# Design of Optical Mark Recognition (OMR) Scanner using ATMega128 Controller

Discipline: Electronics

## Declaration of Sole Authorship

I, [redacted], confirm that this work submitted for assessment is my own and is expressed in my own words. Any uses made within it of the works of any other author, in any form (ideas, equations, figures, texts, tables, programs), are properly acknowledged at the point of use. A list of the references used is included.

Signed: [redacted]

[redacted]

# Approved TR Proposal

## TR Report Results

OACETT ×

[redacted]@oacett.org>
to me

[redacted]

This is a follow up to your Technology Proposal Design of Optical Mark Recognition (OMR) Scanner using ATMega128 Controller that was accepted.

Your Technology Report is due by [redacted]

Please follow the Technology Report Guidelines.

Please send your TR Report electronically as a PDF attachment.

Thank you,

OACETT | [redacted]

# Table of Contents

# List of Figures

# ABSTRACT

Optical Mark Recognition (OMR) is the process of capturing human-marked data from document forms such as surveys and tests. OMR Sheet Correcting machine consists of an electronic device, which upon interfaced with a computer scanner, can provide the score or various other details entered in the OMR sheet. The scanner scans the OMR sheet with optocoupler pair for each answer choice, 4 in this case. The output of optocoupler is then fed to the ADC input of microcontroller. The microcontroller compares the answer detected with correct answer and continues to do so for specified number of times. The external hardware consists of mainly optical sensors and its circuitry. To make the device highly automatic, we have used the paper - rolling mechanism of a ready-made scanner.

# CHAPTER 1

## Introduction

Optical Mark Recognition (OMR), also called mark sensing, is a technique to sense the presence or absence of marks by recognizing their depth (darkness) on sheet. Optical Mark Recognition (OMR) is the process of capturing human-marked data from document forms such as surveys and tests. The OMR scanning machine consists of an electronic device, which when interfaced with a computer scanner, can provide the score or various other details entered in the OMR sheet. OMR software interprets the output from the scanner, and translates it into the desired output.

## Problem Statement

Many traditional OMR devices work with a dedicated scanning mechanism in which a beam of light is projected onto the paper. The contrasting reflectivity at predetermined positions on a page is then utilized to detect the marked areas because they reflect less light than the blank areas of the paper. Due to these characteristics, using dedicated hardware components such as sensors, light source makes design complex and more expensive. Some OMR devices use forms which are preprinted onto 'trans-optic' paper. The process involving manufacturing of such a high-quality paper could significantly increases the operational cost of using OMR Scanner. Furthermore, OMR scanner requires software to perform the scanning operation and dedicated programming could lead to the higher functional cost. Finally, most of the software available in market is designed to perform only specific scanning task and there is a limited scope to modify software in case user's requirement changes.

## Intent

Intent of this proposal is to design an OMR Scanner in order to evaluate the OMR sheets whose performance is not affected by the quality of the paper used for OMR forms/survey sheets. Therefore, the focus of this proposed system is to design a budget friendly OMR scanner which would be used to assess the OMR answer sheets for various competitive exams in school and universities even using low grade paper. Another aspect of this proposed design is to develop a cost-effective scanner integrated with simple user-friendly embedded C software code which could be modified for any further development in order to make this design better in the view of efficiency, cost and speed.

## Proposed Design

- The proposed design detects the marked response to a particular question and then compares it with the correct response. In this way it generates results of the entire OMR sheet. The design is based on Atmel's Atmega128 microcontroller development board.
- The scanner scans the OMR sheet under a strip of IR sensors for each answer choice.
- The output of sensors is then fed to ADC inputs of the microcontroller.
- The microcontroller compares the answer detected with correct answer and continues to do so for specified number of times.
- The external hardware consists of mainly optical sensors and its circuitry. The rolling of the OMR sheet under the series of reflective optical sensors can be automated with the use of a motor. A motor can be used to roll the paper in fixed length, which is the gap between two answer rows. In this way, the scanner will scan one row of answer and then the motor will roll the sheet to bring next answer row under the scanner. Adding this feature will make the device highly automatic.
- The proposed scanner will be designed to maximize the speed of the scanning mechanism and to perform its operations with an improved efficiency.
- The current scanners available in the market need named OMR Software to make scanner operational. The proposed design will be able to scan and correct the OMR sheets even without any specific name software and scanning can be achieved by implementing a user friendly basic embedded C programming which could be modified as per user's requirements. The new design will also be flexible enough to handle OMR Sheets of different dimensions like the one offered by most of the scanners available currently in the market.

### Features of the Proposed System

- Scanning of any number of rows
- LCD display for user interface
- Displays number of correct and wrong answers

# CHAPTER 2

## LITERATURE SURVEY

## 2.1 Research Paper: A Low-Cost OMR Solution for Educational Applications

Author: Hui Deng, Feng Wang, Bo Liang Computer Technology Appliance Key Lab of Yunnan Province, Kunming University of Science and Technology, China, 65005

### 2.1.1 Summary

Mark Recognition (OMR) is a traditional data input technique and an important human computer interaction technique which is widely used in education testing. Aimed at the drawbacks of Optical current OMR technique, a new image-based low-cost OMR technique is presented in the paper. The new technique is capable of processing thin papers and low-printing precision answer sheets. The system key techniques and relevant implementations, which include the images can, tilt correction, scanning error correction, regional deformation correction and mark recognition, are presented. This new technique is proved robust and effective by the processing results of large number of questionnaires.

### 2.1.2 Advantages

LCOMR has following advantages: -

1. A Microsoft Word macro-based sheet design technique to simplify the design of questionnaire.

2. Low-cost image-based OMR technique and the images can be obtained from any kinds of scanner.

3. Global and regional area image deformation corrections to improve the recognition precision.

As a complex procedure, LCOMR system consists of following main parts: -

• OMR sheet design

• Mark recognition

### 2.1.3 Disadvantages

1. The distortion of the thin paper is critical to affect the final recognition precision.

2. Ink infiltration is the main reason of recognition error. The plain paper (70 gsm or less) is easy to be infiltrated, so the writing on the rear side will infiltrate to the front side. This will cause the error of OMR recognition.

## 2.2 Sensors Analysis & Selection

### 2.2.1 Reflective Object Sensor- OPB733TR

The **OPB733TR** consists of an 890nm, Infrared Light Emitting Diode (LED) and an NPN silicon Phototransistor, which are mounted “side-by-side” on parallel axes in a miniature surface mount black plastic housing. The Phototransistor is molded in a dark epoxy package, which minimizes visible ambient light sensitivity. The phototransistor responds to radiation from the LED when a reflective object passes within its field of view. This unfocused reflective object sensor is ideal for non-contact detection of materials such as paper, labels, white plastic and many other reflective materials.

Features:

- Unfocused for sensing diffuse surfaces
- Uses lensed devices for collimation of light beam
- Low-cost plastic housing
- Compact surface mount package 0.300” x 0.160” x 0.114”
- Typical peak emission wavelength 890nm
- Reduced visible ambient light sensitivity
- Optimal operating distance range 0.4”.

### 2.2.2 Reflective Optical Sensor—CNY 70

The CNY70 has a compact construction where the emitting light source and the detector are arranged in the same direction to sense the presence of an object by using the reflective IR beam from the object. The operating wavelength is 940 nm. The detector consists of a phototransistor.

Features:

- Compact construction in center-to-center spacing of 0.1’
- No setting required
- High signal output
- Low temperature coefficient
- Detector provided with optical filter
- Current Transfer Ratio (CTR) of typical 5%

### 2.2.3 Reflective Optical Sensor-TCRT5000


The TCRT5000 have a compact construction where the emitting-light source and the detector are arranged in the same direction to sense the presence of an object by using the reflective IR-beam from the object. The detector consists of a phototransistor. There are 5 sensors connected in an array to form a TCRT 5000 Sensor Module which can be used effectively in different robotic applications such as, Line following robot, Shaft encoder, Obstacle avoidance, Detection of reflective materials such as paper, IBM cards, magnetic tapes etc.

