PROJECT REPORT

ON

Scrolling Display Using Neopixel LED matrix

SUBMITTED BY

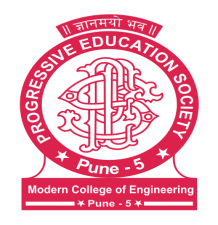
Omkar Kashid

Madhuri Mahale

Shilpa Sanap

Under the Guidance of

Mr. Ramgopal Sahu





DEPARTMENT OF ELECTRONICS & TELECOMMUNICATION

P.E.S’S MODERN COLLEGE OF ENGINEERING

PUNE – 411 005.



SAVITRIBAI PHULE PUNE UNIVERSITY

Academic Year: 2021-22

CERTIFICATE

This is to certify that

**Omkar Kashid Exam No. T190313064**

**Madhuri Mahale Exam No. T190313085**

**Shilpa Sanap Exam No. T190313122**

Of T.E. E&TC have successfully completed the Project titled **Scrolling Display Using** **Neopixel LED Matrix**’during the academic year 2021-22 for the Course **Mini Project**. This report is submitted as per the requirement prescribed by Savitribai Phule Pune University.

**Dr. Mrs. R. S. Kamathe Mr.Ramgopal Sahu**

**H.O.D. (E&TC) Project Guide**

**ACKNOWLEDGEMENT**

In performing our project, we had to take the help and guideline of some respected persons, who deserve our greatest gratitude. The completion of this project gives us much Pleasure. We could like to show our gratitude towards **Mr. Ramgopal Sahu** Sir for giving us guideline for project throughout numerous consultations.

We would also like to expand our deepest towards our HOD **Dr.Mrs. R.S. Kamathe**. for guiding us.In addition, a thank you to our industry experts who introduced us to the Methodology of work, and whose passion for the “underlying sturctures” had lasting effect.

Many people, especially our classmates and team members itself, have made valuable comment suggestions on this project which gave us an inspiration to improve our project. We thank all the people for their help directly and indirectly to complete our project.

Omkar Kashid

Madhuri Mahale

Shilpa Sanap

**ABSTRACT**

The LED Display System is aimed at the colleges and universities for displaying day-to-day information continuously or at regular intervals during the working hours. It offers flexibility to display flash news or announcements neo pixel based display system can also be used at other public places like schools, hospitals, railway stations, gardens etc. without affecting the surrounding environment. The LED display system mainly consists of a receiver and a display toolkit which can be programmed from an Boot Loader platform. It receives the message, through serial port and displays the desired information after necessary code conversion.

It can serve as an electronic notice board and display the important notices instantaneously thus avoiding the latency. Being modular design, the LED display is easy to expand and allows the user to add more display units at any time and at any location in the campus depending on the requirement of the institute

**TABLE OF CONTENTS**

Sr. Page No.

1. Introduction 6
2. Literature Survey 7
3. System Specifications 8
4. Block diagram & description 9
5. Hardware System Design 12
6. Software System Design 19
7. Results and Discussion 20
8. Bill of Material 30
9. Applications & Future modifications 31
10. References 33
11. Data Sheets. 34
12. **Introduction**

Display advertising plays a very importing role in marketing and there are several advertisement methods like newspapers, posters, glow signboards, etc. but digital LED display boards are getting popular nowadays because of their reliability and advantages.

Although they are a little bit expensive still, they are durable and customizable, like the advertising text can be changed easily whenever needed and they can also be used as Digital Notice Board at any public place. LED message scrolling displays are very popular. These displays are used in shopping malls, theaters, schools, traffic signs, public transportation etc.

This project includes LED strip of 300 LED to control the text displayed over it and Scrolling LED display is implemented by using AVR microcontroller. LED display system is aimed at the colleges and universities for displaying day today information continuously or at regular intervals during the working hours. Being Scrolling LED system, it offers flexibility to display flash news or announcements faster than the programmable system.

The LED display system mainly consists of a Usb to TTL Converter and a display board which can be programmed through microcontroller . It receives the message through serial port and display the desired information after necessary code conversion. It can serve as an electronic notice board and display the important notices without any delay thus avoiding the latency. The LED display is easy to expand and it allows the user to add more displays at any time and at any location depending on the requirement.

By using microcontroller it is possible to change the message in the LED neopixel display from anywhere in the world. .The display side with Atmega328P microcontroller to send the message on LED neopixel display board. Along with these a power supply unit and supporting hardware for microcontroller is used.

1. **Literature Survey**

Prof. Simha Shreya Chethan Kumar, Parinitha C, Shashidhar Tantry (Department ofElectronics and Communication Enggineering, PES Institute Of Technology, Banglore College of Engineering Belagavi, India)In this paper simha, it can beeasily integrated with general purpose display board to provide its mobility. The system acept the message from of SMS and display on the notice board [1]

Neeraj Khera, Divya Shukla, Shambhavi Awasthi In this paperthe technological advancement of the notice board is purposed that will help to save time and resources.Also it makes the information available fast to the person. [2]

Author: ForamKamdar, AnubbhavMalhotra and Pritish Mahadik. This paper deals with an SMS based notice board incorporating the widely used GSM to facilitatethe  communication of displaying message on notice board via user's mobilephone. Its operation is based on  microcontroller ATMEGA32 programmed in assembly language. ASIM300 GSM modem with a SIM card is  interfaced to the ports of the microcontroller with the help of AT commands. When the user sends a SMS via a  registered number from his mobile phone, it is received by SIM300 GSM modem at the receiver send.  [3]

1. **Specifications**
   * Neo WS28128B :- LED Matrix
   * 256 Brightness Display
   * 16777216 Color full display
   * Scan frequency :-400Hz/s
   * Power supply:- 6.0 A , 12V
   * ATMEGA 328:- Prosser
   * DS1307:- Real Time Clock IC
   * USB to TTL Converter
   * PCB Board
2. **Block Diagram & Description**

**Neopixel LED matrix**

**USB to TTL converter**

**Input message source (pc)**

**ATMEGA328**

**Power supply**

Fig. 4.1 Block Diagram

* 1. Atmega328

Atmega328is used as a microcontroller, working as a main component handling the whole project.

The **Atmega328 i**s a very commonly used microcontroller board created by the Atmel. It can support data up to eight bits and has a flash memory of thirty-two-kilo bytes.

4.2NeoPixel LED

NeoPixel LED is the displaying component, displays the information given by the input source and the microcontroller.

The code loaded in microcontroller gets displayed, the NeoPixel LED is RGB LED strip displays different shapes and colours.

4.3 Input Message Sorce (PC)

The input message source, is the medium to give the inptut code to the microcontroller.

4.4 USB To TTL converter

The AN-USB-TTL module is a effective way to convert TTL signal a USB interface. When connected to a PC USB port the AN-USB-TTL module is automatically detected and is installed as a native COM port which is compatible with any existing serial communication application

4.5DC Power supply

The Dc power supply supplies 5v Power to the Micrcontroller and the other component to function accordingly.

1. **Hardware System Design**

**5.1 Atmega 328**

High performance, low power AVR 8-bit microcontroller The Atmel ATmega328P is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega328P achieves throughputs approaching 1MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

The high-performance Microchip 8-bit AVR® RISC-based microcontroller combines 32 KB ISP Flash memory with read-while-write capabilities, 1 KB EEPROM, 2 KB SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible timer/counters with compare modes, internal and external interrupts, serial programmable USART, a byte-oriented Two-Wire serial interface, SPI serial port, 6-channel 10-bit A/D converter (8-channels in TQFP and QFN/MLF packages), programmable watchdog timer with internal oscillator, and five software selectable power saving modes. The device operates between 1.8-5.5 volts.

By executing powerful instructions in a single clock cycle, the device achieves throughputs approaching one MIPS per MHz, balancing power consumption and processing speed.

