(alertFlashFile)) {**

**Serial.println("[FLASH FILE FOUND]" + alertFlashFile);**

**File dataFile = LittleFS.open(alertFlashFile, "r");  // 建立File对象用于从LittleFS中读取文件;**

**// 读取文件内容并且通过串口监视器输出文件信息**

**for (unsigned int i = 0; i < dataFile.size(); ++i) Serial.print((char)dataFile.read());**

**dataFile.close();  // 完成文件读取后关闭文件**

**} else {**

**Serial.println("[FLASH FILE NOT FOUND]" + alertFlashFile);**

**}**

**}**

**// WebServer.h--------------------------------------------------------------------------------------------------------**

**#pragma once**

**#ifndef \_\_WEBSERVER\_H**

**#define \_\_WEBSERVER\_H**

**#include <avr/pgmspace.h>**

**// 用于打印的操作系统信息;**

**const char GSG3\_Os\_Info[] PROGMEM = {**

**"GasSensorGen3 OS\nBuild: GS.20230130.Mark0\nUpdate: github.com/RMSHE-MSH\nHardware: GS.Gen3.20230110.Mark1\nPowered by "**

**"RMSHE\nE-mail: asdfghjkl851@outlook.com"};**

**// CMDControlPanel帮助信息;**

**const char CMDCP\_HELP[] PROGMEM = {**

**"GasSenserOS RMSHE >> CMDControlPanel help"**

**"\npwd : Print work directory."**

**"\nls : List work directory files."**

**"\ncd [dirName] : Change work directory."**

**"\ncat [fileName] : Open the file in the work directory."**

**"\ntouch [fileName] : Create an empty file in the work directory."**

**"\nmkdir [dirName] : Create a directory under the work directory."**

**"\necho [string] : Printed to the CMD."**

**"\necho [string] > [fileName] : Overwrite the file in the work directory."**

**"\necho [string] >> [fileName] : Append to the file in the work directory."**

**"\nrm [fileName] : Remove files in the work directory."**

**"\nrm -r [dirName] : Remove the directory under the work directory."**

**"\ncp [sourceFilePath] [targetFilePath] : Copy file."**

**"\ncp -r [sourceDirPath] [targetDirPath] : Copy directory."**

**"\nmv [sourceFilePath] [targetFilePath] : Move file."**

**"\nmv -r [sourceDirPath] [targetDirPath] : Move directory."**

**"\nfind [dirPath] [fileName] : Find files in the directory, fileName example = (\*.\*/\*.txt/a.txt)."**

**"\nosinfo : Display operating system version information."**

**"\nreboot : MCU reset."**

**"\npios : Print GPIO status."**

**"\npss : Print System status."**

**"\nled [color] [state] : Turn on RGB LED, color = (r/g/b), state = ((1/true/enable), (0/false/disable))."**

**"\nbuzz [state] : Turn on BUZZER, state = ((1/true/enable), (0/false/disable))."**

**"\nalertdis : Alert disable."**

**"\nfreeze [enable] : Light sleep, enable = ((1/true/enable), (0/false/disable))."**

**"\ndisk [time\_us] : Deep sleep, time\_us = (1 to 4294967295 Microsecond)."**

**"\nhistory : Show command history."**

**"\nhistory -s : Display commands executed before deep sleep."**

**"\nhistory -c : Remove command history."**

**"\nwho : View the IP address of the user terminal logged into the current host."**

**"\nlast : View system login logs."**

**"\nlast -c : Remove system login logs."**

**"\ndate : Display system time."**

**"\ndate -n : Synchronize network time."**

**"\ndate -s [timeStr] : Set system time, timeStr = 20230203121601 (Year Month Day Hour Minute Second)."**

**"\nweather : Show current real-time weather."**

**"\nweather -n : Synchronize live weather on the web."**

**"\nweather -s [cityID] : To set the city, fill in the \"cityID\" with the city ID of Know Your Weather."**

**"\npgup : Text box scrolls up one line."**

**"\npgup -s [line] : The text box is scrolled up, \"line\" is the number of lines to scroll."**

**"\npgdn : Text box scrolls down."**

**"\npgdn -s [line] : The text box is scrolled down, \"line\" is the number of lines to scroll."**

**"\nclear : Clear console and free memory."**

**"\nupload : Uploading files from the terminal to the server."**

**"\nupload -s : View the results of the last file upload."**

**"\ndf : Display Flash information."**

**"\nfree : Display remaining RAM."**

**"\nwifi [SSID] [PASSWORD] : Configure WIFI connection, set WIFI SSID and PASSWORD."**

**"\npoweroff : Indefinite deep sleep."**

**"\nlogout : Log out and lock CMDCP."**

**"\nlogout -k [clientIP] : Logout of the terminals with the specified IP address."**

**"\nlogout -k other : Logout of other terminals except yourself."**

**"\nlogout -k all : logout of all terminals"};**

**const char CMDCP\_Online[] PROGMEM = R"rawliteral(**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>GasSenserOS Web Server</title>**