Features:

- Detector type: phototransistor
- Dimensions (L x W x H in mm): 10.2 x 5.8 x 7
- Peak operating distance: 2.5 mm
- Operating range within > 20 % relative collector current: 0.2 mm to 15 mm
- Typical output current under test: Ic = 1 mA
- Daylight blocking filter
- Emitter wavelength: 950 nm

## 2.3 Comparison of the electrical characteristics of all 3 sensors:

| Parameter | OPB733TR | TCRT5000 | CNY70 |
|---|---|---|---|
| **Sensing Distance:** | 10.2 mm to 25.4 mm | 0.2 mm to 15 mm | 0 mm to 5 mm |
| **Daylight Filter** | - | In-built | In-built |
| **Maximum Collector Current:** | 20 mA | 100 mA | 50 mA |
| **Vf - Forward Voltage:** | 1.7 V | 1.25 V | 1.25 V |
| **Vr - Reverse Voltage:** | 5 V | 5 V | 5 V |
| **Minimum Operating Temperature:** | - 25 C | - 25 C | - 40 C |
| **Maximum Operating Temperature:** | + 85 C | + 85 C | + 85 C |
| **If - Forward Current:** | 50 mA | 60 mA | 50 mA |
| **Power Dissipation:** | 130 mW | 200mW | 100mW |
| **Unit Weight:** | 300 mg | 2.434 g | 2.502 g |

Selection factors for CNY70 for the project:

- The higher the wavelength the less red glow you will get emitting from the LED which is suitable this project.
- Inbuilt filter prevents interference caused by daylight.
- Lower Power Dissipation compared to OPB733TR.
- Low cost per unit.

# CHAPTER 3

## ABOUT OMR SCANNER

Optical Mark Recognition (**OMR**) is the process of capturing human marked data from document forms such as surveys and tests. OMR (Optical Mark Recognition) devices work with a dedicated scanner device that shines a beam of light onto the form paper. The contrasting reflectivity at predetermined positions on a page is then utilized to detect the marked areas because they reflect less light than the blank areas of the paper.

# CHAPTER 4

## 4.1 Development Board Features

- ATmega128 controller with external crystal of 16MHZ
- ATmega8 based USB programmer for ATmega128
- I2C communication lines
- SPI communication lines
- 16x2 alphanumeric LCD with contrast adjustment
- RTC with Backup Battery
- LDR sensor
- Buzzer (beneath the processor board)
- Onboard Motor Driver (L293D)
- LEDs for USB programmer ready indicator, Programming status indicator and Power ON indicator
- Test LEDs (open for interface with any PORT)
- Open interface Ports such as PORTF/ADC, PORTC
- USB Port for programming the Board
- Dual programmable UARTs
- Push Buttons for External Interrupts, Reset
- Modular design to permit replacement of processor board.
- JTAG interface

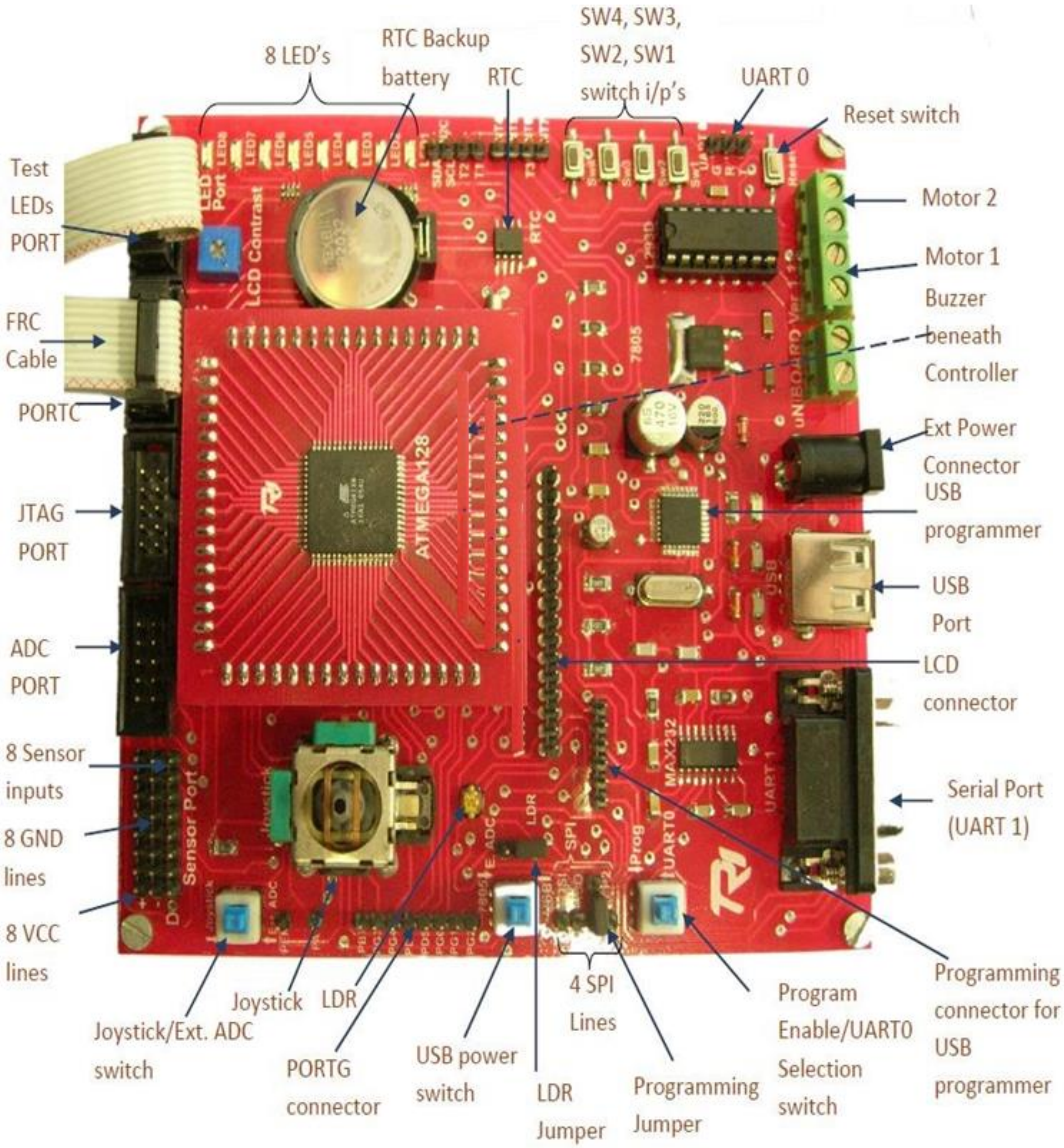



**FIGURE 1.** ***ATMEGA 128 CONTROLLER BOARD***

## 4.2 Atmega 128 microcontroller feature

- Advanced RISC Architecture
- 128K Bytes of In-System Self-programmable Flash program memory
- 4K Bytes EEPROM
- 4K Bytes Internal SRAM
- JTAG (IEEE std. 1149.1 Compliant) Interface
- Two 8-bit Timer/Counters with Separate pre-scalar and Compare Modes
- Two Expanded 16-bit Timer/Counters with Separate pre-scaler, Compare Mode and Capture Mode
- Real Time Counter with Separate Oscillator
- Two 8-bit PWM Channels
- 6 PWM Channels with Programmable Resolution from 2 to 16 Bits
- 8-channel, 10-bit ADC
- Byte-oriented Two-wire Serial Interface
- Dual Programmable Serial USARTs
- Master/Slave SPI Serial Interface
- Programmable Watchdog Timer with On-chip Oscillator
- Power-on Reset and Programmable Brown-out Detection
- External and Internal Interrupt Sources
- Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- Software Selectable Clock Frequency (using Fuse bits)
- 53 Programmable I/O Lines
- 4.5V - 5.5V for Atmega128 Operating Voltages
- 64lead TQFP

## 4.3 On-Board LED Accessibility

LED Patterns used to indicate the status of paper inserted inside the OMR scanner.

| **LED status** | **DESCRIPTION** |
|---|---|
| ALL OFF | OMR scanner is empty (NO paper inside) |
| ALTERNATE LEDs ON | Paper is inserted and currently scanner is performing scanning operation. |
| ALL ON (Few Seconds) | Scanning is completed. |
| Continuous Blinking | Paper Mismatch in the scanner. |

## 4.4 Internal ROM

- Scanning mainly includes verification of marked answers with predefined correct answers.
- These answers need storage section to store them.
- These answers need approx. 1kB space for storage.
- Development Board has 4kB EPROM.
- This purpose can be achieved by using this internal EPROM.