Peripheral features

● Two 8-bit Timer/Counters with separate prescaler and compare mode

● One 16-bit Timer/Counter with separate prescaler, compare mode, and capture mode

● Real time counter with separate oscillator

● Six PWM channels

● 8-channel 10-bit ADC in TQFP and QFN/MLF package

● Temperature measurement

● Programmable serial USART

● Master/slave SPI serial interface

● Byte-oriented 2-wire serial interface (Phillips I2 C compatible)

● Programmable watchdog timer with separate on-chip oscillator

● On-chip analog comparator Interrupt and wake-up on pin change

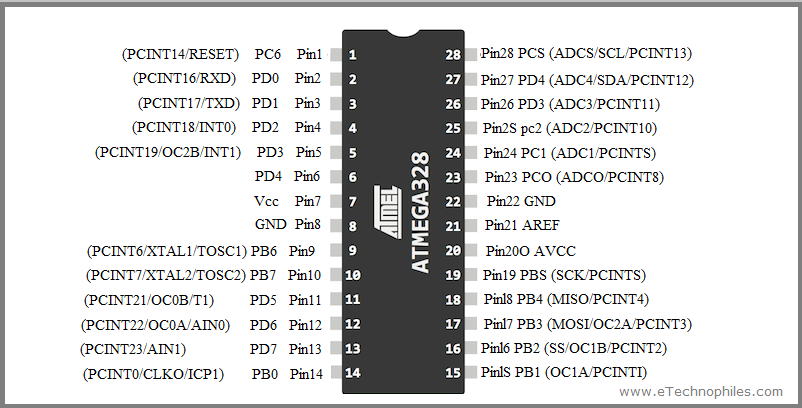
****

Fig. 5.1 Atmega 328

**5.2 NeoPixel LED**

Over the course of a few years, **RGB LEDs** are getting popular day by day due to its beautiful color, brightness, and enticing lighting effects. That is why It is used in many places as a **decorative item**, an example can be the home or an office space. Also, we can use the RGB lights in the kitchen and also in a gaming console. They’re also great in a kid’s playroom or bedrooms in terms of **mood lighting.** Previously, We used the WS2812B NeoPixel LEDs and the [ARM Microcontroller](https://circuitdigest.com/tags/arm-microcontroller) to build a [Music Spectrum Visualizer](https://circuitdigest.com/microcontroller-projects/led-music-spectrum-using-neopixels-arm-controller), so do check that out if that’s interesting to you.

The Arduino compatible **NeoPixel Shield**contains forty individually addressable RGB LEDs each one has the WS2812b driver built-in, which is arranged in a 5×8 matrix to form this **NeoPixel Shield**. Multiple NeoPixel Shields can also be connected to form a larger Shield if that's a requirement. To control the RGB LEDs, a single Arduino pin is required, so in this tutorial, we have decided to use pin 6 of the Arduino to do so.

In the project, the LEDs are powered from the Arduino’s inbuilt 5V pin, which is sufficient for powering about “a third of the LEDs” at full brightness. If you need to power more LEDs, then you can cut the inbuilt trace and use an external 5v supply to power the shield using the External 5V terminal.

****

Fig 5.2 Neopixel LED

**5.3 DS1307**

The DS1307 serial real-time clock (RTC) is a lowpower, full binary-coded decimal (BCD) clock/calendar plus 56 bytes of NV SRAM. Address and data are transferred serially through an I2 C, bidirectional bus. The clock/calendar provides seconds, minutes, hours, day, date, month, and year information. The end of the month date is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12- hour format with AM/PM indicator. The DS1307 has a built-in power-sense circuit that detects power failures and automatically switches to the backup supply. Timekeeping operation continues while the part operates from the backup supply

**5.4 BENEFITS AND FEATURES**

**1.** Completely Manages All Timekeeping Functions o Real-Time Clock Counts Seconds, Minutes, Hours, Date of the Month, Month, Day of the Week, and Year with Leap-Year Compensation Valid Up to 2100 o 56-Byte, Battery-Backed, General-Purpose RAM with Unlimited Writes o Programmable Square-Wave Output Signal

**2.** Simple Serial Port Interfaces to Most Microcontrollers o I 2 C Serial Interface

Low Power Operation Extends Battery Backup Run Time o Consumes Less than 500nA in BatteryBackup Mode with Oscillator Running o Automatic Power-Fail Detect and Switch Circuitry

**3.** 8-Pin DIP and 8-Pin SO Minimizes Required Space

**4.** Optional Industrial Temperature Range: -40°C to +85°C Supports Operation in a Wide Range of Applications Underwriters Laboratories (UL) Recognized

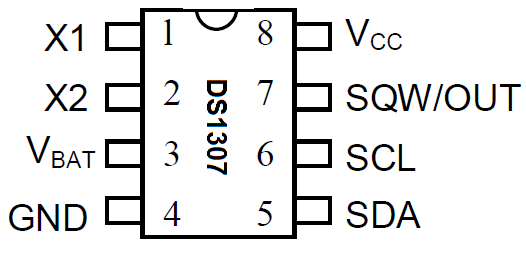


Fig 5.3 DS1307

**5.4 USB To TTL Converter**

This is USB To Serial Adapter Module USB TO TTL RS232 Arduino Cable With CTS RTS. USB To RS232 TTL PL2303HX Auto Converter Module Adapter 5V Output provides the best and convenient way to connect your RS232 TTL devices or DEMO BRD. To your computer, via the USB port.

FTDI based USB to TTL Serial Cable is designed using the standard FT232RL chipset. The cables provide a fast, simple way to connect devices with a TTL level serial interface to USB.

Pin Connections:

*Red* 5V  
*Black* GND  
*White* RXD  
*Green* TXD



Fig 5.4 USB To TTL Converter

**Features:**

1. Built-in USB to TTL Transfer chip.
2. Designed to be used for USB to TTL electronic projects.
3. TTL interface output, easy to connect to your MCU.
4. Dual 3.3V and 5V Power output, work with 3.3v and 5 V target device.
5. The mini-module is designed specifically for STC download and ARDUINO PRO supports all series of STC
6. Microcomputer with cold start reset button
7. Supports WIN7/VISTA/MAC/LINUX (32 bit /64-bit system)
8. USB to TTL conversion board used for STC microcontroller download DVD/ hard disk/router /GPS upgrade
9. For STC Download: SCM must be a minimum system and external crystal oscillator

5.5 Power Supply

The Dc power supply supplies 5v Power to the Micrcontroller and the other component to function accordingly.

**Circuit Diagram:-**



1. **Software System Design**:

**6.1 Algorithm :**

1. Include the required Library.
2. Define Controller Port Pin to LED Matrix Data Pin.
3. Define Led Matrix Size.
4. Define LED Matrix Parameter.
5. Initialize LED Matrix.
6. Initialize RTC.
7. Take the Input Data which is display on LED Matrix.
8. Convert this ASCII text into Matrix format and fill into buffer.
9. Scroll this buffer Data on LED Matrix Display.
10. Goto step 7

**6.2 Flowchart:**

Start

Convert this ASCII text into Matrix format and fill into buffer

Scroll this buffer Data on LED Matrix Display

Goto step 7

END

Include the required Library

Define Controller Port Pin to LED Matrix Data Pin

Define LED Matrix Size

Define LED Matrix Parameter

Initialize LED Matrix

Take the Input Data which is display on LED Matrix

Initialize RTC

1. **Result and Disscussion**

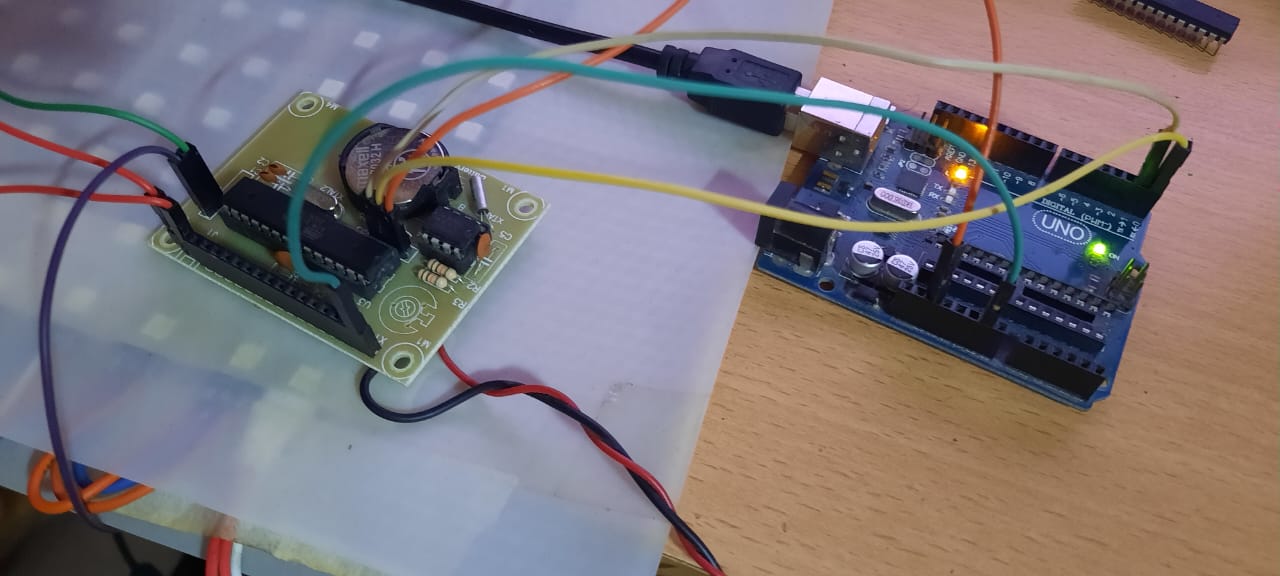


Fig7.1 PCB And Arduino for transfer program form system to the processor Atmega328 using serial pins of Arduino