**<style>**

**body {**

**background-color: #343541;**

**color: #ececf1;**

**font-family: Arial, sans-serif;**

**font-size: 16px;**

**line-height: 1.5;**

**margin: 0;**

**padding: 0;**

**transition: all 0.24s ease-in-out;**

**}**

**#console {**

**position: fixed;**

**top: 30px;**

**left: 0;**

**right: 0;**

**float: left;**

**overflow-y: scroll;**

**overflow-x: auto;**

**width: calc(90vw - 40px);**

**height: calc(100vh - 194px);**

**border: 0px solid rgba(255, 255, 255, 0);**

**padding: 10px;**

**margin: 10px;**

**background-color: rgba(0, 0, 0, 0.7);**

**color: #ececf1;**

**border-radius: 6px;**

**font-size: 20px;**

**scrollbar-width: none;**

**resize: none;**

**position: relative;**

**box-shadow: 0px -12px 0px #202123;**

**left: 50%;**

**transform: translateX(-50%);**

**transition: all 0.24s ease-in-out;**

**}**

**.text-RMSHE {**

**position: absolute;**

**font-size: 80px;**

**color: #40414f00;**

**filter: blur(0px);**

**align-items: center;**

**left: 50%;**

**bottom: 40px;**

**transform: translateX(-50%);**

**transition: all 0.5s ease-in-out;**

**}**

**#console:focus {**

**background-color: #000000;**

**outline: none;**

**}**

**#console::selection {**

**background-color: #33333300;**

**color: #79b8ff;**

**}**

**#console::-webkit-scrollbar {**

**width: 8px;**

**background-color: #1f1f1f00;**

**position: absolute;**

**right: 0;**

**top: 200;**

**border-radius: 20px;**

**}**

**#console::-webkit-scrollbar-thumb {**

**border-radius: 10px;**

**background-color: #565869;**

**border: 0px solid #1f1f1f;**

**}**

**#container {**

**position: absolute;**

**bottom: 30px;**

**float: left;**

**width: calc(90vw - 40px);**

**height: 60px;**

**margin: 10px;**

**padding: 10px;**

**background-color: #40414f00;**

**color: #ececf1;**

**border-radius: 10px;**

**resize: auto;**

**left: 50%;**

**transform: translateX(-50%);**

**display: flex;**

**align-items: center;**

**transition: all 0.24s ease-in-out;**

**}**

**#message {**

**width: calc(100% - 139px);**

**height: 60px;**

**background-color: #40414f;**

**color: #ececf1;**

**border: 0px solid #FFF;**

**border-radius: 10px;**

**resize: auto;**

**font-size: 24px;**

**text-align: auto;**

**padding-left: 18px;**

**box-shadow: 0px 0px 6px #303139;**

**transition: all 0.2s ease-in-out;**

**}**

**#sendButton,**

**#uploadButton {**

**margin-left: 16px;**

**width: 90px;**

**height: 60px;**

**background-color: #40414f;**

**color: #acacbe;**

**border: none;**

**font-size: 24px;**

**border-radius: 10px;**

**box-shadow: 0px 0px 6px #303139;**

**transition: all 0.2s ease-in-out;**

**}**

**#uploadButton {**

**width: 90px;**

**height: 60px;**

**padding: 10px;**

**box-shadow: 0px 0px 6px #303139;**

**transition: all 0.2s ease-in-out;**

**}**

**#message:focus {**

**background-color: #444654;**

**outline: none;**

**box-shadow: 0px 0px 20px #303139;**

**}**

**#sendButton:active,**

**#uploadButton:active {**

**background-color: #40414f;**

**box-shadow: 0px 0px 10px #202123;**

**border: 2px solid #565869;**

**filter: blur(2px);**

**font-size: 22px;**

**}**

**#sendButton:hover,**

**#uploadButton:hover {**

**background-color: #202123;**

**}**

**.text-info {**

**font-family: Arial, sans-serif;**

**font-weight: bold;**

**position: absolute;**

**bottom: 4px;**

**font-size: 12px;**

**color: rgba(153, 153, 161, 100);**

**text-align: center;**

**left: 50%;**

**transform: translateX(-50%);**

**filter: blur(0px);**

**transition: all 0.5s ease-in-out;**

**user-select: none;**

**white-space: nowrap;**

**}**

**</style>**

**<script>**

**function updateMessage() {**

**//服务器响应;**

**var xhttp = new XMLHttpRequest();**

**xhttp.onreadystatechange = function () {**

**if (this.readyState == 4 && this.status == 200 && this.responseText != "") {**

**let input = this.responseText;**

**let text = input.split("\n");//对响应字符串进行按行分割;**

**let index = 0;**

**//创建定时器(每隔8ms)使字符串逐行出现;**

**const interval = setInterval(function () {**

**document.getElementById("console").value += "\n" + text[index];//逐行输出;**

**document.getElementById("console").scrollTop = document.getElementById("console").scrollHeight;//滚动到最底部;**

**index++;**

**//如果全部输出完成则停止定时器;**

**if (index === text.length) {**

**clearInterval(interval);**

**}**

**}, 8);**

**//document.getElementById("console").value += "\n" + this.responseText;**

**//document.getElementById("console").scrollTop = document.getElementById("console").scrollHeight;**

**if (this.responseText == "EnableUpload") {**

**var fileInput = document.getElementById("fileInput");**

**var file = fileInput.files[0];**

**var formData = new FormData();**

**formData.append("file", file);**

**var xhr = new XMLHttpRequest();**

**xhr.open("POST", "http://192.168.43.164:80/upload", true);**

**xhr.send(formData);**

**//等待1000ms;**

**setTimeout(function () {**

**document.getElementById("fileInput").value = "";//清空文件选择器;**

**//向服务器发送指令查看文件是否上传成功;**

**xhttp.open("GET", "http://192.168.43.164:80/CMD?message=upload -s", true);**

**xhttp.send();**

**}, 1000);**

**}**

**}**

**};**

**//向服务器请求;**

**var message = document.getElementById("message").value;**

**document.getElementById("console").value += "\n> " + message;**

**document.getElementById("console").scrollTop = document.getElementById("console").scrollHeight;**

**xhttp.open("GET", "http://192.168.43.164:80/CMD?message=" + message, true);**

**xhttp.send();**

**if (message == "clear") document.getElementById("console").value = "GasSenserOS RMSHE >> CMDControlPanel";**

**document.getElementById("message").value = "";**

**}**

**window.onload = function () {**

**//将uploadButton按钮与隐藏的"选择文件"控件绑定;**

**document.getElementById("uploadButton").addEventListener("click", function (event) {**

**event.preventDefault();**

**document.getElementById("fileInput").click();**

**});**

**//监测文件选择器是否有文件, 如果有文件则将按钮文本修改为"Load";**

**var fileInput = document.getElementById("fileInput");**

**document.querySelector("Input[type=file]").addEventListener("change", function (e) {**

**if (e.target.files.length) {**

**uploadButton.innerHTML = "Load";**

**document.getElementById("message").value = "upload";**

**} else {**

**uploadButton.innerHTML = "File";**

**}**

**});**

**/\***

**//当输入框或编辑框获得焦点时将背景RMSHE虚化;**

**document.getElementById("message").addEventListener("focus", function (event) {**

**document.querySelector(".text-RMSHE").style.filter = "blur(16px)";**

**});**

**document.getElementById("console").addEventListener("focus", function (event) {**

**document.querySelector(".text-RMSHE").style.filter = "blur(16px)";**

**});**

**//当输入框或编辑框获得焦点时将背景RMSHE实体化;**

**document.getElementById("message").addEventListener("blur", function (event) {**

**document.querySelector(".text-RMSHE").style.filter = "blur(0)";**

**});**

**document.getElementById("console").addEventListener("blur", function (event) {**

**document.querySelector(".text-RMSHE").style.filter = "blur(0)";**

**});**

**\*/**

**}**

**//浏览器窗口大小改变时将文本虚化，清除任何现有的计时器并重新设置一个新的计时器。如果 500 毫秒内没有更改窗口大小，则将触发计时器回调函数并将文本实体化。**

**var resizeTimer;**

**window.addEventListener("resize", function (event) {**

**//document.querySelector(".text-RMSHE").style.filter = "blur(30px)";**

**document.querySelector(".text-info").style.filter = "blur(30px)";**

**document.querySelector(".text-info").style.color = "rgba(153, 153, 161, 0)";**

**document.querySelector(".text-info").style.fontSize = "0px";**

**clearTimeout(resizeTimer);**

**resizeTimer = setTimeout(function () {**

**//document.querySelector(".text-RMSHE").style.filter = "blur(0)";**

**document.querySelector(".text-info").style.filter = "blur(0)";**

**document.querySelector(".text-info").style.color = "rgba(153, 153, 161, 100)";**

**document.querySelector(".text-info").style.fontSize = "12px";**

**}, 500);**

**});**

**// 调整字体大小的函数**

**window.onresize = function () {**

**var textarea = document.getElementById("console");**

**var InputBox = document.getElementById("message");**

**var sendButton = document.getElementById("sendButton");**

**var uploadButton = document.getElementById("uploadButton");**

**var container = document.getElementById("container");**

**var fileInput = document.getElementById("fileInput");**

**var width\_size = window.innerWidth / 50;**

**var height\_size = window.innerHeight / 10;**

**var Button\_width\_size = window.innerWidth / 15;**

**//textarea编辑框字体动态调整;**

**if (width\_size < 12) {**

**textarea.style.fontSize = "12px";**

**} else if (width\_size > 20) {**

**textarea.style.fontSize = "20px";**

**} else {**

**textarea.style.fontSize = width\_size + "px";**

**}**

**//message输入框字体动态调整;**

**if (width\_size < 16) {**

**InputBox.style.fontSize = "16px";**

**} else if (width\_size > 24) {**

**InputBox.style.fontSize = "24px";**

**} else {**

**InputBox.style.fontSize = width\_size + "px";**

**}**

**//sendButton发送按钮字体动态调整;**

**if (width\_size < 20) {**

**sendButton.style.fontSize = "20px";**

**} else if (width\_size > 24) {**

**sendButton.style.fontSize = "24px";**

**} else {**

**sendButton.style.fontSize = width\_size + "px";**

**}**

**//sendButton发送按钮宽度动态调整;**

**if (Button\_width\_size < 40) {**

**sendButton.style.width = "40px";**

**} else if (Button\_width\_size > 90) {**

**sendButton.style.width = "90px";**

**} else {**

**sendButton.style.width = Button\_width\_size + "px";**

**}**

**if (Button\_width\_size < 65) {**

**sendButton.innerHTML = "S";//修改发送按钮文本内容为缩写;**

**} else {**

**sendButton.innerHTML = "Sead";//修改发送按钮文本内容为全拼;**

**}**

**//uploadButton文件选择按钮字体动态调整;**

**if (width\_size < 20) {**

**uploadButton.style.fontSize = "20px";**

**} else if (width\_size > 24) {**

**uploadButton.style.fontSize = "24px";**

**} else {**

**uploadButton.style.fontSize = width\_size + "px";**

**}**

**//uploadButton文件选择按钮宽度动态调整;**

**if (Button\_width\_size < 40) {**

**uploadButton.style.width = "40px";**

**} else if (Button\_width\_size > 90) {**

**uploadButton.style.width = "90px";**

**} else {**

**uploadButton.style.width = Button\_width\_size + "px";**

**}**

**if (Button\_width\_size < 65) {**

**//修改选择文件按钮文本内容为缩写;**

**if (fileInput.value != "") { uploadButton.innerHTML = "L"; } else { uploadButton.innerHTML = "F"; }**

**} else {**

**//修改选择文件按钮文本内容为全拼;**

**if (fileInput.value != "") { uploadButton.innerHTML = "Load"; } else { uploadButton.innerHTML = "File"; }**

**}**

**//container输入栏高度动态调整;**

**if (height\_size < 40) {**

**container.style.height = "40px";**

**InputBox.style.height = "40px";**

**sendButton.style.height = "40px";**

**uploadButton.style.height = "40px";**

**} else if (height\_size > 60) {**

**container.style.height = "60px";**

**InputBox.style.height = "60px";**

**sendButton.style.height = "60px";**

**uploadButton.style.height = "60px";**

**} else {**

**container.style.height = height\_size + "px";**

**InputBox.style.height = height\_size + "px";**

**sendButton.style.height = height\_size + "px";**

**uploadButton.style.height = height\_size + "px";**

**}**

**};**

**</script>**

**</head>**

**<body>**

**<div class="console-RMSHE">**

**<p class="text-RMSHE">RMSHE</p>**

**<textarea id="console">GasSenserOS RMSHE >> CMDControlPanel</textarea><br>**

**</div>**

**<div id="container">**

**<input type="text" id="message" onkeydown="if (event.key === 'Enter') { updateMessage(); }">**

**<button id="sendButton" onclick="updateMessage()">Send</button><br>**

**<form method="POST" enctype="multipart/form-data">**

**<input type="file" id="fileInput" style="display:none">**

**<button id="uploadButton">File</button>**

**</form>**

**</div>**

**<p class="text-info">GasSenserOS CMDControlPanel Online. Powered by RMSHE and ChatGPT.</p>**

**</body>**

**</html>**

**)rawliteral";**

**#endif**

**/\***

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>GasSenserOS Web Server</title>**