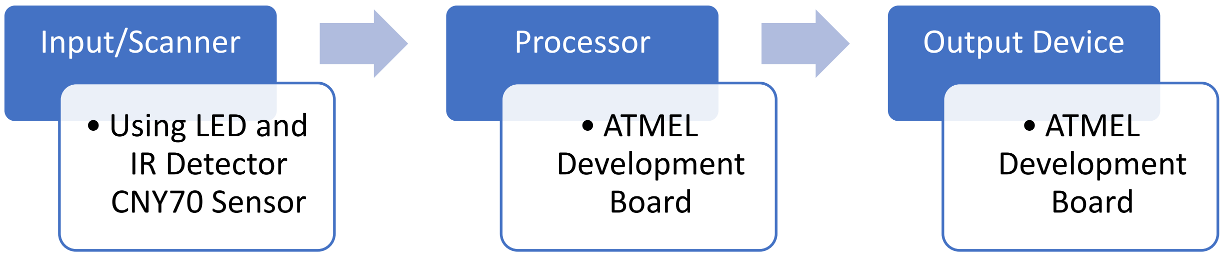
## 4.5 Buzzer Utilization

- When condition like paper mismatch arises inside the scanner, there should be warning alarm.
- This can be achieved using onboard Buzzer.
- In case of paper mismatch, it can display Error as an output.

## 4.6 LCD Interface

- The given LCD library contains the API's for printing character, string, values (decimal) on LCD, changing the LCD positions, clearing the LCD screen.
- Thus, the LCD can be used as an output device to display the outcome of the scanning process.
- After completing scanning one paper, LCD can be used to display Roll number of the student and number of correct answers student has marked.

# CHAPTER 5 HARDWARE DESIGN




## 5.1 Circuit Design

- The OMR Sheet is placed under the Scanner. Scanner accepts the sheet using the motor enabled Rubber Roller and Pressure Roller assembly.
- There are two sensors used to know the location of the OMR Sheet inside the Scanner. Once paper is under the sensor strip, the sensors are powered from processor.
- The Transmitter sends an IR beam. If the bubble under the sensor is darkened the beam would not be reflected from it. However, if it is not darkened, the beam gets reflected. In this way, the receiver receives the beam depending on whether the bubble was darkened or not.
- In sensor CYN 70, Transmitter is IR LED while Receiver is a Phototransistor. The output from the sensor is fed to the 8 channel ADC channel incorporated within the development board of ATMEGA128 Processor. For a question there are four possible answers. Hence, we need four sensors to read answer for a single question.
- The signals from four different Sensors are fed to four different ADC channels. The ADC converts the analog signal form the sensor to Digital value. This value lies within the range of 0 – 255. Hence, we get four different Digital Values.
- Software Program then decodes the marked answer form the four Digital Values recorded from the OMR Sheet. The decoded answer is then compared with the answers already stored in the EEPROM or the SRAM available with the processor.
- The fields storing number of correctly marked and wrongly marked answers are updated after each question is read from the OMR Sheet.
- This result can then be displayed on the LCD incorporated on the Development Board. This procedure is repeated for remaining questions on the OMR Sheet.
- At the end, final result can be declared on the LCD Display.