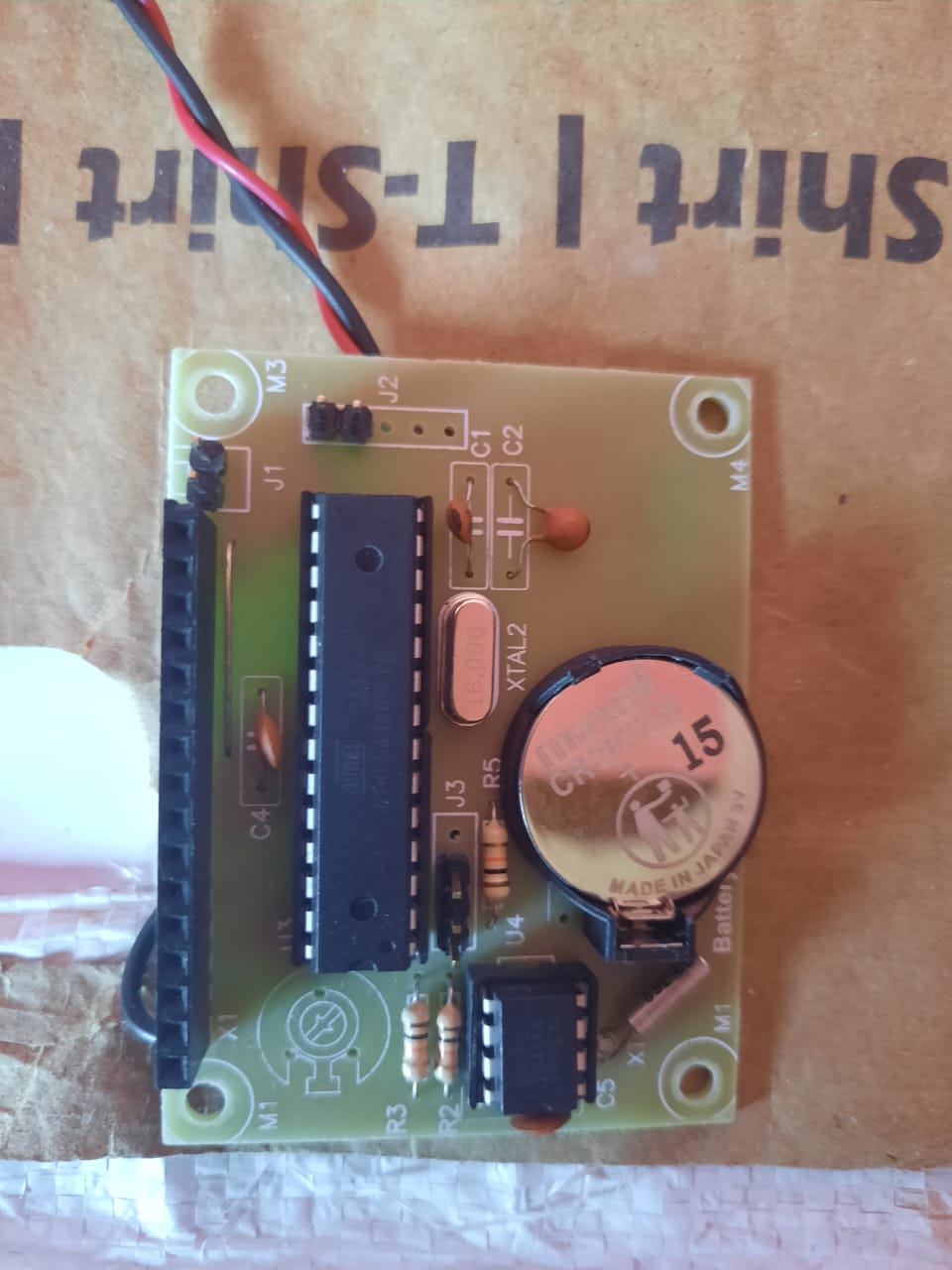


Fig 7.2 PCB Mounted

**Input Image:-**



Fig7.1 Panti Drawing

**Sorce Code:-**

*#include <Adafruit\_GFX.h>*

*#include <Adafruit\_NeoMatrix.h>*

*#include <Adafruit\_NeoPixel.h>*

*#define PIN 12*

*int delayValue=500;*

*int i,j,k,m,n,x;*

*//the Wemos WS2812B RGB shield has 1 LED connected to pin 2*

*Adafruit\_NeoPixel pixels = Adafruit\_NeoPixel(300, PIN, NEO\_GRB + NEO\_KHZ800);*

*const unsigned int bitmap24[]={*

*0xD1A4, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x001f1f, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x00, // 0x0010 1 -(00-14) pixels*

*0x00, 0x00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x001f1f, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x7F007F, 0x0000, // 0x0020 2 -(15-29) pixels*

*0x0000, 0x0000, 0x7F007F, 0x0000, 0x0000, 0x0000, 0x001f1f, 0x001f1f, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, // 0x0030 3 -(30-44) pixels*

*0x0000, 0x00, 0x0000, 0x0000, 0x0000, 0x0000, 0x001f1f, 0x001f1f, 0x000, 0x0000, 0x7F007F, 0x00, 0x0000, 0x00, 0x0000, // 0x0040 4 -(45-59) pixels*

*0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x001f1f, 0x001f1f, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, // 0x0050 5 -(60-74) pixels*

*0x0000, 0x0000, 0x0000, 0x0000, 0x00, 0x0000, 0x0000, 0x001f1f, 0x001f1f, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, // 0x0060 6 -(75-89) pixels*

*0x0000, 0x0000, 0x0000, 0x7F007F, 0x0000, 0x0000, 0x0000, 0x001f1f, 0x001f1f, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, // 0x0070 7 -(90-104) pixels0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x001f1f, 0x001f1f, 0x0000, 0x0000, 0x0000, 0xD1A4, 0x00, 0x0000, // 0x0080 8 -(105-119) pixels*

*0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x00, 0x001f1f, 0x001f1f, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, // 0x0090 9 -(120-134) pixels*

*0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x001f1f, 0x001f1f, 0x00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, // 0x00A0 10(135-159) pixels*

*0x0000, 0x0000, 0x0000, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0x0000, 0x0000, 0x0000, // 0x 00B0 11-(150-164) pixels*

*0x0000, 0x0000, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0x0000, 0x0000, // 0x00C0 12-(165-179) pixels*

*0x0000, 0xA902, 0xA902, 0xA902, 0xA902, 0x00007f, 0x00, 0x00007f, 0x00007f, 0x00007f, 0xA902, 0xA902, 0xA902, 0xA902, 0x0000, // 0x00D0 13-(180-194) pixels*

*0x0000, 0xA902, 0xA902, 0xA902, 0xA902, 0x00007f, 0x7F007F, 0x00007f, 0x7F007F, 0x00, 0xA902, 0xA902, 0xA902, 0xA902, 0x0000, // 0x00E0 14-(195-209) pixels*

*0x0000, 0xA902, 0xA902, 0xA902, 0xA902, 0x00007f, 0x00007f, 0x00007f, 0x00007f, 0x00007f, 0xA902, 0xA902, 0xA902, 0xA902, 0x0000, // 0x00F0 15-(210-224) pixels*

*0x0000, 0x0000, 0xA902, 0xA902, 0xA902, 0x00, 0x7F007F, 0x00007f, 0x7F007f, 0x00007f, 0xA902, 0xA902, 0xA902, 0x0000, 0x0000, // 0x0100 16-(225-239) pixels*

*0x0000, 0x0000, 0xA902, 0xA902, 0xA902, 0x00007f, 0x00007f, 0x00007f, 0x00, 0x00007f, 0xA902, 0xA902, 0xA902, 0x0000, 0x0000, // 0x0110 17-(240-254) pixels*

*0x0000, 0x0000, 0x0000, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0x0000, 0x0000, 0x0000, // 0x0120 18-(255-269) pixels*

*0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0xA902, 0x0000, 0x0000, 0x0000, 0x0000, // 0x0130 19-(270-284) pixels*

*0xD1A4, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x00 // 0x0140 20-(285-300) pixels*

*};*

*void setup()*

*{*

*Serial.begin(9600);*

*pixels.begin(); // This initializes the NeoPixel library.*

*}*

*void loop()*

*{*

*/\**

*for(m=0; m<300; m++) { pixels.setPixelColor(m,0x0); }*

*pixels.show();*

*//-----------------------------------------------------------------------------------------------------*

*// Image rolling from Top to the Center of the Matrix*

*//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*for(m=0; m<15; m++){ Serial.print(m); Serial.print(" "); }*