**<style>**

**body {**

**background-color: #343541;**

**color: #ececf1;**

**font-family: Arial, sans-serif;**

**font-size: 16px;**

**line-height: 1.5;**

**margin: 0;**

**padding: 0;**

**}**

**#console {**

**position: fixed;**

**top: 20px;**

**float: left;**

**overflow-y: scroll;**

**overflow-x: auto;**

**width: 91.3%;**

**height: 500px;**

**border: 0px solid rgba(255, 255, 255, 0);**

**padding: 10px;**

**margin: 10px;**

**background-color: #000000c7;**

**color: #ececf1;**

**border-radius: 6px;**

**scrollbar-width: none;**

**resize: none;**

**position: relative;**

**font-size: 16px;**

**box-shadow: 0px -10px 0px #202123;**

**}**

**#console:focus {**

**background-color: #000000;**

**outline: none;**

**}**

**#console::selection {**

**background-color: #33333300;**

**color: #79b8ff;**

**}**

**#console::-webkit-scrollbar {**

**width: 8px;**

**background-color: #1f1f1f00;**

**position: absolute;**

**right: 0;**

**top: 200;**

**border-radius: 20px;**

**}**

**#console::-webkit-scrollbar-thumb {**

**border-radius: 10px;**

**background-color: #565869;**

**border: 0px solid #1f1f1f;**

**}**

**#message {**

**position: relative;**

**top: 20px;**

**float: left;**

**width: 75%;**

**height: 40px;**

**margin: 10px;**

**border: 0px solid #FFF;**

**padding: 10px;**

**background-color: #40414f;**

**color: #ececf1;**

**font-size: 24px;**

**border-radius: 10px;**

**resize: auto;**

**box-shadow: 0px 0px 6px #303139;**

**}**

**#message:focus {**

**background-color: #444654;**

**outline: none;**

**}**

**button {**

**position: relative;**

**top: 20px;**

**float: left;**

**width: 59px;**

**height: 59px;**

**background-color: #acacbe;**

**color: #343541;**

**border: none;**

**margin: 10px 0 10px 10px;**

**font-size: 24px;**

**border-radius: 10px;**

**box-shadow: 0px 0px 6px #303139;**

**}**

**</style>**

**<script>**

**function updateMessage() {**

**var xhttp = new XMLHttpRequest();**

**xhttp.onreadystatechange = function () {**

**if (this.readyState == 4 && this.status == 200 && this.responseText != "") {**

**document.getElementById("console").value += "\n" + this.responseText;**

**document.getElementById("console").scrollTop = document.getElementById("console").scrollHeight;**

**}**

**};**

**var message = document.getElementById("message").value;**

**document.getElementById("console").value += "\n> " + message;**

**document.getElementById("console").scrollTop = document.getElementById("console").scrollHeight;**

**xhttp.open("GET", "http://192.168.31.175:80/CMD?message=" + message, true);**

**xhttp.send();**

**document.getElementById("message").value = "";**

**}**

**</script>**

**</head>**

**<body>**

**<textarea id="console">GasSenserOS RMSHE >> CMDControlPanel</textarea><br>**

**<input type="text" id="message" onkeydown="if (event.key === 'Enter') { updateMessage(); }">**

**<button onclick="updateMessage()">>></button><br>**

**</body>**

**</html>**

**\*/**

**// WeatherNow.h-------------------------------------------------------------------------------------------------------**

**#pragma once**

**#ifndef \_WEATHERNOW\_H\_**

**#define \_WEATHERNOW\_H\_**

**#include <Arduino.h>**

**#include <ArduinoJson.h>**

**#include <ESP8266WiFi.h>**

**// #define DEBUG  // 调试用宏定义**

**// 获取当前天气信息类**

**class WeatherNow {**

**public:**

**void config(String userKey, String location, String unit);**

**bool update();**

**String getCityID();  // 返回当前城市（字符串格式）**

**String getCityName();  // 返回当前城市名称（字符串格式）**

**String getCountry();  // 返回当前城市国家（字符串格式）**

**String getPath();  // 返回当前城市路径（字符串格式）**

**String getTimezone();  // 返回当前城市时区名称（字符串格式）**

**String getTimezoneOffset();  // 返回当前城市时区（字符串格式）**

**String getWeatherText();  // 返回当前天气信息（字符串格式）**

**int getWeatherCode();  // 返回当前天气信息（整数格式）**

**int getTemperature();  // 返回当前气温;**

**String getLastUpdate();  // 返回心知天气信息更新时间;**

**String getServerCode();  // 返回服务器响应状态码;**

**private:**

**const char\* \_host = "api.seniverse.com";  // 服务器地址**

**String \_reqUserKey;   // 私钥**

**String \_reqLocation;  // 城市**

**String \_reqUnit;      // 摄氏/华氏**

**void \_parseNowInfo(WiFiClient client);  // 解析实时天气信息信息**

**String \_status\_response = "no\_init";  // 服务器响应状态行**

**String \_response\_code = "no\_init";    // 服务器响应状态码**

**String \_now\_id\_str = "no\_init";**

**String \_now\_name\_str = "no\_init";**

**String \_now\_country\_str = "no\_init";**

**String \_now\_path\_str = "no\_init";**

**String \_now\_timezone\_str = "no\_init";**

**String \_now\_timezone\_offset\_str = "no\_init";**

**String \_now\_text\_str = "no\_init";**

**int \_now\_code\_int = -1;**

**int \_now\_temperature\_int = -127;**

**String \_last\_update\_str = "no\_init";**

**};**

**#endif**

**// WeatherNow.cpp-----------------------------------------------------------------------------------------------------**

**#include "WeatherNow.h"**

**/\* 配置心知天气请求信息**

**\* @param userKey  用户心知天气私钥**

**\* @param location 获取信息的城市参数**

**\* @param location 获取信息的温度单位(摄氏/华氏)**

**\*/**

**void WeatherNow::config(String userKey, String location, String unit) {**

**\_reqUserKey = userKey;**

**\_reqLocation = location;**

**\_reqUnit = unit;**

**}**

**/\* 尝试从心知天气更新信息**

**\* @return: bool 成功更新返回真，否则返回假**

**\*/**

**bool WeatherNow::update() {**

**WiFiClient \_wifiClient;**

**String reqRes = "/v3/weather/now.json?key=" + \_reqUserKey + +"&location=" + \_reqLocation + "&language=en&unit=" + \_reqUnit;**

**String httpRequest = String("GET ") + reqRes + " HTTP/1.1\r\n" + "Host: " + \_host + "\r\n" + "Connection: close\r\n\r\n";**

**#ifdef DEBUG**

**Serial.print("Connecting to ");**

**Serial.print(\_host);**

**#endif DEBUG**

**if (\_wifiClient.connect(\_host, 80)) {**

**#ifdef DEBUG**

**Serial.println(" Success!");**

**#endif DEBUG**

**// 向服务器发送http请求信息**

**\_wifiClient.print(httpRequest);**

**#ifdef DEBUG**

**Serial.println("Sending request: ");**

**Serial.println(httpRequest);**

**#endif DEBUG**

**// 获取并显示服务器响应状态行**

**String \_status\_response = \_wifiClient.readStringUntil('\n');**

**#ifdef DEBUG**

**Serial.print("\_status\_response: ");**

**Serial.println(\_status\_response);**

**#endif DEBUG**

**// 查验服务器是否响应200 OK**

**\_response\_code = \_status\_response.substring(9, 12);**

**if (\_response\_code == "200") {**

**#ifdef DEBUG**

**Serial.println("Response Code: 200");**

**#endif DEBUG**

**} else {**

**#ifdef DEBUG**

**Serial.println(F("Response Code: NOT 200"));**

**#endif DEBUG**

**\_wifiClient.stop();**

**return false;**

**}**

**// 使用find跳过HTTP响应头**

**if (\_wifiClient.find("\r\n\r\n")) {**

**#ifdef DEBUG**

**Serial.println("Found Header End. Start Parsing.");**

**#endif DEBUG**

**}**

**\_parseNowInfo(\_wifiClient);**

**\_wifiClient.stop();**

**return true;**

**} else {**

**#ifdef DEBUG**

**Serial.println(" connection failed!");**

**#endif DEBUG**

**\_wifiClient.stop();**

**return false;**

**}**

**}**

**// 配置心知天气请求信息**

**void WeatherNow::\_parseNowInfo(WiFiClient httpClient) {**

**const size\_t capacity = JSON\_ARRAY\_SIZE(1) + JSON\_OBJECT\_SIZE(1) + 2 \* JSON\_OBJECT\_SIZE(3) + JSON\_OBJECT\_SIZE(6) + 230;**

