



HNC ELECTRICAL/ELECTRONIC ENGINEERING, POWER SYSTEMS ENGINEERING & INSTRUMENTATION ENGINEERING

Embedded Systems PCEE4004

Abstract

The Arduino is the brains of the system. It is a micro controller board based on a removable, dual-inline-package (DIP) ATmega328 AVR microcontroller. There are 20 pins which can be utilized as digital inputs and outputs.

Christopher Bendall
155846@pembrokeshire.ac.uk

Contents

Introduction	1
Task 1	1
Arduino Uno R3	2
RGB LED	2
Resistors	2
Piezo-Buzzer	2
LCD 16x2	2
Ultrasonic Distance Sensor	3
Task 2 – Software Design	3
Task 3 – Implementation	3
Task 4 – Testing	4
Task 5 – Discussion, Evaluation and Conclusions	6
List of Appendices	7
Bibliography	7

Introduction

You are required to design and emulate a simple alarm system that will set off an alarm if it detects an object within a 2m vicinity of its sensor.

Your microcontroller-based system should continue to indicate the object is within 2m of the sensor, until the object moves out of its range. An LED could be used to indicate that an object has been detected, however you may use any components you wish in your system.

The scope of the system required to satisfy the criteria for the assignment is fairly vague so offers a certain degree of poetic licence. I will be designing a programme on Tinkercad (online coding tool) which will then be able to be loaded on to an Arduino Uno R3 micro controller to function as intended. I have modelled my design around a ‘social distancing device’ that will sound when somebody or something comes within two metres of our person- the device will be wearable. This application is topical and simple. I will make use of audible and visual aids to warn our user when the two metre zone is not clear and visual aids to tell us the area is safe. This will be done using LEDs and buzzers, along with an ultra-sonic sensor.

Task 1

Design a microcontroller based system. Explain the function and purpose of each of the components in the system. Add the necessary datasheets to the back of the report as an appendix and refer to them in your text.

**All data-sheets attached, appendices 1-7*

Arduino Uno R3

The Arduino is the brains of the system. It is a micro controller board based on a removable, dual-inline-package (DIP) ATmega328 AVR microcontroller. There are 20 pins which can be utilised as digital inputs and outputs. 6 of these pins are labelled as PWM, indicating then can be used for pulse width modulation and 6 can be used as analogue inputs. It operates at a voltage of 5V with a maximum input of 20V, as seen on the data sheet. The code we develop on the website 'Tinkercad' will be loaded onto the Arduino allowing us to achieve our desired function.

RGB LED

LED stands for Light Emitting Diode. In short, an LED is a semiconductor the lights up when a voltage is passed through it in the correct direction. LEDs use little energy and have a long life expectancy making them a great choice for a strong visual aid when our system is alarming. Our RGB LED is especially good as we can achieve Red, green or blue illumination. I will use green for clear and red for danger. For compatibility our LEDs will be rated to 20mA, this is where we need a resistor.

Resistors

I have placed a 250 Ω resistor in series with the LED and a 250 Ω resistor in series with the feed to our LCD backlight. This is to ensure the LED does not receive too much power and cause damage. Simple ohms law can be used to decide the size of the resistor used. We are looking to achieve 20mA.

Therefore we use $V=I \times R$ which transposes to $R = V/I$.
 $5/0.02 = 250 \Omega$.

Piezo-Buzzer

A Piezo-buzzer is an audible device which will sound as an alarm when our 2m perimeter is encroached upon. These buzzers use piezoelectricity, which is created using a material that generates an electric charge when it's physically deformed by a vibration, sound wave, or mechanical strain. In this case vibration is caused when we pass a voltage through the buzzer; different sounds can be achieved by varying the signal we send out. It is a good choice to use with our Arduino as there is a huge range of tones that can be programmed onto it via Tinkercad.

LCD 16x2

LCD stands for Liquid Crystal Display. It is the display screen used to display messages we can read, obviously in this case helping us understand the current status of the system. The LCDs have a parallel interface, meaning that the microcontroller has to manipulate several interface pins at once to control the display. The interface consists of the following pins:

- An RS pin (Register Select). This controls the LCD's memory in which you send data to. It allows you to either select the data register or an instruction register.
- A Read/Write (R/W) pin that selects reading mode or writing mode
- An Enable pin that enables writing to the registers
- 8 data pins (D0 -D7). The states of these pins (high or low) are the bits that you're writing to a register when you write, or the values you're reading when you read.

There's also a display contrast pin (Vo), power supply pins (+5V and Gnd) and LED Backlight (Bklt+ and Bklt-) pins that you can use to power the LCD, control the display contrast, and turn on and off the LED backlight, respectively.