## 5.2 Microcontroller Section

### 5.2.1 Pin Diagram:

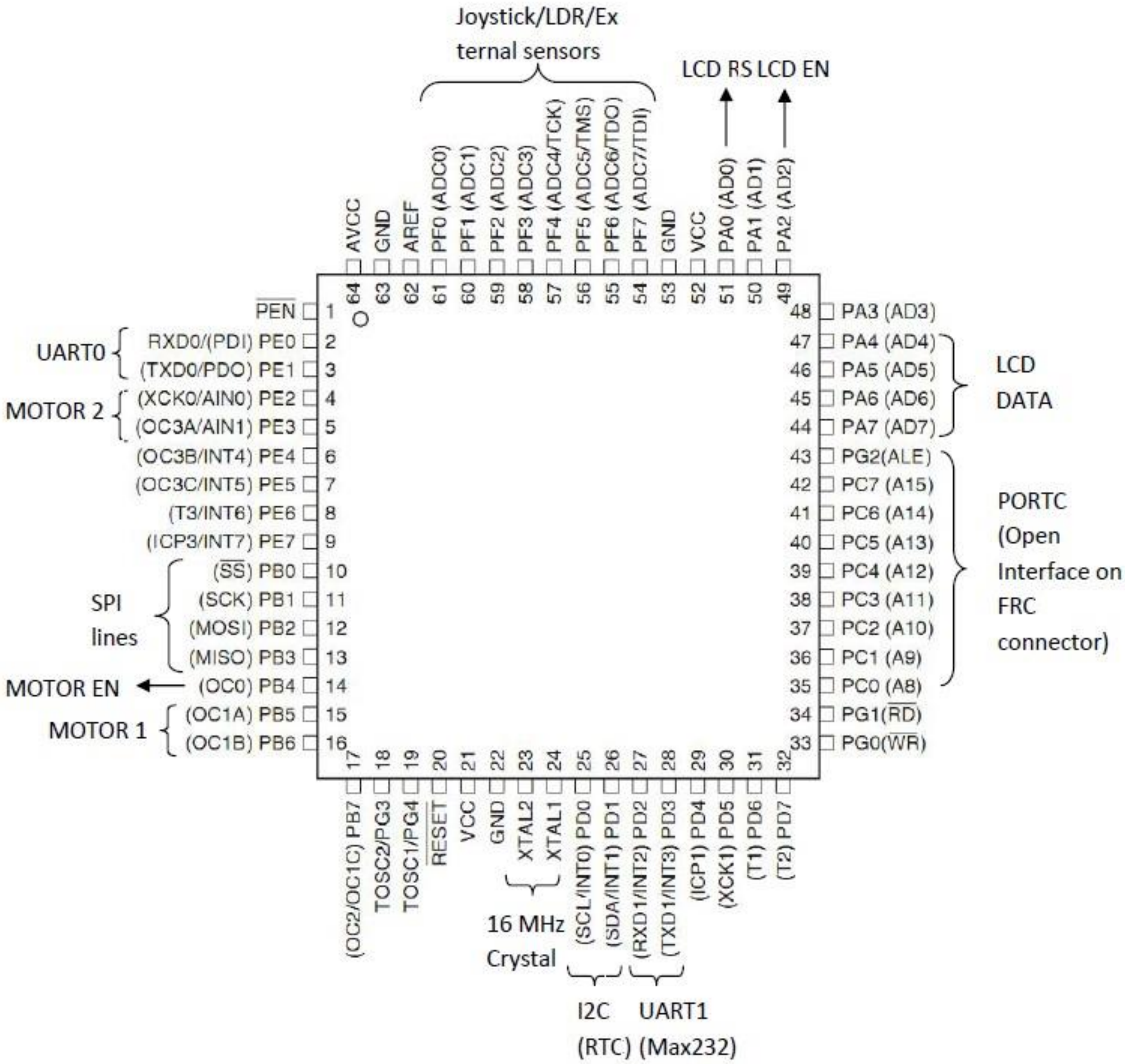



**FIGURE 2.** ***ATMEGA 128 CHIP PIN OUT DIAGRAM***

### 5.2.2. Schematic Diagram

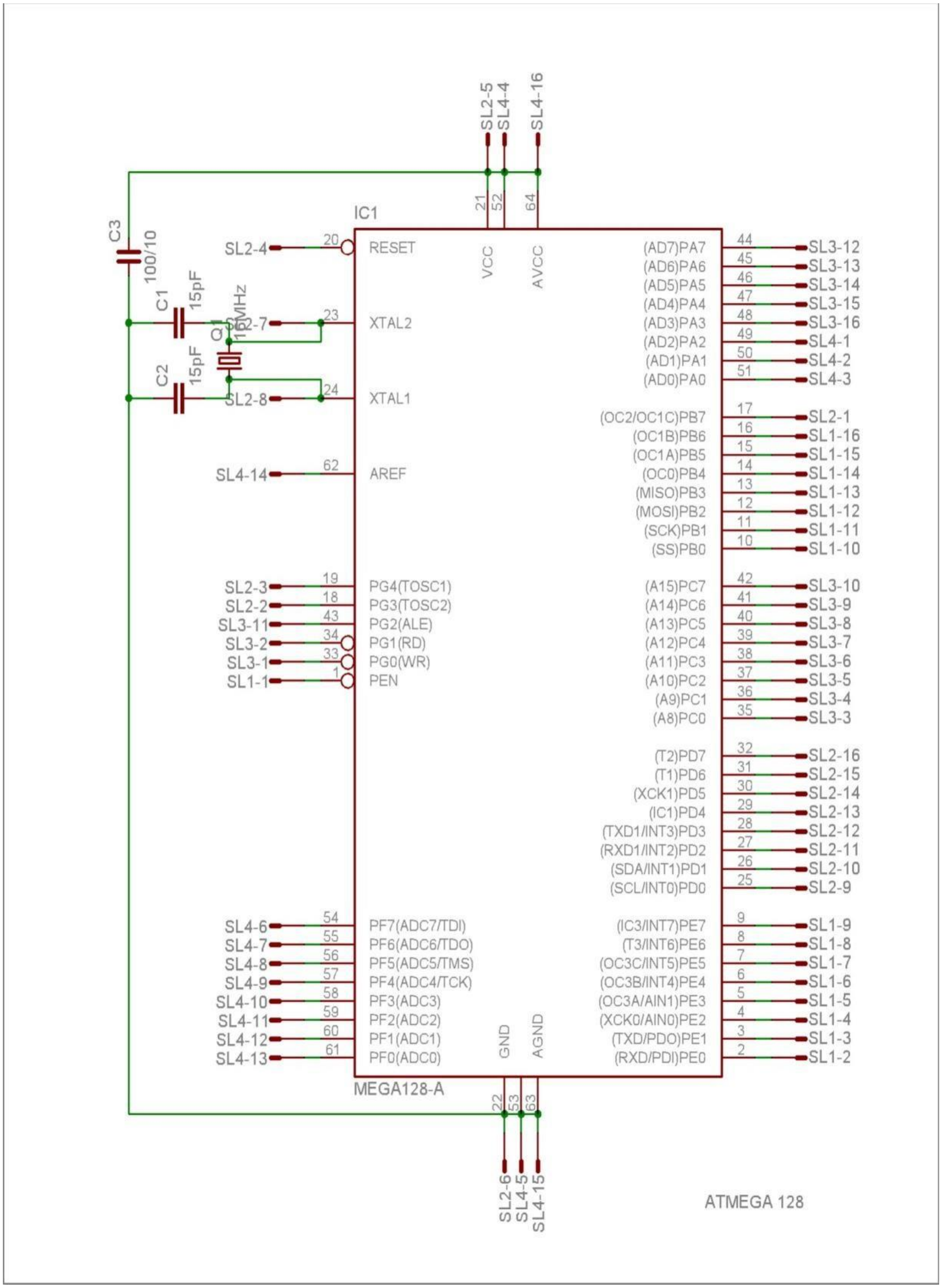



**FIGURE 3.** ***ATMEGA 128 CHIP CONNECTION ON THE BOARD***

## 5.3 LCD Section

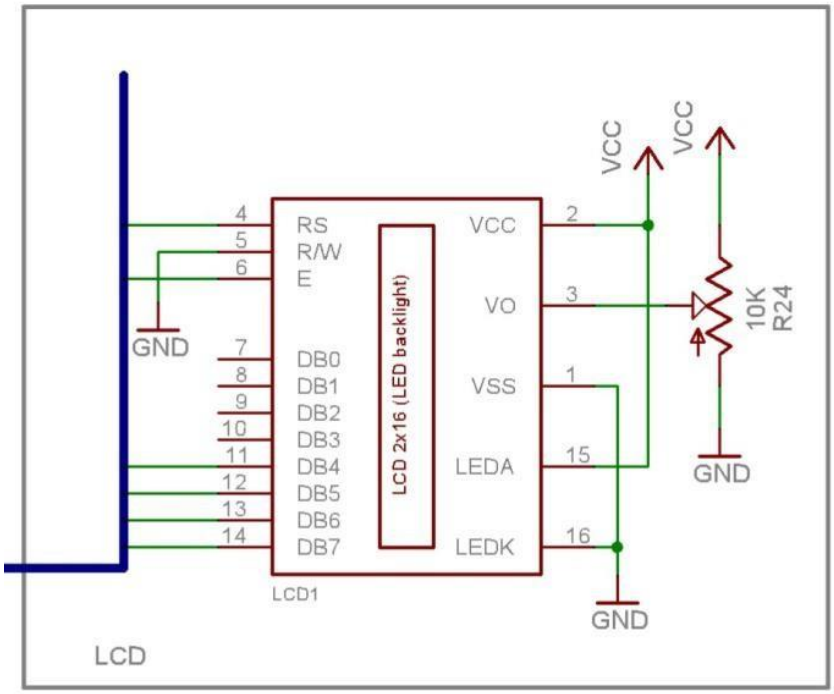



**FIGURE 4.** ***LCD CONNECTION TO MICROCONTROLLER CHIP SCHEMATIC***

The LCD used as an output device and placed on the board's LCD connector. The hardware connections are as follows:

- LCD control lines Enable (PD2)
- R/~W (Hardwired GND)
- RS (PD0)
- LCD data lines (PD4 – PD7).

Therefore, LCD is initialized using 4-line mode. The given LCD library contains the API's for printing character, string, values (decimal) on LCD, changing the LCD positions, clearing the LCD screen.

## 5.4 LED section

- The LEDS are connected to one of the general-purpose ports (PORTC) available of the ATmega128.
- These LEDs are connected using the FRC cable.
- The open FRC connectors are given on the development board so that their status can be monitored by interfacing them to LEDs using the FRC cable.

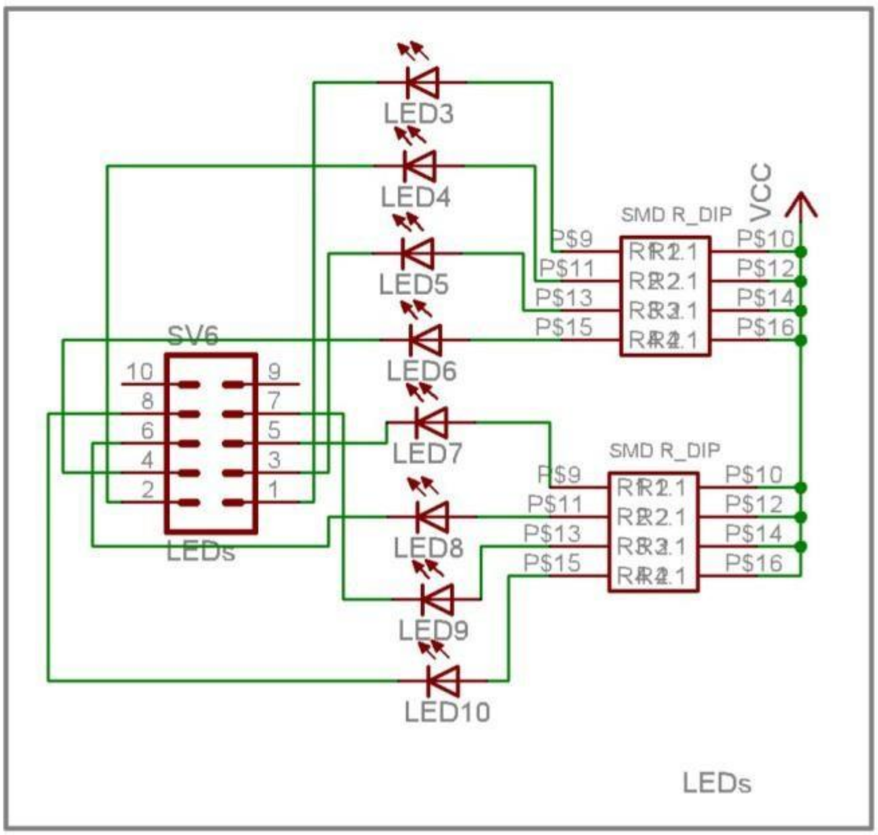



**FIGURE 5.** *LED CONNECTION TO MICROCONTROLLER CHIP SCHEMATIC*

## 5.5 Power Supply Section

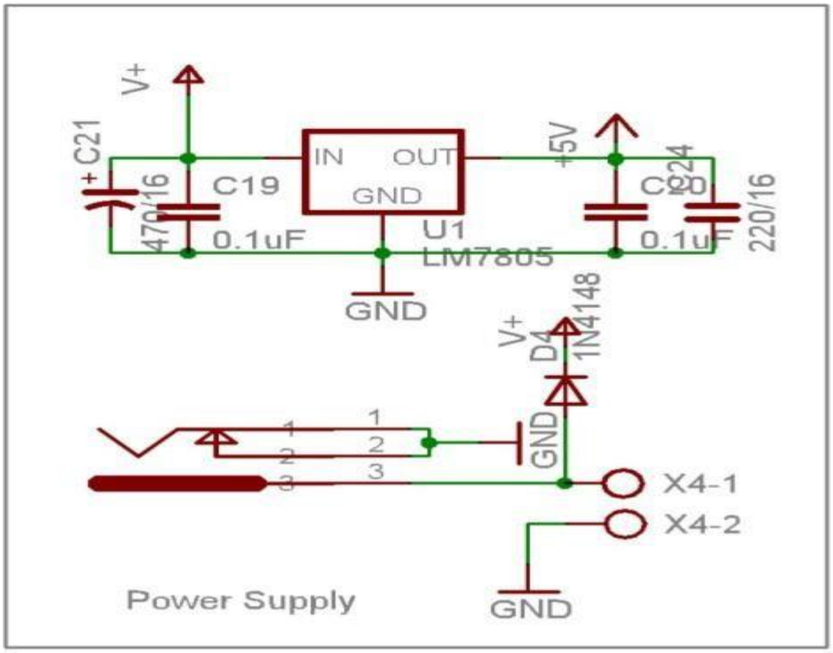



**FIGURE 6.** ***PS CONNECTION TO MICROCONTROLLER CHIP SCHEMATIC***

- It provides power to all the modules in the system, thereby supplying the various voltage requirements of each unit.