**DynamicJsonDocument doc(capacity);**

**deserializeJson(doc, httpClient);**

**JsonObject results\_0 = doc["results"][0];**

**JsonObject results\_0\_location = results\_0["location"];**

**JsonObject results\_0\_now = results\_0["now"];**

**// 通过串口监视器显示以上信息**

**\_now\_id\_str = results\_0\_location["id"].as<String>();**

**\_now\_name\_str = results\_0\_location["name"].as<String>();**

**\_now\_country\_str = results\_0\_location["country"].as<String>();**

**\_now\_path\_str = results\_0\_location["path"].as<String>();**

**\_now\_timezone\_str = results\_0\_location["timezone"].as<String>();**

**\_now\_timezone\_offset\_str = results\_0\_location["timezone\_offset"].as<String>();**

**\_now\_text\_str = results\_0\_now["text"].as<String>();**

**\_now\_code\_int = results\_0\_now["code"].as<int>();**

**\_now\_temperature\_int = results\_0\_now["temperature"].as<int>();**

**\_last\_update\_str = results\_0["last\_update"].as<String>();**

**}**

**// 返回当前城市（字符串格式）**

**String WeatherNow::getCityID() { return \_now\_id\_str; }**

**// 返回当前城市名称（字符串格式）**

**String WeatherNow::getCityName() { return \_now\_name\_str; }**

**// 返回当前城市国家（字符串格式）**

**String WeatherNow::getCountry() { return \_now\_country\_str; }**

**// 返回当前城市路径（字符串格式）**

**String WeatherNow::getPath() { return \_now\_path\_str; }**

**// 返回当前城市时区名称（字符串格式）**

**String WeatherNow::getTimezone() { return \_now\_timezone\_str; }**

**// 返回当前城市时区（字符串格式）**

**String WeatherNow::getTimezoneOffset() { return \_now\_timezone\_offset\_str; }**

**// 返回当前天气信息（字符串格式）**

**String WeatherNow::getWeatherText() { return \_now\_text\_str; }**

**// 返回当前天气信息（整数格式）**

**int WeatherNow::getWeatherCode() { return \_now\_code\_int; }**

**// 返回当前气温**

**int WeatherNow::getTemperature() { return \_now\_temperature\_int; }**

**// 返回心知天气信息更新时间**

**String WeatherNow::getLastUpdate() { return \_last\_update\_str; }**

**// 返回服务器响应状态码**

**String WeatherNow::getServerCode() { return \_response\_code; }**

**// Tool.h-------------------------------------------------------------------------------------------------------------**

**#pragma once**

**#include "Universal.h"**

**class TOOL {**

**public:**

**int findArrMax(int arr[], int n);**

**int findArrMin(int arr[], int n);**

**String XOR\_encrypt(String plaintext, String key);**

**};**

**// Tool.cpp-----------------------------------------------------------------------------------------------------------**

**#pragma once**

**#include "Tool.h"**

**// 查找数组中的最大值(目标数组, 数组长度);**

**int TOOL::findArrMax(int arr[], int n) {**

**int max = arr[0];**

**for (int i = 1; i < n; i++) {**

**if (arr[i] > max) {**

**max = arr[i];**

**}**

**}**

**return max;**

**}**

**// 查找数组中的最小值(目标数组, 数组长度);**

**int TOOL::findArrMin(int arr[], int n) {**

**int min = arr[0];**

**for (int i = 1; i < n; i++) {**

**if (arr[i] < min) {**

**min = arr[i];**

**}**

**}**

**return min;**

**}**

**String TOOL::XOR\_encrypt(String plaintext, String key) {**

**String ciphertext = "";**

**unsigned char j = 0;**

**for (char c : plaintext) {**

**ciphertext += char(c ^ key[j]);**

**j = (j + 1) % key.length();**

**}**

**return ciphertext;**

**}**

**// Hash.h-------------------------------------------------------------------------------------------------------------**

**#pragma once**

**#include <Arduino.h>**

**#include <bearssl/bearssl\_hash.h>**

**#ifndef HASH\_H\_**

**#define HASH\_H\_**

**// #define DEBUG\_SHA1**

**void sha1(const uint8\_t\* data, uint32\_t size, uint8\_t hash[20]);**

**void sha1(const char\* data, uint32\_t size, uint8\_t hash[20]);**

**void sha1(const String& data, uint8\_t hash[20]);**

**String sha1(const uint8\_t\* data, uint32\_t size);**

**String sha1(const char\* data, uint32\_t size);**

**String sha1(const String& data);**

**#endif /\* HASH\_H\_ \*/**

**// Hash.cpp-----------------------------------------------------------------------------------------------------------**

**#include "Hash.h"**

**/\*\***

**\* create a sha1 hash from data**

**\* @param data uint8\_t \***

**\* @param size uint32\_t**

**\* @param hash uint8\_t[20]**

**\*/**

**void sha1(const uint8\_t\* data, uint32\_t size, uint8\_t hash[20]) {**

**br\_sha1\_context ctx;**

**#ifdef DEBUG\_SHA1**

**os\_printf("DATA:");**

**for (uint16\_t i = 0; i < size; i++) {**

**os\_printf("%02X", data[i]);**

**}**

**os\_printf("\n");**

**os\_printf("DATA:");**

**for (uint16\_t i = 0; i < size; i++) {**

**os\_printf("%c", data[i]);**

**}**

**os\_printf("\n");**

**#endif**

**br\_sha1\_init(&ctx);**

**br\_sha1\_update(&ctx, data, size);**

**br\_sha1\_out(&ctx, hash);**

**#ifdef DEBUG\_SHA1**

**os\_printf("SHA1:");**

**for (uint16\_t i = 0; i < 20; i++) {**

**os\_printf("%02X", hash[i]);**

**}**

**os\_printf("\n\n");**

**#endif**

**}**

**void sha1(const char\* data, uint32\_t size, uint8\_t hash[20]) { sha1((const uint8\_t\*)data, size, hash); }**

**void sha1(const String& data, uint8\_t hash[20]) { sha1(data.c\_str(), data.length(), hash); }**

**String sha1(const uint8\_t\* data, uint32\_t size) {**

**uint8\_t hash[20];**

**String hashStr((const char\*)nullptr);**

**hashStr.reserve(20 \* 2 + 1);**

**sha1(&data[0], size, &hash[0]);**

**for (uint16\_t i = 0; i < 20; i++) {**

**char hex[3];**

**snprintf(hex, sizeof(hex), "%02x", hash[i]);**

**hashStr += hex;**

**}**

**return hashStr;**

**}**

**String sha1(const char\* data, uint32\_t size) { return sha1((const uint8\_t\*)data, size); }**

**String sha1(const String& data) { return sha1(data.c\_str(), data.length()); }**

**// Universal.h--------------------------------------------------------------------------------------------------------**

**#pragma once**

**#include <Arduino.h>**

**#include <ArduinoJson.h>**

**#include <ESP8266HTTPClient.h>**

**#include <ESP8266WebServer.h>**

**#include <ESP8266WiFi.h>**

**#include <Ticker.h>**

**#include <Wire.h>**

**#include <vector>**

**#include "LittleFS.h"**

**#include "oledfont.h"**

**#define UINT8\_MIN 0**

**#define UINT16\_MIN 0**

**#define UINT32\_MIN 0**

**#define UINT64\_MIN 0**

**#define UINT8\_MAX 255**

**#define UINT16\_MAX 65535**

**#define UINT32\_MAX 4294967295**

**#define UINT64\_MAX 18446744073709551615**

**#define DeepSleep\_MAX 4294967295**

**// 复位引脚定义**

**#define RST 16**

**// 模拟引脚定义;**

**#define SENANALOG A0**

**// 74HC138译码器输出引脚定义;**

**#define Decoder\_C 0**

**#define Decoder\_B 2**

**#define Decoder\_A 15**

**// I2C管脚的定义;**

**#define SDA 4**

**#define SCL 5**

**// 串口管脚定义;**

**#define TXD 1**

**#define RXD 3**

**// 电池状态引脚定义;**

**#define CHRG 14**

**// 电池低电量引脚定义;**

**#define LOWPOWER 12**

**// 传感器输出引脚定义;**

**#define SENOUT 13**

**// WIFI信息;**

**#define ServerPort 80**

**#define SSID "RMSHE"**

**#define PASSWORD "GAATTC-A23187"**

**/\* 设备的三元组信息\*/**

**#define PRODUCT\_KEY "i6abR7NBjfB"**

**#define DEVICE\_NAME "GasSensor\_OS\_ESP8266"**

**#define DEVICE\_SECRET "6269beb7dd1c4f92a29560441970f9de"**

**#define REGION\_ID "cn-shanghai"**

**/\* 线上环境域名和端口号，不需要改 \*/**

**#define MQTT\_SERVER PRODUCT\_KEY ".iot-as-mqtt." REGION\_ID ".aliyuncs.com"**

**#define MQTT\_PORT 1883**

**#define MQTT\_CLIENT\_ID "i6abR7NBjfB.GasSensor\_OS\_ESP8266|securemode=2,signmethod=hmacsha256,timestamp=1675522358253|"**