*Serial.println("\n");*

*delay(500);*

*//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*for(m=0; m<15; m++){ Serial.print(m); Serial.print(" "); }*

*for(m=0; m<15; m++){ Serial.print(29-m); Serial.print(" "); }*

*Serial.println("\n");*

*delay(500);*

*//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*for(m=0; m<15; m++){ Serial.print(m); Serial.print(" "); }*

*for(m=0; m<15; m++){ Serial.print(29-m); Serial.print(" "); }*

*for(m=0; m<15; m++){ Serial.print(30+m); Serial.print(" "); }*

*Serial.println("\n");*

*delay(500);*

*//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*for(m=0; m<15; m++){ Serial.print(m); Serial.print(" "); }*

*for(m=0; m<15; m++){ Serial.print(29-m); Serial.print(" "); }*

*for(m=0; m<15; m++){ Serial.print(30+m); Serial.print(" "); }*

*for(m=0; m<15; m++){ Serial.print(59-m); Serial.print(" "); }*

*Serial.println("\n");*

*delay(500);*

*delay(3000);*

*}*

*\*/*

**Output:-**

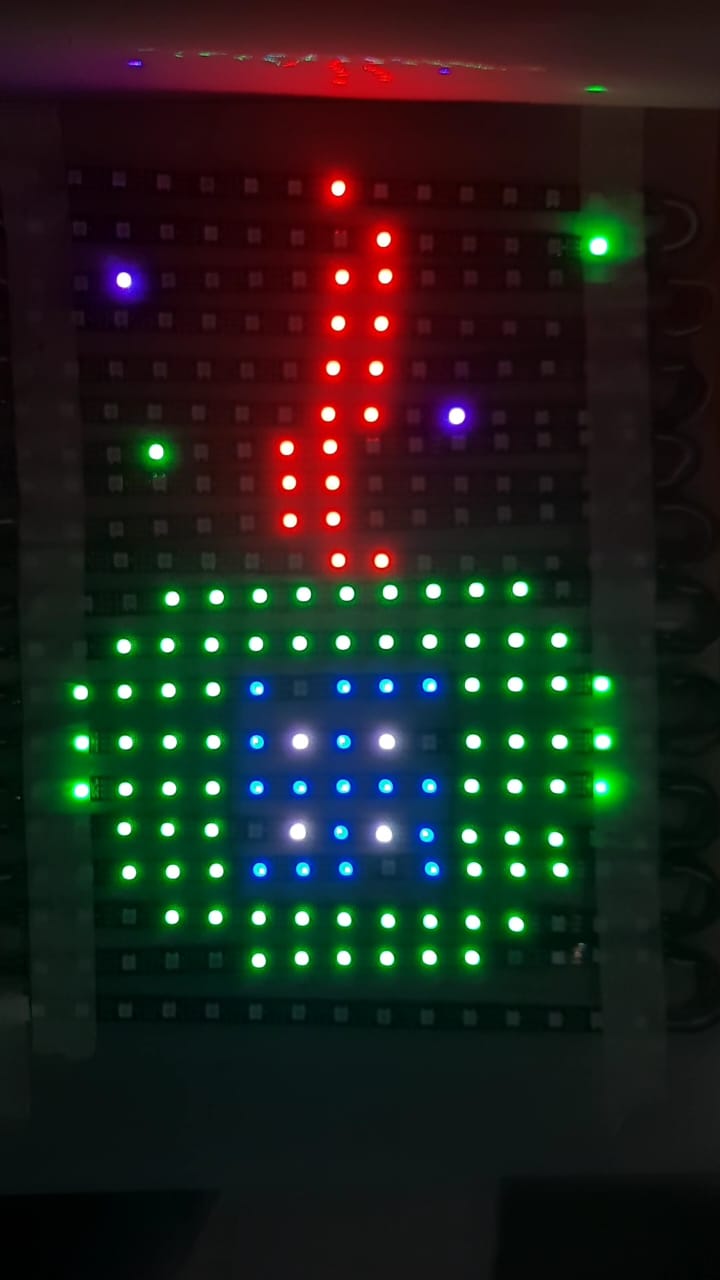


Fig.7.3 Scrolling Panti

**Source Code :**

*/ Project Name : Neo Pixel*

*#include <Adafruit\_GFX.h>*

*#include <Adafruit\_NeoMatrix.h>*

*#include <Adafruit\_NeoPixel.h>*

*#include <FastLED\_NeoMatrix.h>*

*#include <FastLED.h>*

*#include <Wire.h>*

*#include <DS1307.h>*

*DS1307 rtc;*

*/\**

*#ifndef PSTR*

*#define PSTR*

*#endif*

*\*/*

*#define PIN 12*

*#define mw 20*

*#define mh 15*

*// MATRIX DECLARATION:*

*// Parameter 1 = width of NeoPixel matrix*

*// Parameter 2 = height of matrix*

*// Parameter 3 = pin number (most are valid)*

*// Parameter 4 = matrix layout flags, add together as needed:*

*// NEO\_MATRIX\_TOP, NEO\_MATRIX\_BOTTOM, NEO\_MATRIX\_LEFT, NEO\_MATRIX\_RIGHT:*

*// Position of the FIRST LED in the matrix; pick two, e.g.*

*// NEO\_MATRIX\_TOP + NEO\_MATRIX\_LEFT for the top-left corner.*

*// NEO\_MATRIX\_ROWS, NEO\_MATRIX\_COLUMNS: LEDs are arranged in horizontal*

*// rows or in vertical columns, respectively; pick one or the other.*

*// NEO\_MATRIX\_PROGRESSIVE, NEO\_MATRIX\_ZIGZAG: all rows/columns proceed*

*// in the same order, or alternate lines reverse direction; pick one.*

*// See example below for these values in action.*

*// Parameter 5 = pixel type flags, add together as needed:*

*// NEO\_KHZ800 800 KHz bitstream (most NeoPixel products w/WS2812 LEDs)*

*// NEO\_KHZ400 400 KHz (classic 'v1' (not v2) FLORA pixels, WS2811 drivers)*

*// NEO\_GRB Pixels are wired for GRB bitstream (most NeoPixel products)*

*// NEO\_GRBW Pixels are wired for GRBW bitstream (RGB+W NeoPixel products)*

*// NEO\_RGB Pixels are wired for RGB bitstream (v1 FLORA pixels, not v2)*

*// Example for NeoPixel Shield. In this application we'd like to use it*

*// as a 5x8 tall matrix, with the USB port positioned at the top of the*

*// Arduino. When held that way, the first pixel is at the top right, and*

*// lines are arranged in columns, progressive order. The shield uses*

*// 800 KHz (v2) pixels that expect GRB color data.*

*Adafruit\_NeoMatrix matrix = Adafruit\_NeoMatrix(20, 15, PIN,*

*NEO\_MATRIX\_BOTTOM + NEO\_MATRIX\_LEFT +*

*NEO\_MATRIX\_COLUMNS + NEO\_MATRIX\_ZIGZAG,*

*NEO\_GRB + NEO\_KHZ800); // changes--> LEFT, ZIGZAG,ROWS*

*const uint16\_t colors[] = {*

*matrix.Color(255, 0, 0), matrix.Color(0, 255, 0), matrix.Color(0, 0, 255) };*

*void setup()*

*{*

*//only set the date+time one time*

*// rtc.set(0, 48, 4, 2, 4, 2022); //08:00:00 24.12.2014 //sec, min, hour, day, month, year*

*// Serial.begin(9600);*

*matrix.begin();*

*matrix.setTextWrap(false);*

*matrix.setBrightness(100);*

*matrix.setTextColor(colors[0]);*

*rtc.start();*

*}*

*int x = matrix.width();*

*int pass=1;*

*void loop()*

*{*

*// rolling\_message();*

*// read\_time();*

*test\_matrix();*

*}*

*//======================== Rolling Message Routine ===================================*

