REMOTE DISPLAY

-LED Matrix-

By : Ashwini Kumar Gupta
B. Engg Electronics & Telecommunication
August 10, 2019

Contents

List of Figures

List of Tables

1 Project Description

1.1 Introduction

The Remote Display utilises a simple static scrolling Light Emitting Diode (LED) display and makes it accessible from a nearby location using the Bluetooth Low Energy (BLE) feature of an Android phone. Using such a feature the user can modify the display at will and with ease, this reduces the time to manually reprogram the controlling unit and does not require any computer. Having remote access makes the display very handy for a users who keeps the display at a non accessible location.

1.2 Feature and Functionality

The Remote Display has following features and functionality.

Table 1: Feature and Functionality

Feature	Functionality
BLE	Provide remote Access
USB	Communicate with computer
BLE LED	BLE status indicator
Speed Controller	Control scrolling speed
Parameter Mode	To edit control parameters

1.3 Hardware

1.3.1 MCU

The remote display is built around a ATmega328P Alf and Vegard's RISC (AVR) microcontroller. This MCU is based on advanced Reduced Instruction Set Computer (RISC) [?] architecture, 8 bit MCU and 23 programmable Input and Output Line (I/O) lines. For controlling the REMOTE DISPLAY shift registers

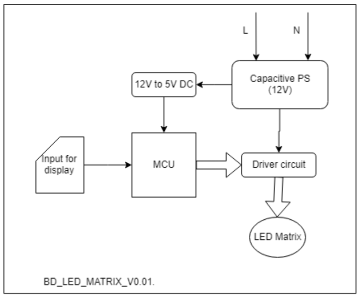


Figure 1: Block Diagram V1.0

are used, 2 for Column and 1 for row, due to shift registers few MCU I/O lines are used.

1.3.2 LED Matrix

Single colour red 5mm LED used to build the matrix, total of 128 LED required for 16x8 matrix. An array of transistors configured as switch to provide required current for each row of LED matrix.

1.3.3 Power supply

The project utilises a transformerless capacitive power supply design. Such a design is helpful in reducing the overall cost of project and also utilises fewer components thus saving space and cost.

1.3.4 Input

The project is aimed to dynamically modify the display commands through an input source like computer or BLE, through Universal Synchronous Asynchronous Receiver Transmitter (USART). Such a feature helps in modifying the display at will rather than modifying the source code.

2 List Of Tools

2.1 Introduction

Enter some text here

3 Hardware

3.1 Introduction

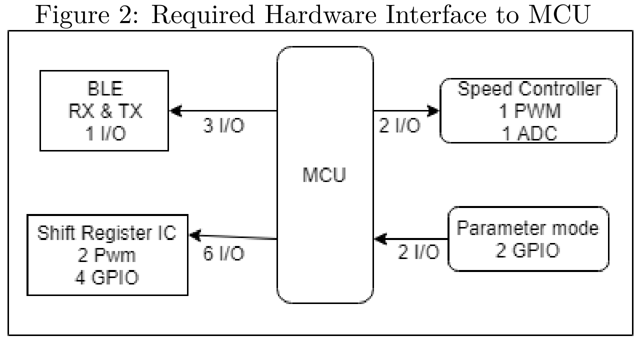
Enter some text here

3.2 Control Unit

3.2.1 MCU

Selecting the MCU

Figure 2: Required Hardware Interface to MCU



Step 1: *Make a list of required hardware interfaces*
To select a particular MCU, the required hardware interfacing is mandatory because hardware interface defines the required peripherals and General Purpose Input and Output (GPIO). The list of hardware interface in this project is provided in Table 2(also refer 2).

Step 2: *Examine the architecture*
The project requires fast execution of instruction and repetitive access to memory due to use of communication protocols and USART peripheral. The **Harvard architecture** suites this application more than the Von-Neuman architecture, since Harvard architecture has dedicated buses for different sections of MCU

and has pipeline instruction set feature, thus executing more instructions than Von-Neuman architecture in a given period of time. The MCU should have be either 8 bit or 16 bit, since the application does not require high and complex processing. A 8 bit MCU supporting few Mhz of frequency will also suit this application.

Step 3: *Identify Memory Needs*
This project would require the following minimum memory(as per rough calculations).

EEPROM : 1 KB

Flash : 16 KB

SRAM : 512 Bytes

Step 4: *Searching for MCU*
Selecting MCU as per above parameters, AVR series from Atmel crop and MSP40 series of Texas Instruments are good options.considering future enhancement, cost, development kits, support for MCU and development platform (like IDE and drivers) the AVR series ATmega328P is selected.