**// 算法工具: http://iot-face.oss-cn-shanghai.aliyuncs.com/tools.htm 进行加密生成password**

**// password教程 https://www.yuque.com/cloud-dev/iot-tech/mebm5g**

**#define MQTT\_USRNAME DEVICE\_NAME "&" PRODUCT\_KEY**

**#define MQTT\_PASSWD "01a7128c6d546845b23f4c0521355c96e285b360a469be8f906371ac3a5e9d01"**

**// 连接超时时间;**

**#define TimeOut 5000 // ms;**

**// 授时网站;**

**#define GetSysTimeUrl** [**http://quan.suning.com/getSysTime.do**](http://quan.suning.com/getSysTime.do)

**// rivest\_cipher\_4.hpp------------------------------------------------------------------------------------------------**

**/\*\***

**\* @file rivest\_cipher\_4.hpp**

**\* @date 05.04.2023**

**\*/**

**#pragma once**

**#include <array>**

**#include <cstring>**

**#include <iomanip>**

**#include <vector>**

**class RivestCipher4 {**

**public:**

**// 构造函数，接收一个字符串 key 作为参数**

**explicit RivestCipher4(const std::string &key) {**

**// 初始化 S 盒**

**for (uint16\_t i = 0; i < 256; ++i) S\_box\_[i] = static\_cast<uint8\_t>(i);**

**// 对 S 盒进行置换**

**uint32\_t j = 0;**

**for (uint16\_t i = 0; i < 256; ++i) {**

**// 用循环的方式扩展 key**

**j = (j + S\_box\_[i] + key[i % key.size()]) & 0xFF;**

**uint8\_t tmp = S\_box\_[i];**

**S\_box\_[i] = S\_box\_[j];**

**S\_box\_[j] = tmp;**

**}**

**}**

**// 加密函数，接收一个字符串 plaintext 作为参数**

**std::string encrypt(const std::string &plaintext) {**

**// 初始化一个输出字符串流**

**std::ostringstream ciphertext\_stream;**

**// 使用十六进制输出，并用 0 填充未满两位的字节**

**ciphertext\_stream << std::hex << std::setfill('0');**

**// 初始化两个状态变量 i 和 j**

**uint8\_t i = 0, j = 0;**

**// 遍历明文字符串中的每个字节**

**for (size\_t index = 0; index < plaintext.size(); ++index) {**

**// 更新状态变量 i 和 j**

**i = (i + 1) & 0xFF;**

**j = (j + S\_box\_[i]) & 0xFF;**

**// 交换 S 盒中的两个元素**

**std::swap(S\_box\_[i], S\_box\_[j]);**

**// 计算 k 和 temp**

**uint8\_t t = (S\_box\_[i] + S\_box\_[j]) & 0xFF;**

**uint8\_t k = S\_box\_[t];**

**uint8\_t temp = static\_cast<uint8\_t>(plaintext[index]) ^ k;**

**// 将加密后的字节写入输出流**

**ciphertext\_stream << std::setw(2) << static\_cast<uint16\_t>(temp);**

**}**

**// 返回加密后的字符串**

**return ciphertext\_stream.str();**

**}**

**// 解密函数，调用加密函数实现**

**std::string decrypt(const std::string &ciphertext) { return encrypt(ciphertext); }**

**private:**

**// S 盒，使用 std::array 存储**

**std::array<uint8\_t, 256> S\_box\_;**

**};**

**// make\_ptr.hpp-------------------------------------------------------------------------------------------------------**

**#pragma once**

**#include <memory>**

**/\***

**用于创建一个智能指针对象:**

**这段代码定义了一个模板函数 make\_unique，用于创建并返回一个智能指针 unique\_ptr，其中模板参数 T 代表指向的类型，参数 Args 代表构造函数参数类型的列表。**

**函数的实现部分首先创建一个 unique\_ptr，指向一个通过 new 运算符创建的类型为 T 的对象，并传递构造函数所需的参数列表。 std::forward 用于将 args**

**转发给构造函数，确保构造函数接收到正确类型的参数。**

**由于返回的是 unique\_ptr，因此可以确保指针所有权的唯一性，避免内存泄漏的问题.**

**\*/**

**template <typename T, typename... Args>**

**std::unique\_ptr<T> make\_unique(Args&&... args) {**

**return std::unique\_ptr<T>(new T(std::forward<Args>(args)...));**

**}**

**template <typename T, typename... Args>**

**std::shared\_ptr<T> make\_shared(Args&&... args) {**

**return std::shared\_ptr<T>(new T(std::forward<Args>(args)...));**

**}**

**// fourier\_transform.hpp----------------------------------------------------------------------------------------------**

**/\*\***

**\* @file fourier\_transform.hpp**

**\* @date 05.04.2023**

**\* @author RMSHE**

**\*/**

**#pragma once**

**#include <algorithm>**

**#include <cmath>**

**#include <complex>**

**#include <vector>**

**class FastFourierTransform {**

**public:**

**/\*\***

**\* @brief 快速傅里叶变换**

**\* @param input\_sequence std::vector<float>类型实数域数据**

**\* @param inverse false 为正变换, true 为反变换(默认为正变换)**

**\*/**

**std::vector<std::complex<float>> FFT(std::vector<float> input\_sequence, bool inverse = false) {**

**/\***

**\* 首先，n & (n - 1) 的结果为0，当且仅当n是2的幂次方。如果n是2的幂次方，则m直接取n，不需要进行补零或截断操作。**

**\* 否则，m需要取离n最近的较小的2的幂次方或较大的2的幂次方。**

**\* 使用 builtin\_clz 函数来计算n的二进制表示中前导零的个数，然后通过移位运算来得到最接近的2的幂次方。**

**\* 最后，使用三目运算符来判断是否需要进行补零或截断操作，直接resize到m即可。**

**\*/**

**uint32\_t n = input\_sequence.size();**

**uint32\_t m = n & (n - 1) ? 1 << (32 - CountLeadingZeros(n)) : n;**

**input\_sequence.resize(std::min(m, n));**

**// 这里将实数域转为复数域;**

**std::vector<std::complex<float>> data(input\_sequence.begin(), input\_sequence.end());**

**// 若反转为 false 则进行FFT正变换, 若反转为 true 则进行FFT逆变换;**

**inverse == false ? fast\_fourier\_transform(data) : inverse\_fast\_fourier\_transform(data);**

**return data;**

**}**

**private:**

**// 计算一个32位无符号整数的二进制表示中前导零的个数;**

**static uint32\_t CountLeadingZeros(uint32\_t x) {**

**uint32\_t n = 0;**

**if (x == 0) return 32;**

**if (x <= 0x0000FFFF) n += 16, x <<= 16;**

**if (x <= 0x00FFFFFF) n += 8, x <<= 8;**

**if (x <= 0x0FFFFFFF) n += 4, x <<= 4;**

**if (x <= 0x3FFFFFFF) n += 2, x <<= 2;**

**if (x <= 0x7FFFFFFF) n += 1;**

**return n;**

**}**

**// 快速傅里叶变换**

**static void fast\_fourier\_transform(std::vector<std::complex<float>> &input\_sequence) {**

**const uint32\_t sequence\_size = input\_sequence.size();**

**// 如果序列长度小于等于1，则不需要继续计算**

**if (sequence\_size <= 1) return;**

**// 缓存计算结果，避免重复计算**

**std::vector<std::complex<float>> even\_sequence(sequence\_size / 2), odd\_sequence(sequence\_size / 2);**

**for (uint32\_t i = 0; i < sequence\_size / 2; ++i) {**

**even\_sequence[i] = input\_sequence[2 \* i]; // 偶数序列**

**odd\_sequence[i] = input\_sequence[2 \* i + 1]; // 奇数序列**

**}**

**// 递归计算偶数和奇数序列的FFT**

**fast\_fourier\_transform(even\_sequence);**

**fast\_fourier\_transform(odd\_sequence);**

**// 合并偶数和奇数序列的FFT结果**

**const auto twiddle\_factor = std::polar(1.0, -2 \* PI / sequence\_size);**

**std::complex<float> factor(1, 0);**

**for (uint32\_t i = 0; i < sequence\_size / 2; ++i) {**

**input\_sequence[i] = even\_sequence[i] + factor \* odd\_sequence[i];**

**input\_sequence[i + sequence\_size / 2] = even\_sequence[i] - factor \* odd\_sequence[i];**

**factor \*= twiddle\_factor; // 更新旋转因子**

**}**

**}**

**// 快速傅里叶逆变换**

**static void inverse\_fast\_fourier\_transform(std::vector<std::complex<float>> &input\_sequence) {**

**// 先计算FFT，然后对结果取共轭并除以n即可得到逆变换的结果**

**fast\_fourier\_transform(input\_sequence);**

**const uint32\_t sequence\_size = input\_sequence.size();**

**for (auto &element : input\_sequence) {**

**element = std::conj(element) \* static\_cast<float>(1.0 / sequence\_size); // 取共轭并除以n**

**}**

**}**

**};**

**class DiscreteFourierTransform {**

**public:**

**std::vector<std::complex<float>> DFT(std::vector<float> input\_sequence, bool inverse = false) {**

**// 这里将实数域转为复数域;**

**std::vector<std::complex<float>> data(input\_sequence.begin(), input\_sequence.end());**

**// 若反转为 false 则进行DFT正变换, 若反转为 true 则进行DFT逆变换;**

**if (inverse == false)**

**return discrete\_fourier\_transform(data);**

**else**

**return inverse\_discrete\_fourier\_transform(data);**

**}**

**private:**

**// 计算离散傅里叶变换**

**std::vector<std::complex<float>> discrete\_fourier\_transform(const std::vector<std::complex<float>> &input\_vector) const {**