*void rolling\_message()*

*{*

*matrix.fillScreen(0);*

*matrix.setCursor(x, 3);*

*// matrix.print(F("OMKAR Kashid"));*

*matrix.print(F("WELCOME TO MODERN COLLEGE OF ENGINEERING, E&TC DEPARTMENT"));*

*if(--x < -356)*

*{*

*x = matrix.width();*

*if(++pass >= 3) pass = 0;*

*matrix.setTextColor(colors[pass]);*

*}*

*matrix.show();*

*delay(150);*

**OUTPUT:-**

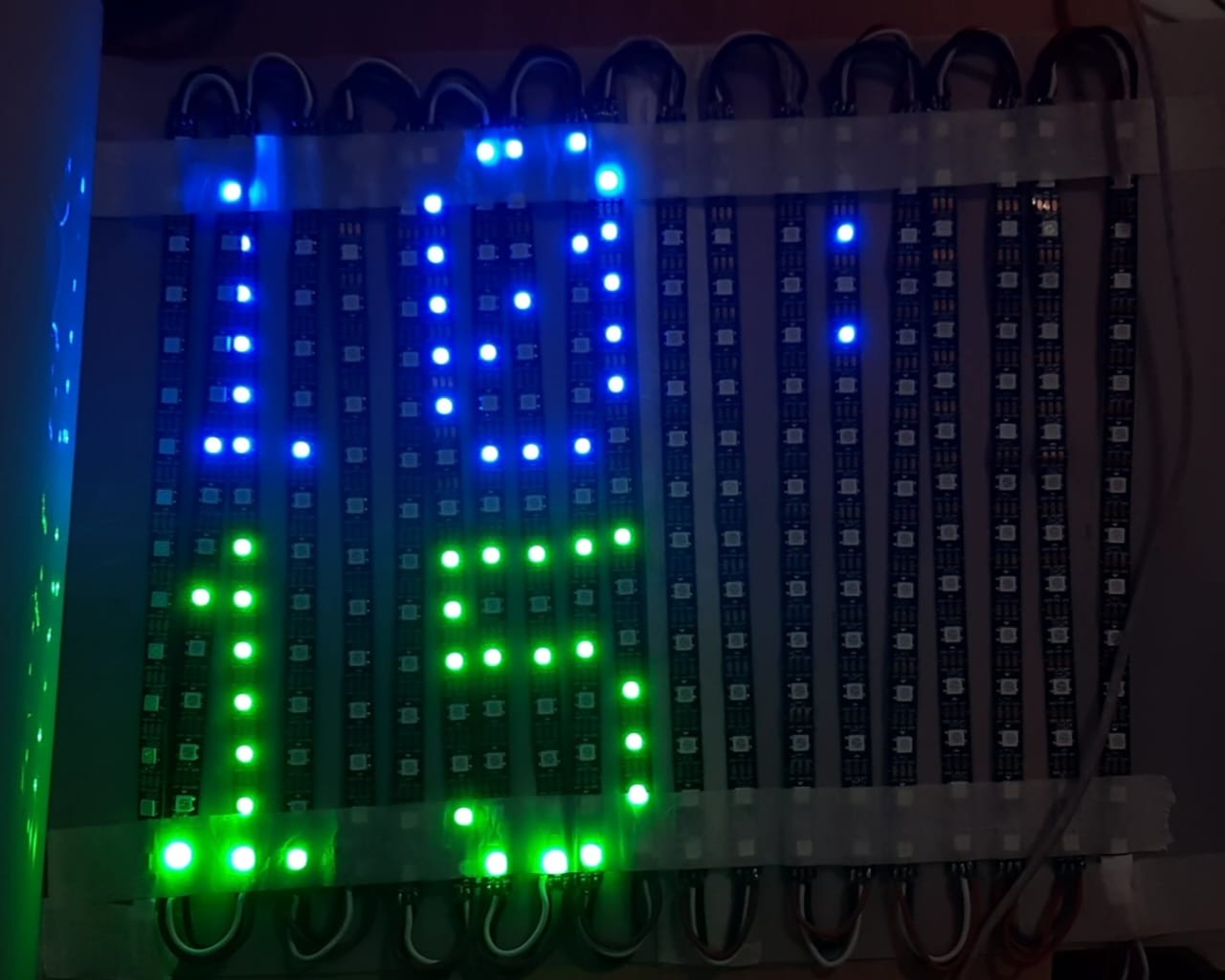


Fig 7.5 RTC real time clock

**Source Code:-**

*void rolling\_message()*

*{*

*matrix.fillScreen(0);*

*matrix.setCursor(x, 3);*

*// matrix.print(F("OMKAR Kashid"));*

*matrix.print(F("WELCOME TO MODERN COLLEGE OF ENGINEERING, E&TC DEPARTMENT"));*

*if(--x < -356)*

*{*

*x = matrix.width();*

*if(++pass >= 3) pass = 0;*

*matrix.setTextColor(colors[pass]);*

*}*

*matrix.show();*

*delay(150);*

*}*

*//========================= Clock display Routine ==================================*

*void read\_time()*

*{*

*uint8\_t sec, min, hour, day, month,h;*

*uint16\_t year;*

*rtc.get(&sec, &min, &hour, &day, &month, &year); //get time from RTC*

*h=hour % 12 ;*

*if(h == 0) { h=12;}*

*matrix.fillScreen(0);*

*matrix.setCursor(0, 0);*

*if(h<9) { matrix.print("0"); }*

*matrix.print((h));*

*matrix.print(":");*

*matrix.setTextColor(colors[1]);*

*matrix.show();*

*matrix.setCursor(0, 8);*

*matrix.print((min));*

*matrix.setTextColor(colors[2]);*

*matrix.show();*

*delay(5000);*

*}*

**Output:-**

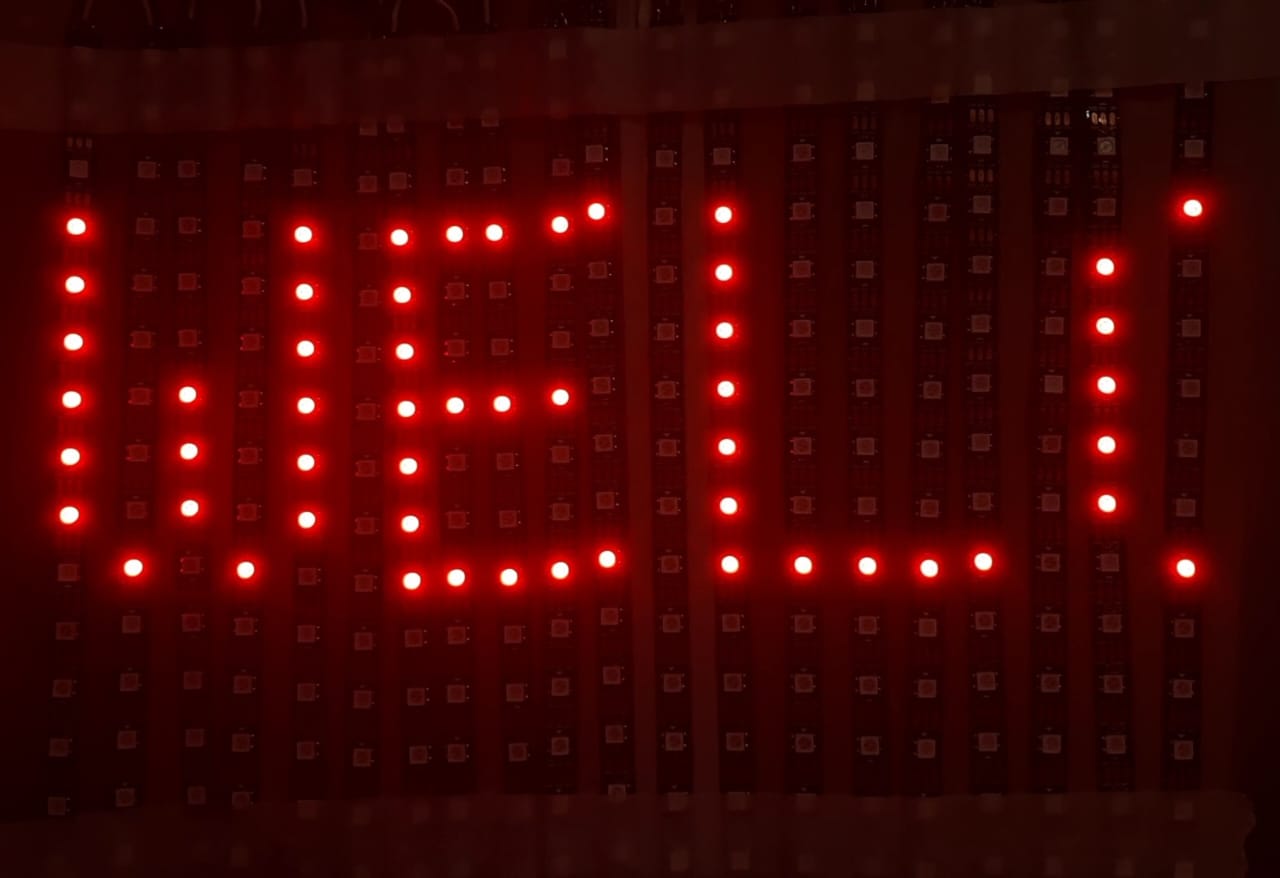


Fig 7.4 Scrolling Message

1. **Bill of Material :**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr.No** | **Component name , Model Name and Specification** | **Quantity** | **Rate/ Item (Rs.)** | **Total Cost(Rs.)** |
| 1 | Neo WS28128B LED Matrix | 1 | 3500/- | 3500/- |
| 2 | PCB LAYOUT | 1 | **500/-** | 3500/- |
| 3 | Film and PCB | 1 | **1200/-** | 1200/- |
| 4 | Atmega328 | 1 | **2500/-** | 2500/- |
| 5 | USB To TTL Convrter | 1 | **700/-** | 700/- |
| 6 | USB Type D cable | 1 | **700/-** | 700/- |
| 7 | DS1307 | 1 | **90/-** | 90/- |
| 8 | Battery | 1 | **25/-** | 25/- |
| 9 | Crystal | 2 | **12/-** | 24/- |
|  |  |  | **Total Amount (Rs.)** | 6736/- |

1. **Applications& Future modifications:**

**9.1 Applications :**

A. Educational Institution and Organization: Currently we rely on putting up papers on notice boards to inform people of events. This method can be discarded by using GSM based LED display to display information in real time.