## 5.6 Sensor Ports

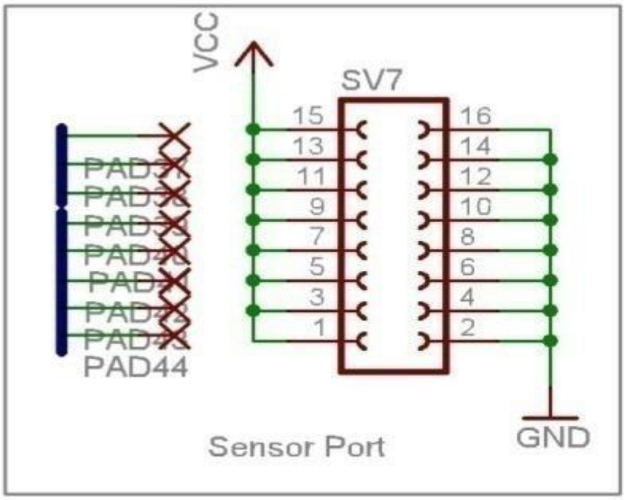



**FIGURE 7.** ***SENSOR PORT TO MICROCONTROLLER CHIP SCHEMATIC***

- ADC Inputs are given to the sensor port which is connected to the input channels of the ADC module.

## 5.7 USB Programmer Section

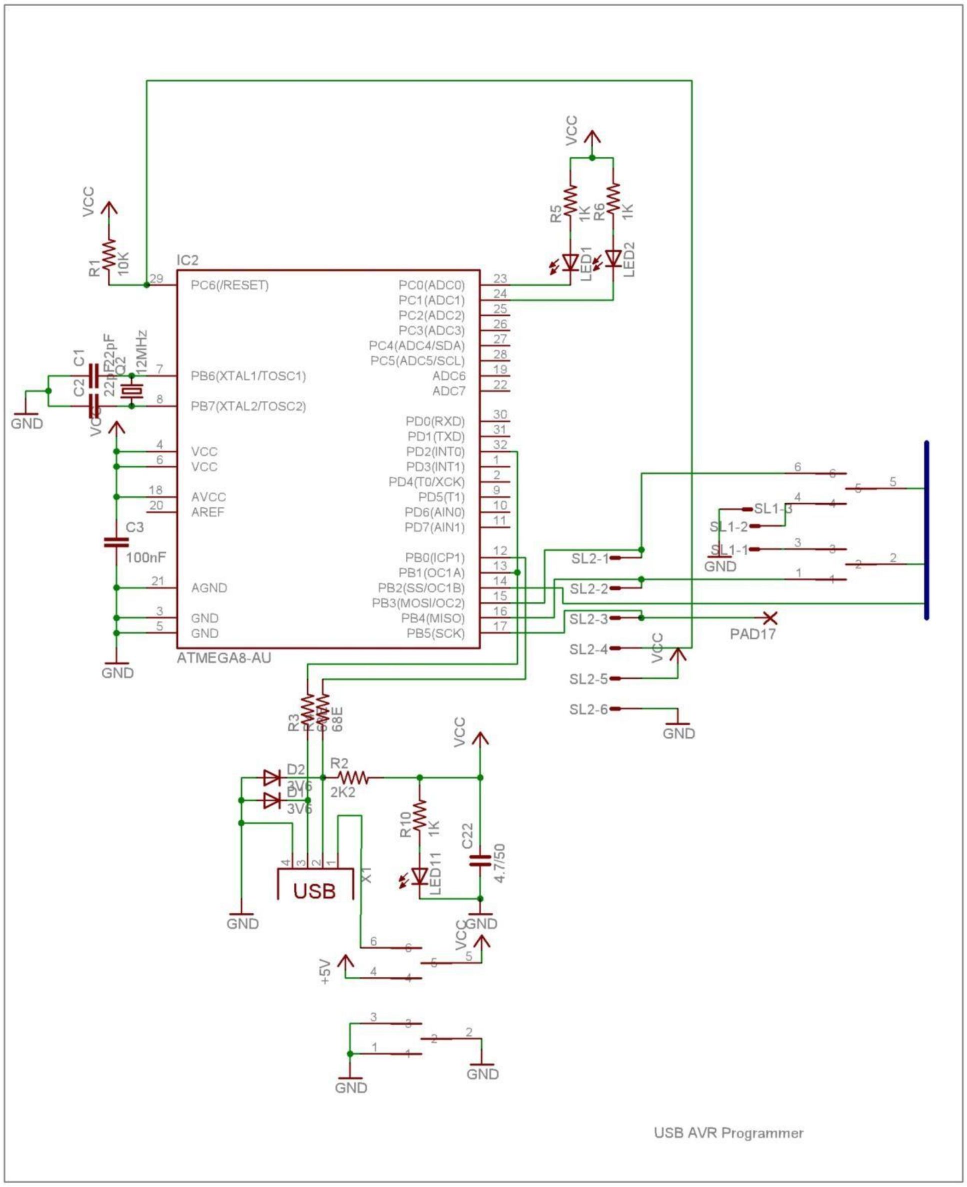



**FIGURE 8.** ***USB CONNECTION TO MICROCONTROLLER CHIP SCHEMATIC***

- The board has in-circuit programming feature (ISP) which has been used for programming through USB port.

## 5.8 Sensor Circuit

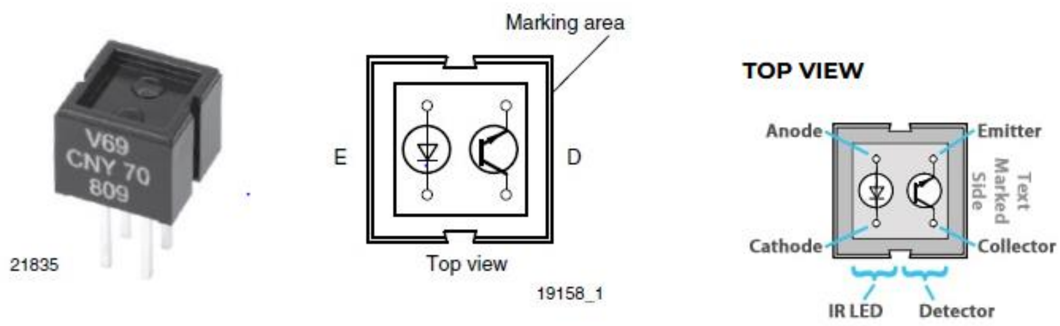



**FIGURE 9.** ***CNY70 PIN OUT DIAGRAM***

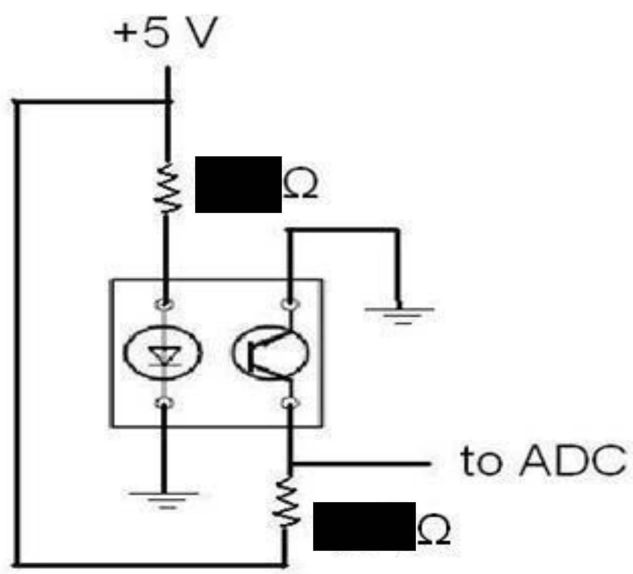



**FIGURE 10.** ***SINGLE INPUT SENSING CIRCUITRY USING CNY70 SENSOR***

- For each sensor there are 2 resistors for proper functioning of the sensor.
- A current limiting resistor for LED and a resistor at the collector of transistor. The sensor circuit consists of 8 CNY70 sensors.
- The output of the sensor is obtained at the collector of the phototransistor and fed to an ADC channel.
- The sensor PCB consists of 10 wire bus to contain the outputs of 8 sensors and 2 lines for power supply taken from microcontroller board.
- With this configuration the **output signal** is:
  - Black colour: 5V to ADC (Logic 1)
  - White colour: 0V to ADC (Logic 0)