**uint32\_t N = input\_vector.size();**

**std::vector<std::complex<float>> output\_vector(N);**

**for (uint32\_t k = 0; k < N; k++) {**

**// 对于每个频率 k，计算对应的和**

**for (uint32\_t n = 0; n < N; n++) {**

**// 对于每个样本 n，计算频率为 k 的分量**

**output\_vector[k] += input\_vector[n] \* std::exp(std::complex<float>(0, -2 \* M\_PI \* k \* n / N));**

**}**

**}**

**return output\_vector;**

**}**

**// 计算离散傅里叶逆变换**

**std::vector<std::complex<float>> inverse\_discrete\_fourier\_transform(const std::vector<std::complex<float>> &input\_vector) const {**

**uint32\_t N = input\_vector.size();**

**std::vector<std::complex<float>> output\_vector(N);**

**for (uint32\_t n = 0; n < N; n++) {**

**// 对于每个样本 n，计算对应的和**

**for (uint32\_t k = 0; k < N; k++) {**

**// 对于每个频率 k，计算样本为 n 的分量**

**output\_vector[n] += input\_vector[k] \* std::exp(std::complex<float>(0, 2 \* M\_PI \* k \* n / N));**

**}**

**// 对和进行缩放，得到正确的输出**

**output\_vector[n] /= N;**

**}**

**return output\_vector;**

**}**

**};**

**// tree.hpp-----------------------------------------------------------------------------------------------------------**

**/\*\***

**\* @file tree.hpp**

**\* @date 26.02.2023**

**\* @author RMSHE**

**\*/**

**/\***

**这段代码实现了一个通用的树数据结构，包括节点的添加、删除、查找、遍历等基本操作。**

**该树数据结构由两个类组成：TreeNode和Tree。TreeNode表示一个树节点，包含节点数据、子节点列表、父节点等属性；Tree表示整个树结构，**

**包含根节点以及树的遍历、节点查找、节点删除等操作。**

**具体来说，TreeNode类包含了添加子节点、查找子节点、删除子节点等方法。其中，添加子节点使用了C++11中的智能指针std::unique\_ptr，**

**确保了子节点的内存管理安全；查找子节点使用了递归的方式，深度优先遍历整个子树；**

**删除子节点则使用了迭代的方式，遍历整个子树进行删除操作。**

**Tree类包含了树的遍历、节点查找、节点删除等方法。其中，遍历操作分为深度优先遍历和广度优先遍历两种方式；**

**节点查找操作同样使用了递归的方式，在根节点开始向下搜索整个子树；节点删除操作则使用了递归的方式，在整个子树中进行删除操作。**

**总的来说，该树数据结构提供了基本的树操作，能够满足一些基本的需求。但是需要注意的是，该树数据结构没有进行任何的平衡操作，因此对于较大的树可能会存在效率问题。**

**\*/**

**#pragma once**

**#include <algorithm>**

**#include <make\_ptr.hpp>**

**#include <memory>**

**#include <queue>**

**#include <unordered\_map>**

**#include <vector>**

**// @note 一个 TreeNode 对象代表了一棵树中的一个节点，其中包含了当前节点的数据和指向它的父节点的指针以及指向其子节点的所有指针。**

**template <typename T>**

**class TreeNode {**

**public:**

**T node\_data; // 储存这个节点的值**

**std::vector<std::unique\_ptr<TreeNode<T>>> children; // 储存指向这个节点的子节点的指针**

**TreeNode<T>\* parent; // 储存指向这个节点的父节点的指针**

**/\*\***

**\* @brief "TreeNode"树节点构造函数: 创建一个新的节点对象，构造节点.**

**\* @param data const T&类型的参数，表示根节点的数据(data的数据类型可任意).**

**\* @param parent\_node\_ptr TreeNode<T>\* 类型的参数，表示指向父节点的指针, 默认为nullptr.**

**\* @note 用法：TreeNode< std::string > node("data", parent\_node\_ptr);**

**\*/**

**TreeNode(const T& data, TreeNode<T>\* parent\_node\_ptr = nullptr) : node\_data(data), parent(parent\_node\_ptr) {}**

**/\*\***

**\* @brief 向当前节点添加一个子节点**

**\* @param data const T&类型的参数，表示节点的值.**

**\* @return TreeNode<T>\* 返回一个指向新加子节点的指针**

**\* @note 当调用 addChild() 函数时，它将创建一个新的 TreeNode 对象，该对象保存传递给函数的数据，并将指向新创建节点的指针添加到当前节点的 children**

**\* 向量中。也就是说，addChild() 添加的是一个新的子节点。使用示例：parent\_node\_ptr->addChild(data) / parent\_node\_ptr->addChild(data0)->addChild(data1);**