ATmega328P

The MCU is the central processing unit of the system. For this application/project the ATmega328P, by Atmel Corporation is selcted. Following is the list of features [**?**].

- Advanced RISC architecture
- 32K bytes of in-system self-programmable flash program memory.
- 1Kbytes Electrically Erasable Programmable Read-Only Memory (EEPROM).
- 2Kbytes Static Random Access Memory (SRAM).
- Two 8-bit Timer/Counters
- One 16-bit Timer/Counter
- Six Pulse Width Modulation (PWM) channels
- 8-channel 10-bit Analog to Digital Converter (ADC)
- USART
- Master/slave Serial Peripheral Interface (SPI)
- Inter-Integrated Circuit (I2C)
- watchdog timer
- On-chip analog comparator
- Six sleep modes

3.2.2 Oscillator Circuit

The oscillator circuit is required for generating clock for MCU.Accuracy of timing application is dependent on the accuracy of the oscillator provided in the MCU. The oscillators supported by ATmega328P are listed below.

- Low power crystal oscillator
- **Full swing crystal oscillator**
- Low frequency crystal oscillator
- Internal 128kHz RC oscillator
- Calibrated internal RC oscillator
- External clock

For this application the MCU will be in Full swing crystal oscillator mode.

Table 2: Required Hardware Interface to MCU

Sr. no.	**Interface**	**GPIO**	**PWM**	**ADC**	**USART**	**Remark**
1	**BLE**	0	0	0	1	RX and TX
2	**BLE LED**	1	0	0	0	Show BLE status
3	**Speed Control**	0	1	1	0	Control scrolling speed
4	**Parameter mode**	2	0	0	0	Edit parameters
5	**Serial Interface**	0	0	0	1	RX and TX Multiplexed with BLE
6	**IC 74HC595**	4	2	0	0	1 Row 2 Col
	Total	**7**	**3**	**1**	**1**	**12**

Advantages:

- 16 MHz.
- rail to rail swing.
- Drive other clock sources
- Less effect of noisy environment

Disadvantages

- Higher current consumption than Low power crystal oscillator.
- Operating Voltage of MCU becomes 2.7 V to 5.00 V.

Overcoming the disadvantage : Using 230V AC source, so power consumption wont be an issue, MCU will be working on 5 V DC which is within the operating voltage defined.

Selecting crystal requires following considerations.

1. Through-Hole Technology (THT) or Surface Mount Technology (SMT).
2. Load capacitance.
3. Frequency of operation
4. Q - factor
5. Equivalent Series Resistance (ESR)
6. Frequency Pulling
7. Drive level
8. Minimum negative Resistance
9. Frequency stability
10. Frequency Tolerance

Printed Circuit Board (PCB) design. [?]

3.2.3 Reset

3.2.4 Port Assignment

3.2.5 Voltage Level Indicator

3.2.6 Display Intensity controller

3.3 Power Supply

3.3.1 Power Switch

3.3.2 Capacitive Power Supply

3.3.3 Filter

3.3.4 voltage Regulation

3.3.5 Current Consumption

3.4 LED Matrix

3.4.1 LED

3.4.2 Transistors

3.4.3 Shift Registers

3.5 Input

4 Software

4.1 Introduction

Enter some text here

5 PCB Design

5.1 Introduction

Enter some text here

6 Mechanical CAD

6.1 Introduction

Enter some text here

A Crystal

some text here

B Title of app 22

some thext theri

Acronyms

ADC Analog to Digital Converter. 5, 6

AVR Alf and Vegard's RISC. 1, 5

BLE Bluetooth Low Energy. 1, 2, 6

EEPROM Electrically Erasable Programmable Read-Only Memory. 5

ESR Equivalent Series Resistance. 6

GPIO General Purpose Input and Output. 4, 6

I/O Input and Output Line. 1, 2

I2C Inter-Integrated Circuit. 5

IC Integrated Circuits. 6

KB Kilo Bytes. 5

LED Light Emitting Diode. 1, 2, 6

MCU Micro-controller Unit. iii, iv, 1, 2, 4, 5, 6

PCB Printed Circuit Board. 6

PWM Pulse Width Modulation. 5, 6

RISC Reduced Instruction Set Computer. 1, 5

RX Receive Pin. 6

SMT Surface Mount Technology. 6

SPI Serial Peripheral Interface. 5

SRAM Static Random Access Memory. 5

THT Through-Hole Technology. 6

TX Transmit Pin. 6

USART Universal Synchronous Asynchronous Receiver Transmitter. 2, 4, 5, 6

USB Universal Serial Bus. 1