Ultrasonic Distance Sensor

Our ultrasonic distance sensor uses ultrasonic sound waves to gauge the distance of nearby objects. An ultrasonic sensor uses a transducer to send and receive ultrasonic pulses that relay back information about an object's proximity. Through our coding, we can programme the sensor to send and receive signals at timed intervals (every second for this application). For this application the sensor fulfils all the criteria we need. The alarms, lights and LCDs will be programmed to work on the signals received back from the sensor so this is a vital component for us in our alarm system.

Task 2 – Software Design

Design a C program to detect if an object is within a 2m vicinity of the sensor. Your system should alert the user if an object is detected.

Explain how the software will control the hardware in your system.

See flow chart attached, appendix 7.

In my system, everything goes through the Arduino. After energising, our loop begins. Firstly, an output signal is sent to our ultra-sonic sensor. This triggers a 'ping' to be sent out. The sensor will then return a signal to the Arduino with how long the ping took to reverberate from the nearest object. This reading in milli-seconds is then converted into a distance measure, as written in my code.

Line 45: distance=microseconds/29/2; //TIME INTO DISTANCE

With this distance measure input, all other accessories in the circuit will work accordingly through 'IF' statements. If two metres or under, our alarm will be triggered. This will be both a buzzer a red LED. If everything is clear, we will hear no sound and see a green LED.

Line 49: if (distance <= 200 && distance >1) //IF PARAMETER IS BREACHED

I have included &&>1 as I have been experimenting with an 'off' switch in which the reading returned is 0. If the distance is over two metres however, I have an 'ELSE' statement in place on line 64. This will cover distances greater than two meters, basically normal operation. Here we will see just a green LED.

Task 3 – Implementation

Create your alarm system, implementing the designs used in the previous tasks. If you have to alter your designs during this stage, explain why the changes are required.

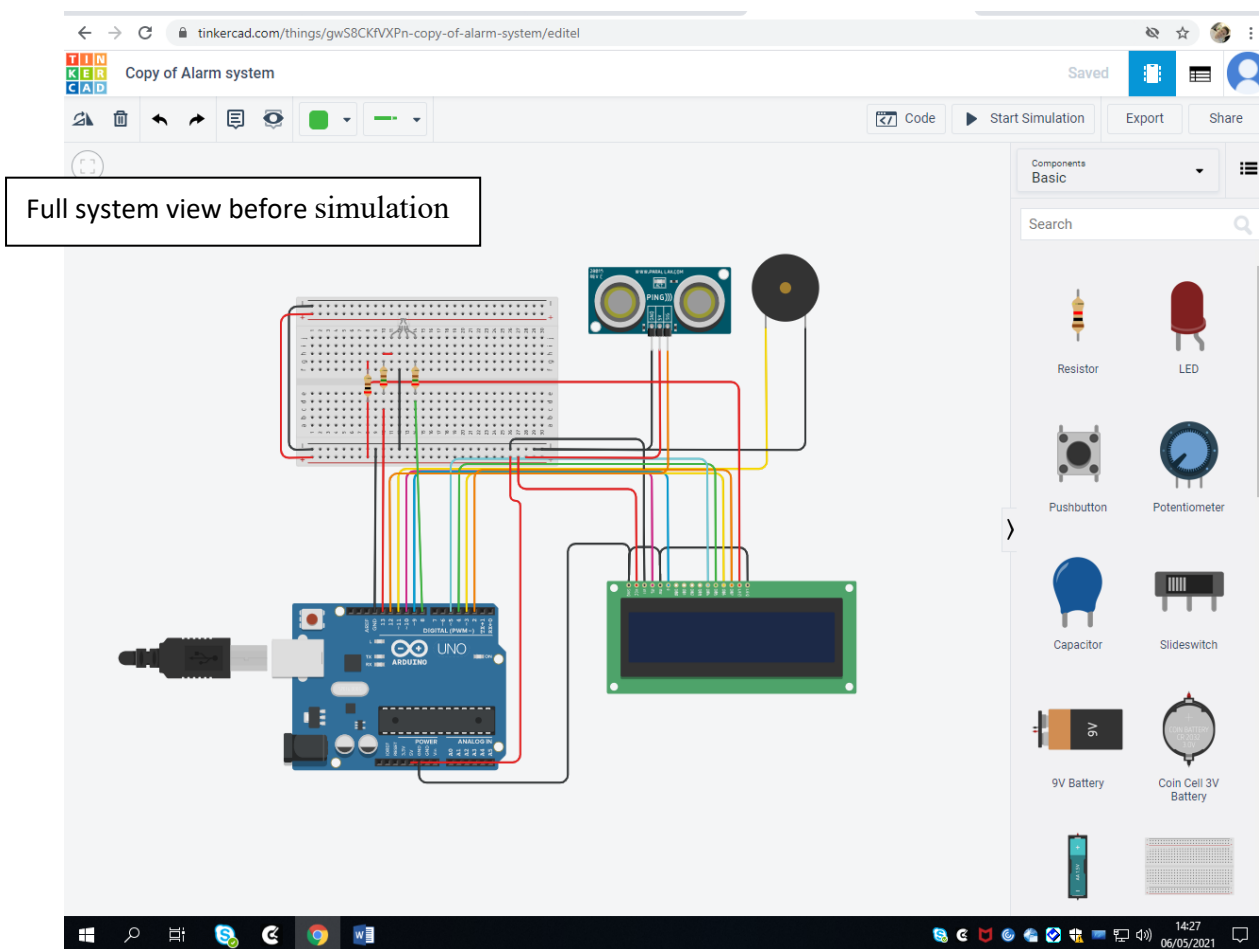
Marks will be awarded for functionality, adherence to programming conventions, quality and efficiency of code. Higher marks will be awarded for the use of your own defined functions in the program making it more readable.