# CHAPTER 6 SOFTWARE DESIGN

## 6.1 Embedded C Language

This Embedded C is extensive and contains many advanced concepts. The range of modules covers a full introduction to C, real-time and embedded systems concepts through to the design and implementation of real time embedded or standalone systems based on real-time operating systems and their device drivers. The Embedded C basics contains essential information for anyone developing embedded systems such as microcontrollers, real-time control systems, mobile device, PDAs and similar applications.

Performance is a key to signal-processing applications because it directly translates into end-user features. A 10-percent lower clock speed generally results in a corresponding reduction in power consumption. With more effective code generation, an application needs less processing cycles and thus a lower clock speed, which results in less EMI, longer battery life, and less heat generation.

Embedded C is not part of the C language as such. Rather, it is a C language extension that is the subject of a technical report by the ISO working group named "Extensions for the Programming Language C to Support Embedded Processors". It aims to provide portability and access to common performance-increasing features of processors used in the domain of DSP and embedded processing. The Embedded C specification for fixed-point, named address spaces, and named registers gives the programmer direct access to features in the target processor, thereby significantly improving the performance of applications. The hardware I/O extension is a portability feature of Embedded C.

C remains a very popular language for micro-controller developers due to the code efficiency and reduced overhead and development time. C offers low-level control and is considered more readable than assembly. Many free C compilers are available for a wide variety of development platforms. The compilers are part of an IDEs with ICD support, breakpoints, single-stepping and an assembly window. The performance of C compilers has improved considerably in recent years, and they are claimed to be more or less as good as assembly, depending on who you ask. Most tools now offer options for customizing the compiler optimization. Additionally, using C increases portability, since C code can be compiled for different types of processors.

## 6.2 Advantages of Embedded C Programming Language

Use of C in embedded systems is driven by following advantages:

- It is small and reasonably simpler to learn, understand, program and debug.
- C Compilers are available for almost all embedded devices in use today, and there is a large pool of experienced C programmers.
- Unlike assembly, C has advantage of processor-independence and is not specific to any particular microprocessor/ microcontroller or any system. This makes it convenient for a user to develop programs that can run on most of the systems.
- As C combines functionality of assembly language and features of high-level languages, C is treated as a ‘middle-level computer language’ or ‘high level assembly language’
- It is fairly efficient
- It supports access to I/O and provides ease of management of large embedded projects.
- Compared to other high-level languages, C offers more flexibility because C is relatively small, structured language; it supports low-level bit-wise data manipulation.
- Compared to assembly language, C Code written is more reliable and scalable, more portable between different platforms (with some changes). Moreover, programs developed in C are much easier to understand, maintain and debug. Also, as they can be developed more quickly, codes written in C offers better productivity. C is based on the philosophy ‘programmers know what they are doing’; only the intentions are to be stated explicitly. It is easier to write good code in C & convert it to an efficient assembly code (using high quality compilers) rather than writing an efficient code in assembly itself. Benefits of assembly language programming over C are negligible when we compare the ease with which C programs are developed by programmers.

## 6.3 Programming Code

```
#include<avr/io.h>

#include<LCD_mega128.h>

#include<util/delay.h>

unsigned int adc_result=0;

unsigned char total_marked=0;

unsigned char marked_ans[9];


#define lim1 63

#define lim2 145

#define lim3 150

#define lim4 162

#define lim5 210

#define lim6 210

#define lim7 145

#define lim8 195

#define rows 5


void adc_init()
```

```
{
ADMUX=[redacted]        // channel 0 is selected , 5V reference.
ADCSRA=[redacted]       //ADC is turned ON, start conversion, free running mode, ADC
interrupt enable
ADCSRA=[redacted]       // ADC ON, Free Running OFF, ADC interrupt enable
}
void adc_read()
{
ADCSRA=[redacted]       // Free Running mode, Start Conversion
for(char i=0;i<5;i++)
{
while(!(ADCSRA&[redacted]       // while ADIF=0, stay here
[redacted]       // clearing ADIF
adc_result+=ADCH;
}
adc_result/=5;
ADCSRA=ADCSRA&[redacted]       // Free Running ADC is turned OFF
}


void decision (unsigned int adc_result, unsigned char ch)
```

```
{
    switch(ch)
    {
  case 0: lcd_gotoxy1(0);
            if(adc_result>lim1)
            marked_ans[1]=1;
            break;
    case 1: lcd_gotoxy1(4);
              if(adc_result>lim2)
              marked_ans[2]=1;
              break;
    case 2: lcd_gotoxy1(8);
              if(adc_result>lim3)
              marked_ans[3]=1;
              break;
    case 3: lcd_gotoxy1(12);
              if(adc_result>lim4)
              marked_ans[4]=1;
              break;
    case 4: lcd_gotoxy2(0);
```

```
{
        if(adc_result>lim5)
        marked_ans[5]=1;
        break;
    case 5: lcd_gotoxy2(4);
        if(adc_result>lim6)
        marked_ans[6]=1;
        break;
    case 6: lcd_gotoxy2(8);
        if(adc_result>lim7)
        marked_ans[7]=1;
        break;
    case 7: lcd_gotoxy2(12);
        if(adc_result>lim8)
        marked_ans[8]=1;
        break;
    }
      lcd_showvalue((char))
adc_result);
}
void main()
{
```

```
unsigned char channel option;

unsigned char correct_ans1=2;

unsigned char correct_ans2=1;

unsigned char correct[12]= {'c','o','r','r','e','c','t'};

unsigned char wrong[12]={'w','r','o','n','g'};


DDRC=[redacted]                    // PORTC is selected as Output

PORTC=[redacted]                 // All LEDs off

lcd_init();

adc_init();

lcd_gotoxy1(0);

lcd_string("OMR scanner");

_delay_ms(1000);

PORTC=[redacted]                 // All LEDs turned off

PORTC=PORTC<<1;

lcd_[redacted]    // Clear LCD screen display

lcd_gotoxy1(0);

for (unsigned char i=0;i<rows;i++)

{

for(channel=0;channel<8;channel++)

{
```

```
ADCSRA=ADCSRA&[redacted]                    // Free Running OFF

ADMUX=(ADMUX&[redacted]                    //Changing channel

adc_result=0;

adc_read();

decision(adc_result,channel);

PORTC=PORTC<<1;

_delay_ms(1000);

 }

 lcd_init();

 lcd_[redacted];

 PORTC=[redacted]                              // All LEDs turned off

 for(char i=1;i<5;i++)

 total_marked+=marked_ans[i];

 if(total_marked==1)

 {

 for(char option=1;option<5;option++)

 {

 if(marked_ans[option]==1)              // 'option' will have marked option no.

 break;
```

```
    }

    if(correct_ans1==option)

    {
    PORTC[redacted]//LED 1 OFF
    lcd_gotoxy1(3);

    lcd_string(correct);

    }
}
else

{
PORTC[redacted]                    // LED 1 ON

lcd_gotoxy1(3);

lcd_string(wrong);

}

total_marked=0;

for(char i=5;i<9;i++)

total_marked+=marked_ans[i];

if(total_marked==1)

{

for(char option=5;option<9;option++)

{

if(marked_ans[option]==1)                // 'option' will have marked option no.

break;
```

```
}

if(correct_ans2==(option-4))

{

PORTC[redacted] LED 8 OFF

lcd_gotoxy2(3);

lcd_string(correct);

}

}

else

{

PORTC&=[redacted]            // LED 8 turned ON

lcd_gotoxy2(3);

lcd_string(wrong);

}

while(1);

}
```

# CHAPTER 7

## RESULT

The digital values of all eight channels were displayed on the LCD and then the LEDs would indicate whether the answer is correct or incorrect. The indication of incorrect answer is given by switching ON the corresponding LED on the board. At the same time, Correct or Wrong is displayed on the LCD.