E.g. Placement news, cultural activities news, etc.

B. Advertisement: In shopping malls we get to hear the offers on various products from time to time. Instead, we continuously display the information regarding the products and related offers on electronic display boards.

C. Railway Station: Instead of announcing the delay in arrival of trains we can display the information.

D. Hotels: To display the availability of the rooms and the room rents the type of rooms.

E. Nursing homes: To display the staff attendance, the availability of the doctors, the list of the specialized doctors, no of in patients etc.

**9.2 Future modifications:**

1. In future we can add a speaker which will tell us what is showing on LED matrix Display for e.g:- If time is showing on display then it will tell through speaker.
2. In future we can make it wireless using NodeMCU or Wifi , Blutooth for sending code through any system to Product.
3. Also we can make rolling display or folding display which will be easy to transport and take less space or efforts.
4. **Conclusion:**

Scrolling LED display is yet another simple device which can be used to display information or notice for various applications. Microcontroller with niopixel LED matrix can perform various scrolling pattern which are attractive as well as informative. Thus, this project is very suitable for small scale as well as medium scale information display. This model can be used very efficiently in establishments like chain restaurants wherein the order and special discounts can be displayed at all branches simultaneously, in colleges wherein students and staffs can be informed simultaneously in no time. It can be set up at public transport places like railways, bus station, and airport and also at roadside for traffic control and in emergency situations, it is cost efficient system and very easy to handle. Latency involved in using of papers in displaying of notices is avoided and the information can be updated by the authorized persons.

**11.References:**

[1] Jeena Joy, Athira Poovathody, Athul R S, Athul Subran, Basil Paul, Implementation of Digital Dice Game‖, Professor, Dept. of EEE, Mar Athanasius College of Engineering, Kothamangalam1 UG Student, Dept. Of EEE, Mar Athanasius College of Engineering, Kothamangalam, India2, 3, 4, 5, Vol. 3, Issue 2, February 2014.

[2] Harold Thimbleby FIT Lab Interaction Laboratory, Swansea University, Don’t use seven segment displays‖ Swansea, Wales. p- 1-6

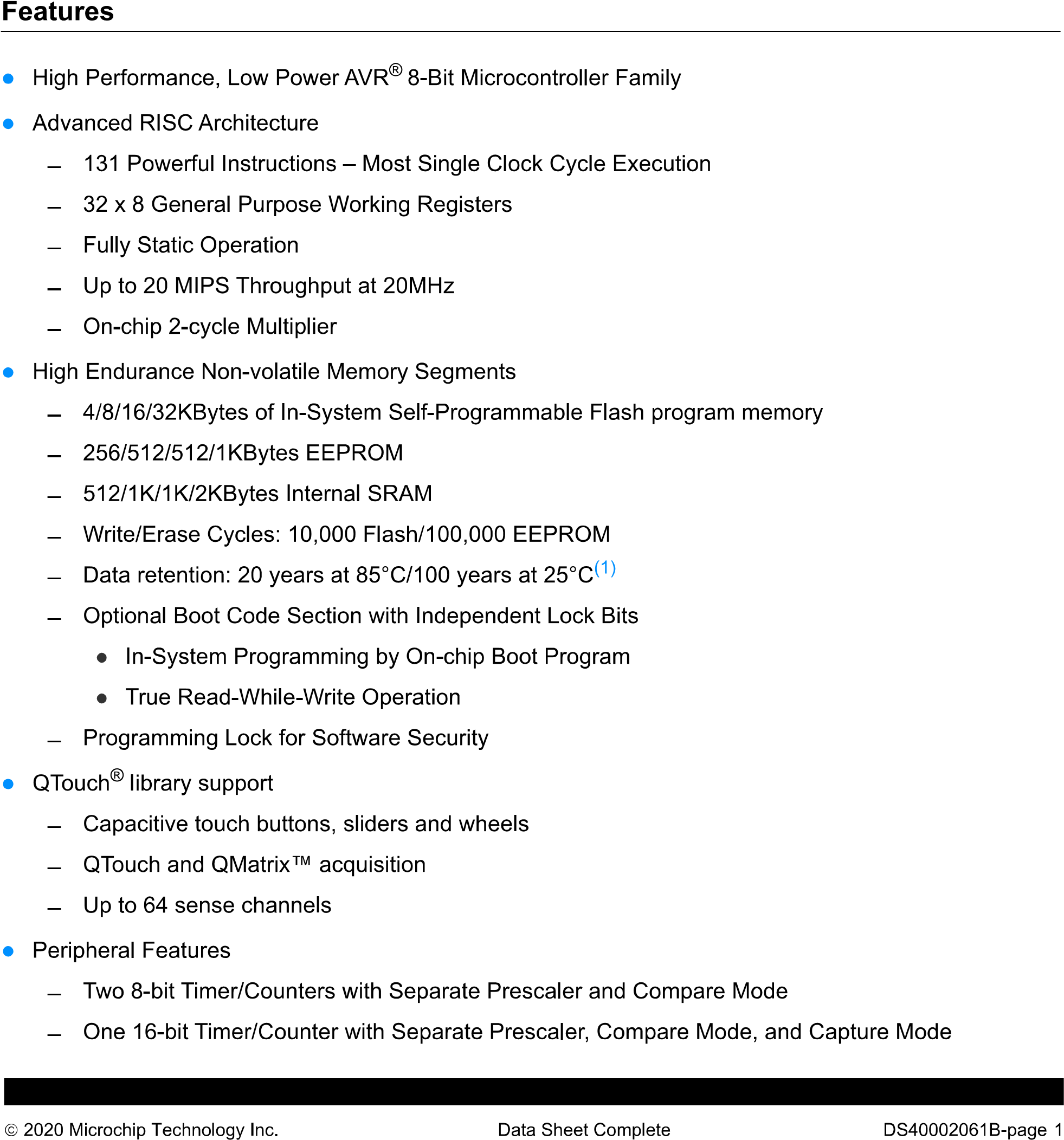
[3] Pang, GKH; Chan, CH; Kwan, TTO, Tricolor lightemitting diode dot matrix display system with audio output‖, IEEE TRANSACTIONS ON INDUSTRY APPLICATIONS, VOL. 37, NO. 2, MARCH/APRIL 2001 , p.534-540.

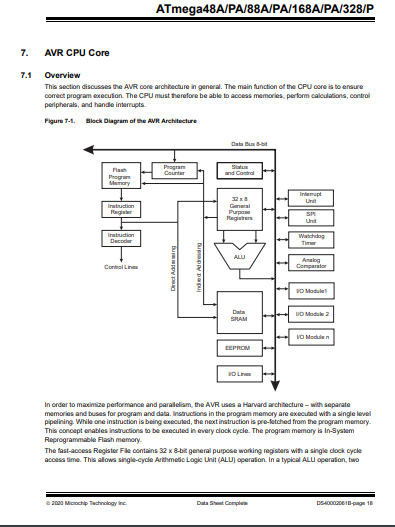
[4] Ervin John U. Benigra, Bryan Leonard D. Montaño and Engr. Maridee B. Adiong,‖ RUNNING MESSAGE BOARD USING DOT-MATRIX DISPLAY‖ Capitol University ,College of Engineering, Cagayan de Oro City.