**\*/**

**TreeNode<T>\* addChild(const T& data) {**

**// 为类分配内存并创建对象时会自动调用类的构造函数TreeNode(const T& data, TreeNode<T>\* parent\_node\_ptr = nullptr);**

**// parent\_node\_ptr->addChild(data); 在这个语句中 this 即是 parent\_node\_ptr;**

**children.emplace\_back(make\_unique<TreeNode>(data, this)); // 向父节点添加一个指向子节点的指针；**

**return this->findChild(data); // 返回一个指向刚刚添加的子节点的指针**

**}**

**/\*\***

**\* @brief 在当前节点的子节点中查找指定数据的节点**

**\* @param target\_child\_data const T&类型的参数，表示要查找的节点数据(值).**

**\* @return TreeNode<T>\* 指向查找到的节点的指针，如果未找到返回 nullptr.**

**\* @note 使用示例：parent\_node\_ptr->findChild(target\_child\_data);**

**\*/**

**TreeNode<T>\* findChild(const T& target\_child\_data) {**

**// 遍历当前节点的每一个子节点**

**for (auto& child : children) {**

**if (child->node\_data == target\_child\_data) {**

**// 如果当前子节点的数据等于要查找的数据，则返回该子节点的指针。**

**return child.get();**

**}**

**}**

**return nullptr; // 如果遍历完所有子节点都没有找到，则返回 nullptr 表示没有找到该节点。**

**}**

**/\*\***

**\* @brief 在当前节点的后裔中查找一个指定数据值的节点.**

**\* @param target\_node\_data const T&类型的参数，表示要查找的节点数据(值).**

**\* @return TreeNode<T>\* 指向查找到的节点的指针，如果未找到返回 nullptr.**

**\* @note 使用示例：parent\_node\_ptr->findDescendant(target\_node\_data);**

**\*/**

**TreeNode<T>\* findDescendant(const T& target\_node\_data) {**

**// 遍历当前节点的每一个子节点**

**for (auto& child : children) {**

**if (child->node\_data == target\_node\_data) {**

**// 如果当前子节点的数据等于要查找的数据，则返回该子节点的指针。**

**return child.get();**

**} else {**

**// 否则，递归地调用子节点的 findDescendant**

**// 方法来查找是否存在指定数据的节点，如果找到，则返回该子节点的指针。这是一个深度优先的递归遍历方式。**

**TreeNode<T>\* found = child->findDescendant(target\_node\_data);**

**if (found != nullptr) return found;**

**}**

**}**

**return nullptr; // 如果遍历完所有子节点都没有找到，则返回 nullptr 表示没有找到该节点。**

**}**

**/\*\***

**\* @brief 判断当前节点的一个子节点是否有孩子.**

**\* @param node\_ptr TreeNode<T>\* 待判断节点的指针**

**\* @return 如果存在子节点返回true, 否则返回false**

**\* @note 使用方法: node\_ptr->hasChildren();**

**\*/**

**bool hasChildren() const {**

**return !this->children.empty(); // 如果 node\_ptr->children 不为空，则表示这个节点有子节点。**

**}**

**/\*\***

**\* @brief 删除父节点的一个无孩子子节点(删除树叶节点)**

**\* @param target\_child\_data const T& 这里需要提供待删除的目标子节点的值**

**\* @return 删除成功返回true，否则返回false**

**\* @note 注意这个函数只支持删除没有子节点的节点，即树枝的末端(树叶)。**

**\* 若要删除整颗树或部分树枝请使用"deleteNode();"函数**

**\* 使用示例：parent\_node\_ptr->deleteChild(target\_child\_data);**

**\*/**

**bool deleteChild(const T& target\_child\_data) {**

**TreeNode<T>\* child\_node\_ptr = findChild(target\_child\_data); // 从当前父节点查找要删除的子节点的指针**

**// 判断这个子节点是否也存在子节点，这里只支持删除没有子节点的节点(树叶), 如果存在子节点或找不到要删除的子节点，返回 false**

**if (child\_node\_ptr == nullptr || hasChildren(child\_node\_ptr) == true) return false;**

**// 遍历当前父节点的所有子节点，在父节点中删除要删除的节点**

**for (auto it = children.begin(); it != children.end(); ++it) {**

**// 从迭代器获取子节点的指针，如果该指针是要删除的目标子节点则删除它，**

**// 由于使用了 std::unique\_ptr来管理子节点，所以父节点可以在内存管理方面自动处理子节点的内存释放，不需要手动释放。**

**if (it->get() == child\_node\_ptr) {**

**children.erase(it); // 移除的目标子节点**

**break;**

**}**

**}**

**return true;**

**}**

**};**

**template <typename T>**

**class Tree {**

**public:**

**std::unique\_ptr<TreeNode<T>> root; // 储存树的根节点**

**TreeNode<T>\* current\_node\_ptr = root.get(); // 储存最后一次添加节点后的指针位置(初始化时设为根节点指针)**

**/\*\***

**\* @brief "Tree"树构造函数: 创建一个新的Tree对象，构造根节点.**

**\* @param data const T&类型的参数，表示根节点的数据.**

**\* @return void**

**\* @note 用法：Tree< std::string > tree0("root");**

**\*/**

**Tree(const T& data) : root(make\_unique<TreeNode<T>>(data)) {}**

**/\*\***

**\* @brief 向当前节点添加一个子节点**

**\* @param node\_ptr TreeNode<T>\*类型的参数(指向节点的指针)，表示在该节点下添加子节点.**

**\* @param data const T&类型的参数，表示要添加的节点的数据.**

**\* @return TreeNode<T>\* 返回一个指向刚刚添加的子节点的指针**

**\* @note 当调用 `addNode()` 函数时，它将创建一个新的 `TreeNode`**

**\* 对象，该对象保存传递给函数的数据，并将指向新创建节点的指针添加到当前节点的`children`向量中。也就是说，`addNode()` 添加的是一个新的子节点。`addNode`**

**\* 与 `addChild` 不同, `addNode` 是 `Tree class` 的成员, 而 `addChild` 是`TreeNode class` 的成员, `addNode` 将父节点指针作为参数传递.**

**\* 该函数还会将指向新增节点的指针保存到类成员变量 `current\_node\_ptr`中,以便用户更清楚当前树的编辑位置.**

**\*/**

**TreeNode<T>\* addNode(TreeNode<T>\* node\_ptr, const T& data) { return current\_node\_ptr = node\_ptr->addChild(data); }**