Detection of sheet by system: Successful.
Detection of Each Marks by System: Successful.
Matching of detected marks with Predefined Marks: Successful.

# CHAPTER 8

## FUTURE SCOPE

1. Scanner can be modified so that it can scan variable size OMR Sheets. Sensor positions can be varied on the strip depending upon the size of the bubbles on the datasheet.
2. Provisions can be made such that user can insert his own set of answers.
3. Software Program can be modified so that it can display the correct answers of all questions on User Demand or in the end of Scanning Process.
4. Faster Algorithms for Scanning as well as for corrections can be developed to improve the efficiency of the Scanner.

# CHAPTER 9

## ACRONYMS

| | |
|---|---|
| API | Application Programming Interface |
| OMR | Optical Mark Recognition |
| CTR | Current Transfer Ratio |
| LDR | Light-Dependent Resistor |
| UART | Universal Asynchronous Receiver / Transmitter |
| RTC | Real Time Clock |
| SPI | Serial Peripheral Interface |
| I2C | Inter-Integrated Circuit |
| PWM | Pulse Width Modulation |
| RISC | Reduced Instruction Set Computer |
| FRC | Flat Ribbon Cable |
| ADC | Analog to Digital Converter |
| EEPROM | Electrically Erasable Programmable Read-Only Memory |
| LED | Light Emitting Diode |
| LCD | Liquid Crystal Display |

# CHAPTER 10

## APPENDIX

### References

#### WEBSITES:


www.atmel.com
www.avrfreaks.net
www.maxim-ic.com
http://en.wikibooks.org/wiki/Embedded_Systems/Atmel_AVR


#### BOOKS:


1. Micro C/OS-II by JEAN J. LABROSSE
2. Programming & Customizing the AVR Microcontroller By Dhananjay V. Gadre
3. Embedded C Programming and the Atmel AVR by Barnett and Cox
4. AVR RISC Microcontroller Handbook by Atmel


#### PUBLICATIONS:

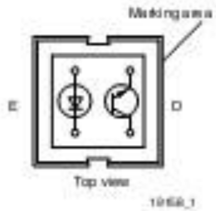

1. Optical mark reading; A. M. Smith, In Proceedings of the 9th annual ACM SIGUCCS conference on User services, United States, 1981, pp: 257263.
2. OCR and OMR Are Hotter Than Ever; Moore B, http://www.idat.com/a-ocromr.html
3. Optical Mark Recognition; Bergeron B P, Postgraduate Medicine, 104(2): 23-24


#### DATASHEETS:


1. Sensor CNY70
2. Atmega 128 Microcontroller

# Reflective Optical Sensor with Transistor Output


21835




## DESCRIPTION

The CNY70 is a reflective sensor that includes an infrared emitter and phototransistor in a leaded package which blocks visible light.

## FEATURES

- Package type: leaded
- Detector type: phototransistor
- Dimensions (L x W x H in mm): 7 x 7 x 6
- Peak operating distance: < 0.5 mm
- Operating range within > 20 % relative collector current: 0 mm to 5 mm
- Typical output current under test: $I_C$ = 1 mA
- Emitter wavelength: 950 nm
- Daylight blocking filter
- Lead (Pb)-free soldering released
- Compliant to RoHS directive 2002/95/EC and in accordance to WEEE 2002/96/EC

e4
RoHS
COMPLIANT

## APPLICATIONS

- Optoelectronic scanning and switching devices i.e., index sensing, coded disk scanning etc. (optoelectronic encoder assemblies).

| PRODUCT SUMMARY | | | | |
|---|---|---|---|---|
| PART NUMBER | DISTANCE FOR MAXIMUM $CTR_{rel}$ (1) (mm) | DISTANCE RANGE FOR RELATIVE $I_{out}$ > 20 % (mm) | TYPICAL OUTPUT CURRENT UNDER TEST (2) (mA) | DAYLIGHT BLOCKING FILTER INTEGRATED |
| CNY70 | 0 | 0 to 5 | 1 | Yes |

**Notes**
(1) CTR: current transfere ratio, $I_{out}/I_{in}$
(2) Conditions like in table basic charactristics/sensors

| ORDERING INFORMATION | | | |
|---|---|---|---|
| ORDERING CODE | PACKAGING | VOLUME (1) | REMARKS |
| CNY70 | Tube | MOQ: 4000 pcs, 80 pcs/tube | - |

**Note**
(1) MOQ: minimum order quantity

| ABSOLUTE MAXIMUM RATINGS (1) | | | | |
|---|---|---|---|---|
| PARAMETER | TEST CONDITION | SYMBOL | VALUE | UNIT |
| **COUPLER** | | | | |
| Total power dissipation | $T_{amb}$ ≤ 25 °C | $P_{tot}$ | 200 | mW |
| Ambient temperature range | | $T_{amb}$ | - 40 to + 85 | °C |
| Storage temperature range | | $T_{stg}$ | - 40 to + 100 | °C |
| Soldering temperature | Distance to case 2 mm, t ≤ 5 s | $T_{sd}$ | 260 | °C |
| **INPUT (EMITTER)** | | | | |
| Reverse voltage | | $V_R$ | 5 | V |
| Forward current | | $I_F$ | 50 | mA |
| Forward surge current | $t_p$ ≤ 10 µs | $I_{FSM}$ | 3 | A |
| Power dissipation | $T_{amb}$ ≤ 25 °C | $P_V$ | 100 | mW |
| Junction temperature | | $T_j$ | 100 | °C |

| ABSOLUTE MAXIMUM RATINGS [1] | | | | |
|---|---|---|---|---|
| **PARAMETER** | **TEST CONDITION** | **SYMBOL** | **VALUE** | **UNIT** |
| **OUTPUT (DETECTOR)** | | | | |
| Collector emitter voltage | | $V_{CEO}$ | 32 | V |
| Emitter collector voltage | | $V_{ECO}$ | 7 | V |
| Collector current | | $I_C$ | 50 | mA |
| Power dissipation | $T_{amb} \le 25\ °C$ | $P_V$ | 100 | mW |
| Junction temperature | | $T_j$ | 100 | °C |

**Note**

[1] $T_{amb}$ = 25 °C, unless otherwise specified

## ABSOLUTE MAXIMUM RATINGS

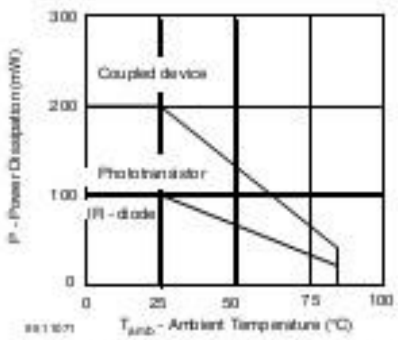



Fig. 1 - Power Dissipation vs. Ambient Temperature

| BASIC CHARACTERISTICS [1] | | | | | | |
|---|---|---|---|---|---|---|
| **PARAMETER** | **TEST CONDITION** | **SYMBOL** | **MIN.** | **TYP.** | **MAX.** | **UNIT** |
| **COUPLER** | | | | | | |
| Collector current | $V_{CE}$ = 5 V, $I_F$ = 20 mA, d = 0.3 mm (figure 1) | $I_C$ [2] | 0.3 | 1.0 | | mA |
| Cross talk current | $V_{CE}$ = 5 V, $I_F$ = 20 mA, (figure 2) | $I_{CX}$ [3] | | | 600 | nA |
| Collector emitter saturation voltage | $I_F$ = 20 mA, $I_C$ = 0.1 mA, d = 0.3 mm (figure 1) | $V_{CEsat}$ [2] | | | 0.3 | V |
| **INPUT (EMITTER)** | | | | | | |
| Forward voltage | $I_F$ = 50 mA | $V_F$ | | 1.25 | 1.6 | V |
| Radiant intensity | $I_F$ = 50 mA, $t_p$ = 20 ms | $I_e$ | | | 7.5 | mW/sr |
| Peak wavelength | $I_F$ = 100 mA | $\lambda_P$ | 940 | | | nm |
| Virtual source diameter | Method: 63 % encircled energy | d | | 1.2 | | mm |
| **OUTPUT (DETECTOR)** | | | | | | |
| Collector emitter voltage | $I_C$ = 1 mA | $V_{CEO}$ | 32 | | | V |
| Emitter collector voltage | $I_E$ = 100 µA | $V_{ECO}$ | 5 | | | V |
| Collector dark current | $V_{CE}$ = 20 V, $I_F$ = 0 A, E = 0 lx | $I_{CEO}$ | | | 200 | nA |