Fully document your C program with comments explaining how it works and how it relates to your algorithm. Get a clear program printout, making sure each in-comment does not overflow onto the next line.

Task 4 – Testing

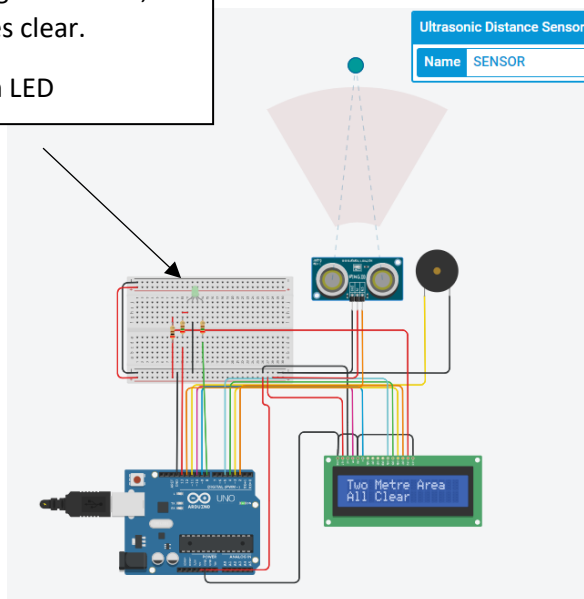
Test your program to see how much of it works, using a structured approach. Record your results using suitable screen dumps and photographs. Ensure all eventualities are tested and evidenced.

See appendix 8 for full testing sheet.



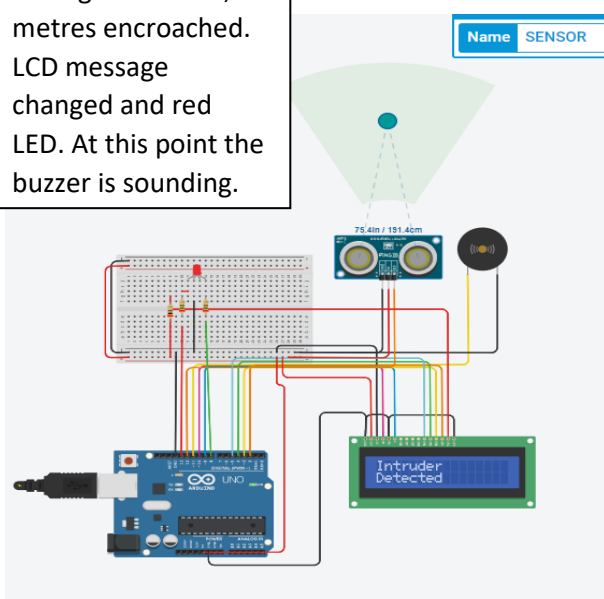
During simulation, 2 metres clear.

Green LED



During simulation, 2 metres encroached.

LCD message changed and red LED. At this point the buzzer is sounding.