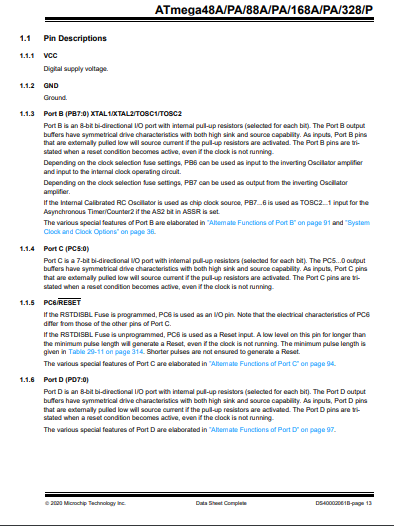
[5] Wojciech Kunikowski, Ernest Czerwiński, Paweł Olejnik, Jan Awrejcewicz, An Overview of ATmega AVR Microcontrollers used in Scientific Research and Industrial Applications, Department of Automation, Biomechanics and Mechatronics, Lodz University of Technology, 90–924 Łódź, 1/15 Stefanowski str. R. 19, Nr 1/2015, 15–20, DOI: 10.14313/PAR\_215/15

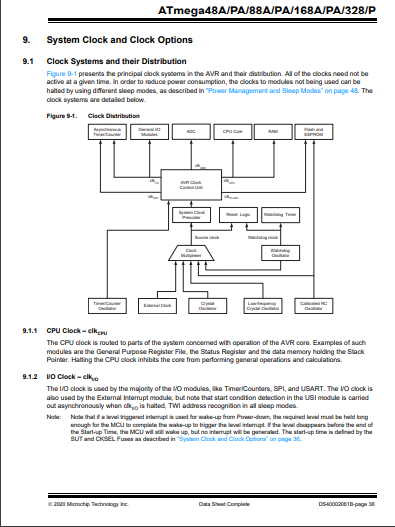
1. **Data Sheets :**

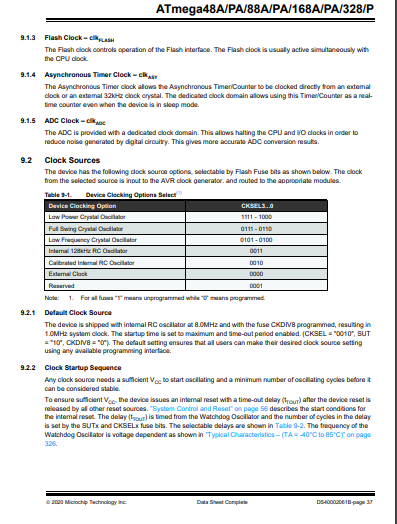
10.1 Atmega328