**Notes**

[1] $T_{amb}$ = 25 °C, unless otherwise specified

[2] Measured with the "Kodak neutral test card", white side with 90 % diffuse reflectance

[3] Measured without reflecting medium

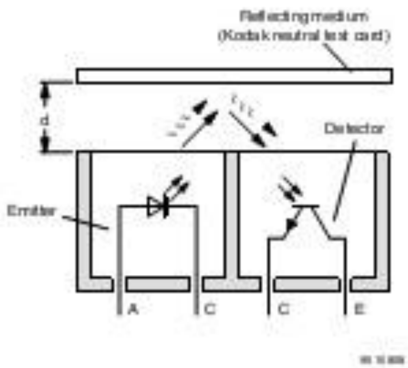



Fig. 2 - Pulse diagram

## BASIC CHARACTERISTICS

$T_{amb}$ = 25 °C, unless otherwise specified

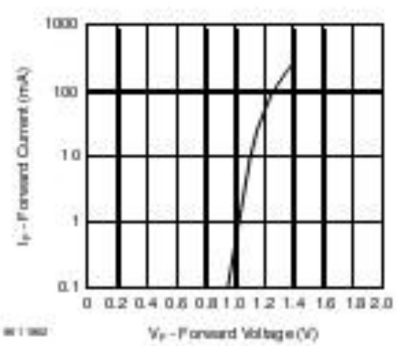



Fig. 3 - Forward Current vs. Forward Voltage

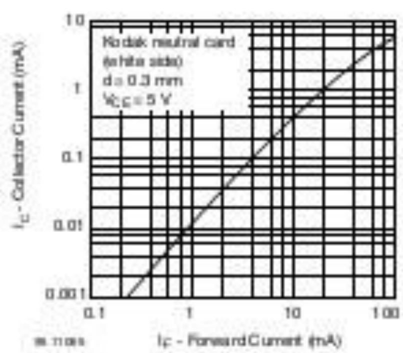



Fig. 5 - Collector Current vs. Forward Current

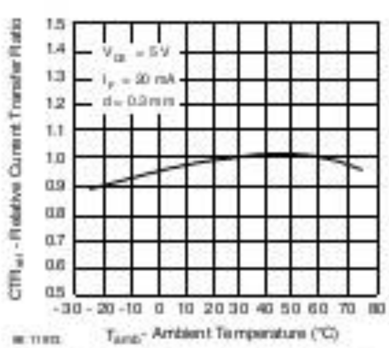



Fig. 4 - Relative Current Transfer Ratio vs. Ambient Temperature

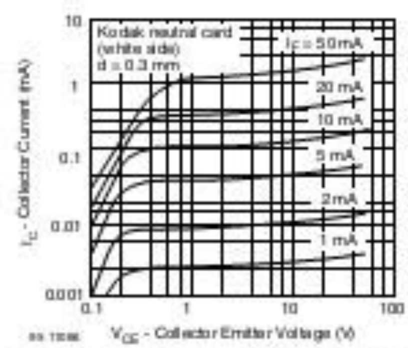



Fig. 6 - Collector Current vs. Collector Emitter Voltage

## Features

- High-performance, Low-power AVR® 8-bit Microcontroller
- Advanced RISC Architecture
  - 133 Powerful Instructions – Most Single Clock Cycle Execution
  - 32 x 8 General Purpose Working Registers + Peripheral Control Registers
  - Fully Static Operation
  - Up to 16 MIPS Throughput at 16 MHz
  - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory segments
  - 128K Bytes of In-System Self-programmable Flash program memory
  - 4K Bytes EEPROM
  - 4K Bytes Internal SRAM
  - Write/Erase cycles: 10,000 Flash/100,000 EEPROM
  - Data retention: 20 years at 85°C/100 years at 25°C[1]
  - Optional Boot Code Section with Independent Lock Bits
    In-System Programming by On-chip Boot Program
    True Read-While-Write Operation
  - Up to 64K Bytes Optional External Memory Space
  - Programming Lock for Software Security
  - SPI Interface for In-System Programming
- JTAG (IEEE std. 1149.1 Compliant) Interface
  - Boundary-scan Capabilities According to the JTAG Standard
  - Extensive On-chip Debug Support
  - Programming of Flash, EEPROM, Fuses and Lock Bits through the JTAG Interface
- Peripheral Features
  - Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes
  - Two Expanded 16-bit Timer/Counters with Separate Prescaler, Compare Mode and Capture Mode
  - Real Time Counter with Separate Oscillator
  - Two 8-bit PWM Channels
  - 6 PWM Channels with Programmable Resolution from 2 to 16 Bits
  - Output Compare Modulator
  - 8-channel, 10-bit ADC
    8 Single-ended Channels
    7 Differential Channels
    2 Differential Channels with Programmable Gain at 1x, 10x, or 200x
  - Byte-oriented Two-wire Serial Interface
  - Dual Programmable Serial USARTs
  - Master/Slave SPI Serial Interface
  - Programmable Watchdog Timer with On-chip Oscillator
  - On-chip Analog Comparator
- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated RC Oscillator
  - External and Internal Interrupt Sources
  - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
  - Software Selectable Clock Frequency
  - ATmega103 Compatibility Mode Selected by a Fuse
  - Global Pull-up Disable
- I/O and Packages
  - 53 Programmable I/O Lines
  - 64-lead TQFP and 64-pad QFN/MLF
- Operating Voltages
  - 2.7 - 5.5V for ATmega128L
  - 4.5 - 5.5V for ATmega128
- Speed Grades
  - 0 - 8 MHz for ATmega128L
  - 0 - 16 MHz for ATmega128

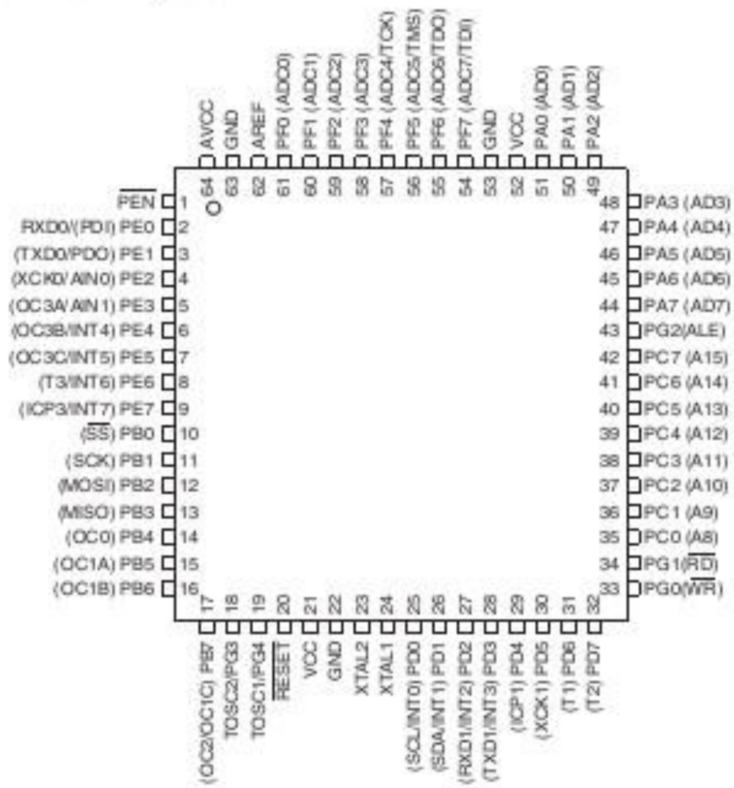
**8-bit AVR® Microcontroller with 128K Bytes In-System Programmable Flash**

**ATmega128**
**ATmega128L**

Rev. 2467P-AVR-08/07

## Pin Configurations

**Figure 1.** Pinout ATmega128




Note: The Pinout figure applies to both TQFP and MLF packages. The bottom pad under the QFN/MLF package should be soldered to ground.

## Overview

The ATmega128 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 ATmega128 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

## Block Diagram

**Figure 2.** Block Diagram