**/\*\***

**\* @brief 以深度优先的方式遍历树**

**\* @param node\_ptr TreeNode<T>\* 提供一个节点指针，函数会以该节点为根节点递归遍历它所有的子嗣节点(若不传参则默认遍历整颗树).**

**\* @return 返回一个向量, 其中包含从指定节点开始子树的所有节点数据值和对应的指针 std::vector<std::pair<T, TreeNode<T>\*>>**

**\* @note 深度优先遍历算法是递归的，它首先访问根节点，然后再递归地遍历每个子树。在每个节点访问完成后，递归函数回溯到其父节点继续遍历其他子树**

**\*/**

**std::vector<std::pair<T, TreeNode<T>\*>> traversalDFS(TreeNode<T>\* node\_ptr = nullptr) {**

**// 默认节点指针设置为根节点**

**if (node\_ptr == nullptr) node\_ptr = root.get();**

**// 如果节点为空，直接返回一个空向量**

**if (node\_ptr == nullptr) return {};**

**// 创建一个向量，用于存储当前节点和其子节点的数据**

**std::vector<std::pair<T, TreeNode<T>\*>> tree\_data;**

**// 将当前节点的数据插入到 tree\_data 向量的末尾**

**tree\_data.emplace\_back(std::pair<T, TreeNode<T>\*>(node\_ptr->node\_data, node\_ptr));**

**// 遍历当前节点的每个子节点**

**for (auto& child : node\_ptr->children) {**

**// 递归遍历当前子节点的子树，并将其存储在 deep\_tree\_data 向量中**

**auto deep\_tree\_data = traversalDFS(child.get());**

**// 将当前子节点的子树数据插入到 tree\_data 向量的末尾**

**tree\_data.insert(tree\_data.end(), deep\_tree\_data.begin(), deep\_tree\_data.end());**

**}**

**// 返回包含当前节点及其所有子节点的数据的向量**

**return tree\_data;**

**}**

**/\*\***

**\* @brief 以广度优先的方式遍历树。**

**\* @param node\_ptr TreeNode<T>\* 提供一个节点指针，函数会以该节点为根节点递归遍历所有的子节点(若不传参则默认遍历整颗树).**

**\* @return 返回一个向量, 其中包含从指定节点开始子树的所有节点数据值和对应的指针 std::vector<std::pair<T, TreeNode<T>\*>>**

**\* @note 广度优先遍历算法是按层遍历，从根节点开始，先遍历根节点，然后按照从左到右的顺序遍历其子节点，再依次遍历下一层的所有节点。**

**\*/**

**std::vector<std::pair<T, TreeNode<T>\*>> traversalBFS(TreeNode<T>\* node\_ptr = nullptr) {**

**// 默认节点指针设置为根节点**

**if (node\_ptr == nullptr) node\_ptr = root.get();**

**// 当节点为空时返回空向量**

**if (node\_ptr == nullptr) return {};**

**// 创建一个向量，用于存储当前节点和其子节点的数据**

**std::vector<std::pair<T, TreeNode<T>\*>> tree\_data;**

**// 定义队列，并将根节点推入队列**

**std::queue<TreeNode<T>\*> node\_queue;**

**node\_queue.push(node\_ptr);**

**// 遍历队列，直到队列为空**

**while (!node\_queue.empty()) {**

**// 取出队列头部的节点后将其弹出**

**auto current\_node = node\_queue.front();**

**node\_queue.pop();**

**// 将当前节点的数据值和指针插入到 tree\_data 向量的末尾**

**tree\_data.emplace\_back(std::pair<T, TreeNode<T>\*>(current\_node->node\_data, current\_node));**

**// 将当前节点的所有子节点推入队列**

**for (auto& child : current\_node->children) node\_queue.push(child.get());**

**}**

**// 返回包含当前节点及其所有子节点的数据的向量**

**return tree\_data;**

**}**

**/\*\***

**\* @brief 在树中查找指定数据的节点**

**\* @param target\_node\_data const T&类型的参数，表示要查找的节点数据(值).**

**\* @return TreeNode<T>\* 指向查找到的节点的指针，如果未找到返回 nullptr.**

**\* @note 使用示例：tree.findNode(target\_node\_data);**

**\*/**

**TreeNode<T>\* findNode(const T& target\_node\_data) {**

**// 在根节点下查找目标节点(查找范围不包含根节点).**

**TreeNode<T>\* node\_ptr = root->findDescendant(target\_node\_data);**

**/\***

**如果上一步的查找结果 node\_ptr 为空指针，则：**

**如果根节点的数据等于要查找目标数据，则我们可断定用户查找的是根节点直接(返回根节点的指针)，否则即在包含根节点的整颗树的范围内找不到目标节点(返回空指针)**

**否则：**

**在根节点下查找到目标节点, 直接返回查找到的节点的指针.**

**\*/**

**return node\_ptr == nullptr ? (root->node\_data == target\_node\_data ? root.get() : nullptr) : node\_ptr;**

**}**

**/\*\***

**\* @brief 判断指定节点是否存在子节点**

**\* @param node\_ptr TreeNode<T>\* 待判断节点的指针**

**\* @return 如果存在子节点返回true, 否则返回false**

**\*/**

**bool hasChildren(TreeNode<T>\* node\_ptr) const {**

**return !node\_ptr->children.empty(); // 如果 node\_ptr->children 不为空，则表示这个节点有子节点。**

**}**

**/\*\***

**\* @brief 递归地计算树的深度(高度)(默认统计整颗树的深度)**

**\* @param node\_ptr TreeNode<T>\*类型的参数(指向节点的指针)，表示从该节点开始统计树枝的深度，若设为root则为统计整颗树的深度(这也是无传参时的默认设置)**

**\* @return uint32\_t 返回树的深度.**

**\* @note 使用示例：1.统计整颗树的深度：tree.getHeight(); 2.统计从 node1 节点开始的树枝深度：tree.getHeight(node1\_ptr);**

**\*/**

**uint32\_t getDepth(TreeNode<T>\* node\_ptr = nullptr) {**

**// 默认节点指针设置为根节点**

**if (node\_ptr == nullptr) node\_ptr = root.get();**

**// 如果节点没有子节点，说明当前节点是叶子节点，返回 1 作为高度**

**if (hasChildren(node\_ptr) == false) return 1;**

**uint32\_t max\_depth = 0; // 定义最大深度（一个树中最少有一个树枝最长）；**

**// 如果节点有子节点，则递归计算子节点的高度，并找到其中最大的高度**

**for (const auto& child : node\_ptr->children) max\_depth = std::max(max\_depth, getDepth(child.get()));**

**// 返回最大高度加上 1，即为整个树的高度**

**return max\_depth + 1;**

**}**

**/\*\***

**\* @brief 获取节点的度(对于一个给定的节点，其子节点的数量称为度. 一个叶子的度数一定是零)**

**\* @param node\_ptr TreeNode<T>\*类型的参数(指向节点的指针)，表示获取该节点的子节点的个数，无传参时默认获取根节点的度.**

**\* @return uint32\_t 返回目标节点的子节点的个数.**

**\*/**

**uint32\_t getDegree(TreeNode<T>\* node\_ptr = nullptr) {**

**// 默认节点指针设置为根节点**

**if (node\_ptr == nullptr) node\_ptr = root.get();**

**// 直接返回目标节点的子节点个数；**

**return node\_ptr->children.size();**

**}**

**/\*\***

**\* @brief 获取树的度(树的度是指树中一个节点的最大度, 即树中某个拥有最多子节点的父节点的子节点数)**

**\* @return 返回树的度**

**\*/**

**uint32\_t get\_degree\_of\_tree() {**

**uint32\_t max\_degree = 0; // 初始化最大度为0**

**// 对树进行深度优先遍历获取所有节点的子节点个数，这里将当前最大值max\_degree与遍历到的父节点的子节点个数进行比较后取较大值更新回max\_degree中.**

**for (auto& node : traversalDFS()) max\_degree = std::max(max\_degree, node.second->children.size());**

**// 遍历完成后返回树中所有结点的度的最大值**

**return max\_degree;**

**}**

**/\*\***

**\* @brief 获取树或树枝的叶子数量(叶子即没有子节点的节点，也称做终端节点)**

**\* @param node\_ptr TreeNode<T>\*类型的参数(指向节点的指针)，表示获取以该节点为起点的树枝的叶子个数，无传参时默认获取整个树的叶子数量.**

**\* @return 返回树或指定树枝的叶子数量**

**\*/**

**uint32\_t getBreadth(TreeNode<T>\* node\_ptr = nullptr) {**

**// 默认节点指针设置为根节点**

**if (node\_ptr == nullptr) node\_ptr = root.get();**

**uint32\_t num\_leaves = 0; // 初始化叶子数为0**

**// 对树进行深度优先遍历;**

**for (auto& node : traversalDFS(node\_ptr))**

**// 如果一个节点没有子节点，则增加叶子数.**

**if (hasChildren(node.second) == false) ++num\_leaves;**

**// 返回树的叶子数量**

**return num\_leaves;**

**}**

**/\*\***

**\* @brief 获取树的宽度或指定节点所在层的宽度(宽度指一个层的节点数)**

**\* @param node\_ptr TreeNode<T>\*类型的参数(指向节点的指针)，表示获取该节点所在层的宽度，无传参时默认获取整个树的宽度(拥有最大宽度的层级).**

**\* @return 无参数时返回树的宽度，有参数时返回参数节点所在层的宽度.**

**\*/**

**uint32\_t getWidth(TreeNode<T>\* node\_ptr = nullptr) {**

**// 如果没有传递指针，则默认使用根节点指针。**

**if (node\_ptr == nullptr) node\_ptr = root.get();**

**uint32\_t max\_level\_width = 0; // 用于储存树的宽度(拥有最大宽度的层级)**

**std::unordered\_map<uint32\_t, uint32\_t> level\_widths; // 用于存储每个层级的节点数的哈希表**

**// 使用广度优先搜索遍历每个节点**

**for (auto& node : traversalBFS(node\_ptr)) {**

**uint32\_t level = static\_cast<uint32\_t>(getLevel(node.second)); // 获取当前节点的层级**

**auto iter = level\_widths.find(level); // 查找当前层级是否已经在哈希表中存在**

**if (iter != level\_widths.end()) {**

**++iter->second; // 如果存在，就增加该层级的节点数**

**max\_level\_width = std::max(max\_level\_width, iter->second); // 更新树的宽度**

**} else**

**level\_widths.insert({level, 1}); // 如果不存在，就将该层级的节点数设置为1**

**}**

**// 如果传入的指针是根节点指针或空指针，则返回树的宽度；否则返回该节点所在层级的节点数**

**return node\_ptr == root.get() ? max\_level\_width : level\_widths.find(getLevel(node\_ptr))->second;**

**}**

**/\*\***

**\* @brief 获取指定节点的层级(一个节点的层级是它与根节点之间唯一路径上的边的数量, 根节点层级为零)**

**\* @param node\_ptr TreeNode<T>\*类型的参数(指向节点的指针)，表示获取该节点的层级.**

**\* @return 返回指定节点的所在层级数, 如果提供的节点指针为空指针则返回-1**

**\*/**

**int32\_t getLevel(TreeNode<T>\* node\_ptr) {**

**// 当节点为空时返回错误信息**

**if (node\_ptr == nullptr) return -1;**

**int32\_t level = 0; // 初始化节点层级为0**

**// 从提供的节点 node\_ptr 开始向根节点查找, 在每一次循环中增加 level 的数量, 直到遍历到根节点(根节点没有父节点)**

**while (node\_ptr->parent != nullptr) {**

**level++;**

**node\_ptr = node\_ptr->parent;**

**}**

**return level;**

**}**

**/\*\***

**\* @brief 获取树或树枝的大小(节点数)**

**\* @param node\_ptr TreeNode<T>\*类型的参数(指向节点的指针)，表示获取以该节点为起点的树枝的节点总数，无传参时默认获取整个树的节点总数.**

**\* @return 返回树或指定树枝的节点总数**

**\*/**

**uint32\_t getSize(TreeNode<T>\* node\_ptr = nullptr) {**

**// 默认节点指针设置为根节点**

**if (node\_ptr == nullptr) node\_ptr = root.get();**

**// 对树进行深度优先遍历然后返回树中的节点数;**

**return traversalDFS(node\_ptr).size();**

**}**

**/\*\***

**\* @brief 判断树是否为空**

**\* @return 如果树为空，则返回 true；如果向量不为空，则返回 false。**

**\*/**

**bool empty() { return root == nullptr ? true : false; }**

**/\*\***

**\* @brief 递归删除一个节点及其后裔**

**\* @param node\_ptr TreeNode<T>\* 这里需要提供指向待删除节点的指针**

**\* @return 删除成功返回true，否则返回false**

**\* @note 使用示例：tree.deleteNode(node\_1);**

**\*/**

**bool deleteNode(TreeNode<T>\* node\_ptr) {**

**// 如果节点为空，直接返回**

**if (node\_ptr == nullptr) return false;**

**// 获取该节点的所有子节点，并遍历删除它们**

**auto& children = node\_ptr->children;**

**for (auto it = children.begin(); it != children.end();) {**

**auto& child = \*it; // it 是迭代器, \*it 是迭代器所指的内容.**

**// 这里是在判断一个节点是否有孩子.**

**if (hasChildren(child.get()) == true)**

**deleteNode(child.get()); // 如果子节点不是树叶节点(有孩子)，则递归调用 deleteNode**

**else**

**it = children.erase(it); // 移除树叶节点(如果删除成功，该函数会返回指向被删除元素之后的元素的迭代器)**

**}**

**// 检查当前节点node\_ptr是否为根节点，如果是，则返回 false, 这里无法删除根节点.**

**if (node\_ptr->parent == nullptr) return false;**

**// 在删除完这个节点的孩子后, 删除它自身**

**auto parent = node\_ptr->parent; // 获取当前节点的父节点指针**

**// 在父节点的子节点列表中查找并移除当前节点**

**parent->children.erase(**

**// 在容器中查找符合某个条件的元素，并将其移动到容器的末尾**

**std::remove\_if(parent->children.begin(), // 查找的起始位置**

**parent->children.end(), // 查找的终止位置**

**// 查找谓词，用于确定哪些元素符合要求。该函数或函数对象接受一个元素作为参数.**

**// remove\_if算法会将child传入这个匿名函数, 如果 child指针与node\_ptr相同则返回 true.**

**[node\_ptr](std::unique\_ptr<TreeNode<T>>& child) { return child.get() == node\_ptr; }),**

**parent->children.end() // 移除的终止位置**

**);**

**return true;**

**}**

**/\*\***

**\* @brief 移除一颗树.**

**\* @note 使用示例：tree.deleteTree();**

**\*/**

**void deleteTree() {**

**deleteNode(root.get()); // 删除根节点的所有子嗣节点;**

**root.reset(); // 移除根节点(将根节点重置为nullptr);**

**}**

**// Tree的析构函数**

**~Tree() { deleteTree(); }**

**};**