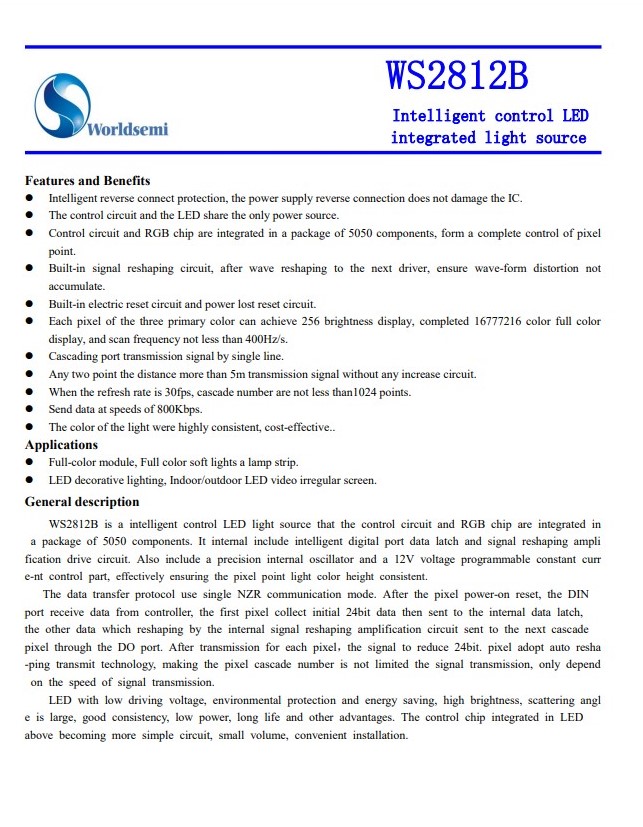


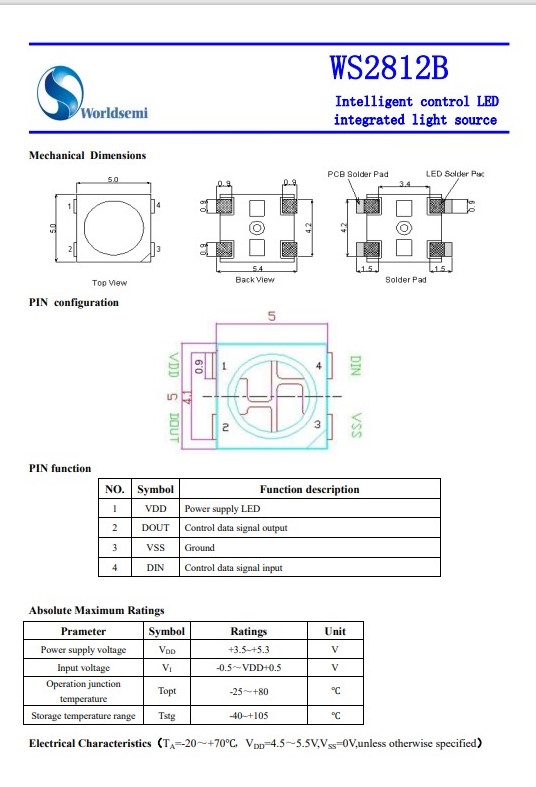


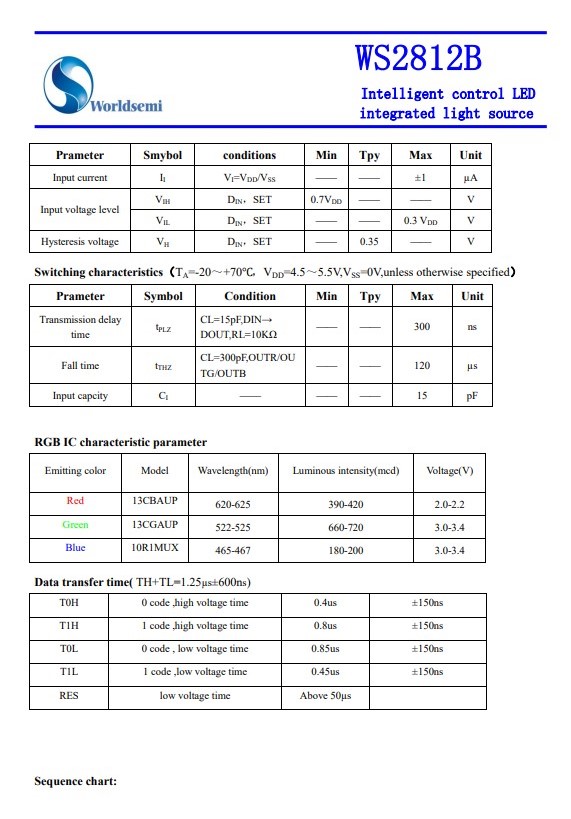


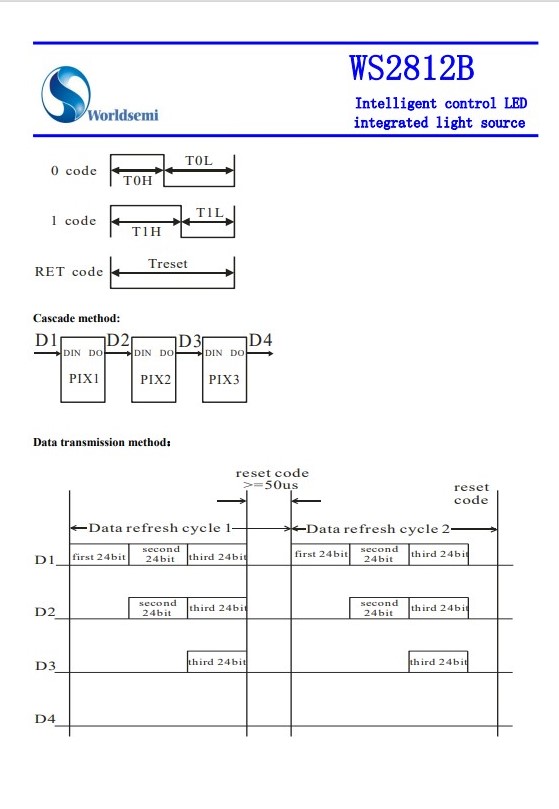


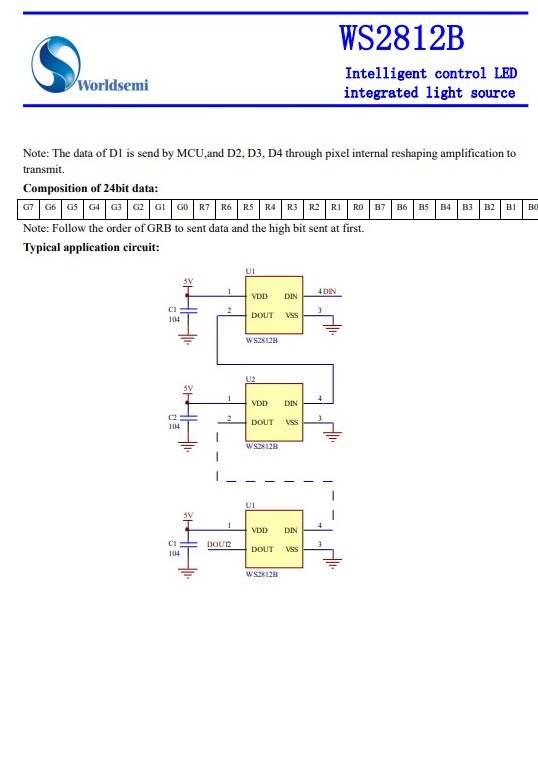


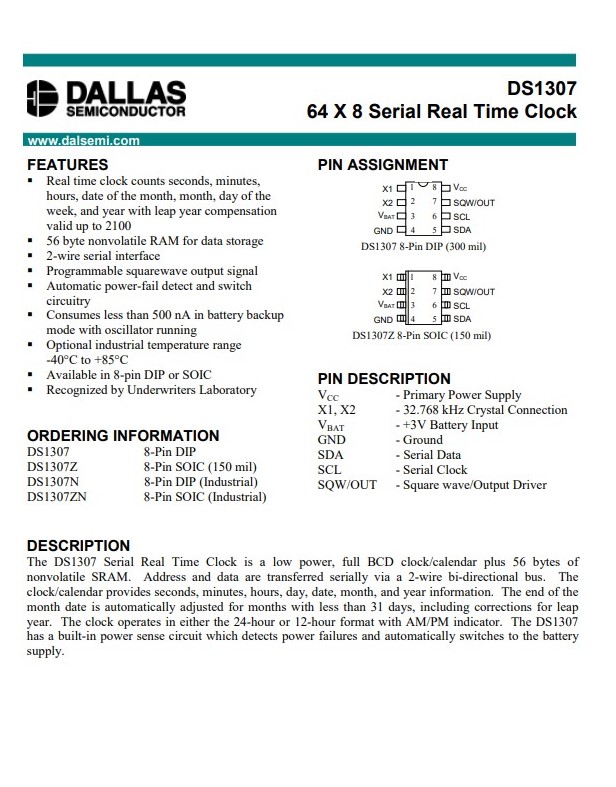


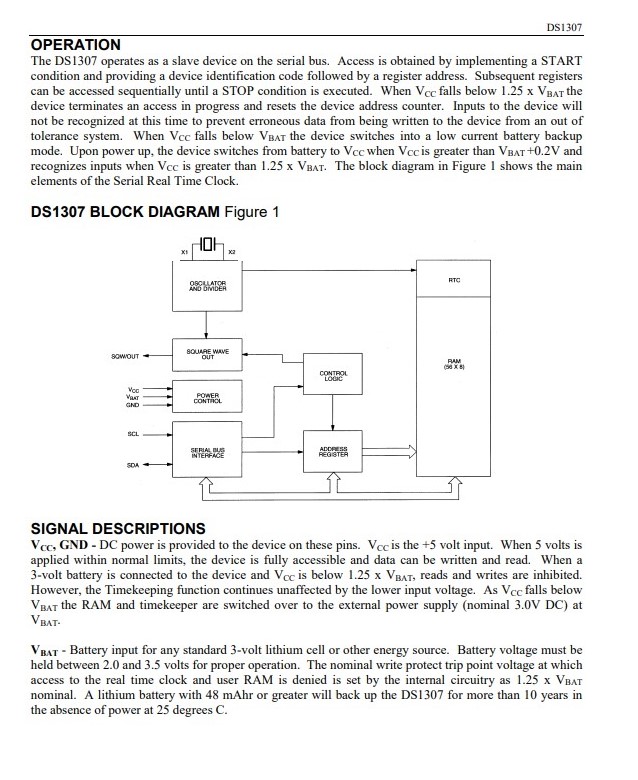


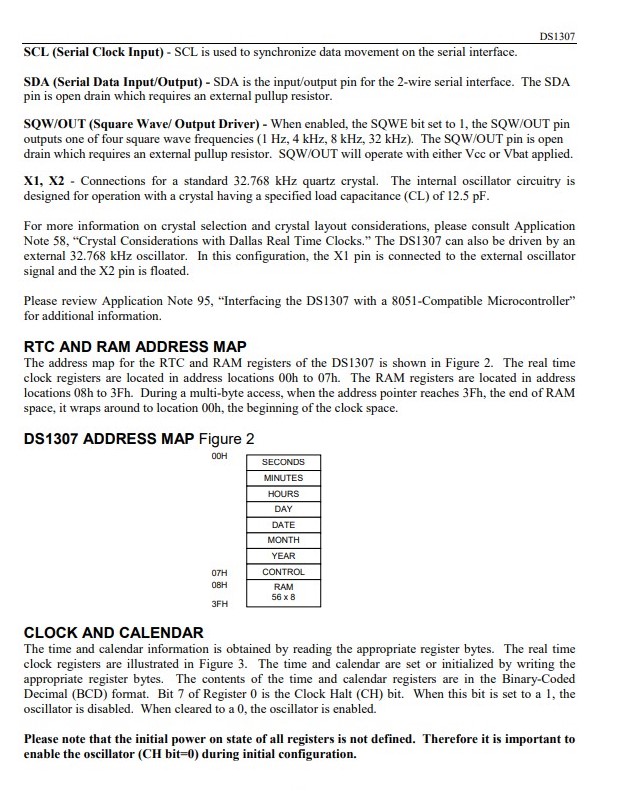


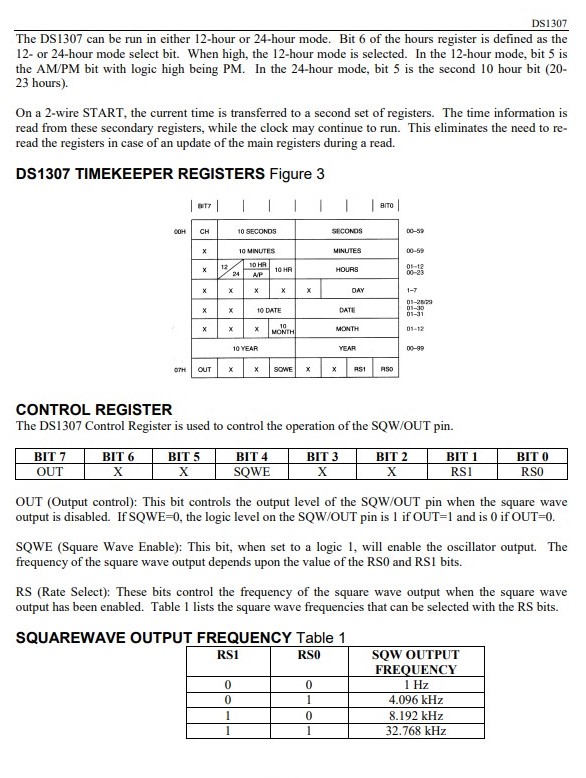


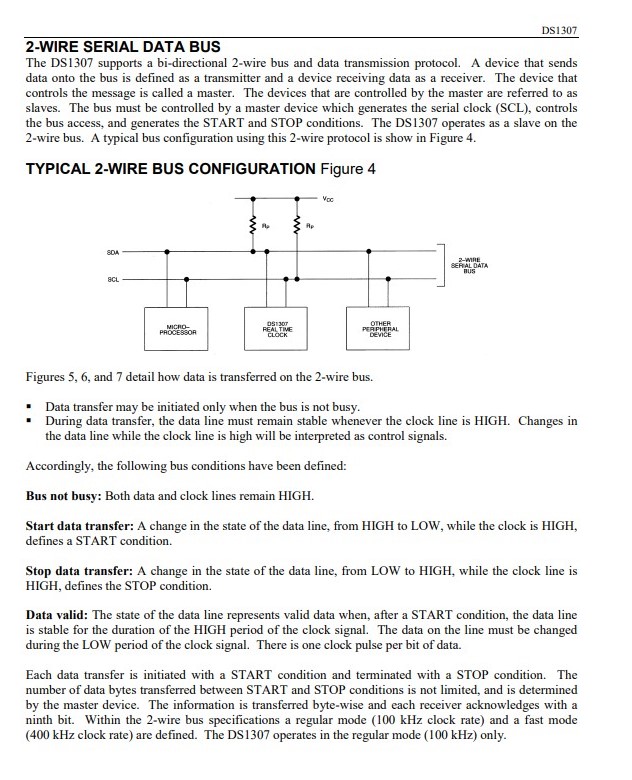












Dr. Mrs. K.A.Adoni

Mr.Ramgopal Sahu

Mrs. Seema Bhalgaonkar