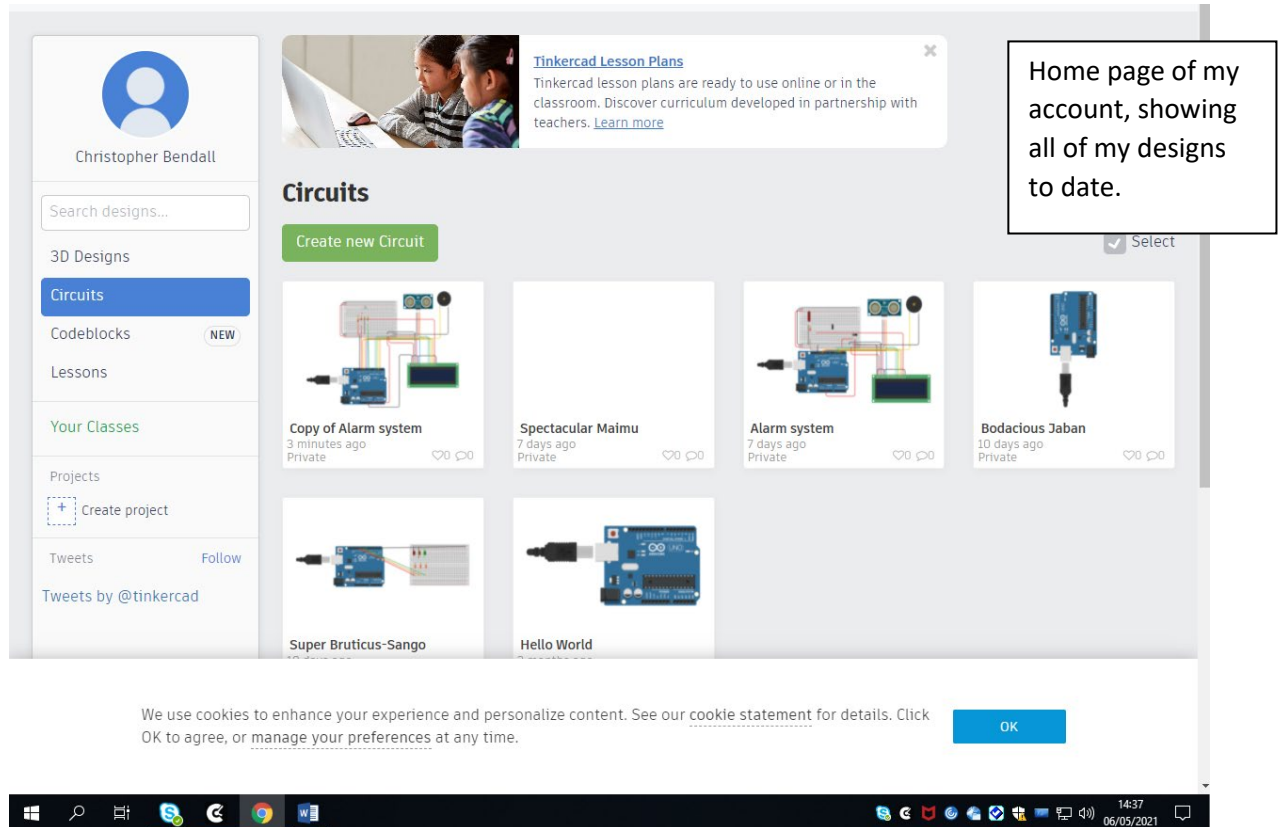


The code between lines 1 and 77

```
1 #include <LiquidCrystal.h>
2
3 const int COLUMNS= 16;
4 const int ROWS= 2;
5
6 LiquidCrystal lcd(10, 9, 5, 4, 3, 2);
7
8 #define SENSOR_PIN 12
9 #define NOTE_C4 262
10 #define NOTE_A3 220
11 const int BUZZER=11;
12
13
14 int LED=13;
15 int LEDGRN=8;
16
17 void setup()
18 {
19   Serial.begin(9600);
20   pinMode(BUZZER,OUTPUT); //designate buzzer as output
21   pinMode(LED,OUTPUT); //designate LED as output
22
23   lcd.begin(COLUMNS, ROWS); //INITIAL LCD MESSAGE
24   lcd.print("Alarm system");
25   lcd.setCursor(0, 1);
26   lcd.print("Energized");
27   delay (3000);
28   lcd.clear();
29 }
30
31
32 void loop()
33 {
34
35   int distance,microseconds;
36
37   pinMode(SENSOR_PIN,OUTPUT); //OUTPUT PING
38   digitalWrite(SENSOR_PIN,LOW); //CLEAR SIGNALS
39   delayMicroseconds(1000);
40   digitalWrite(SENSOR_PIN,HIGH); //SEND PING
41   delayMicroseconds(2000); //WAIT
42   digitalWrite(SENSOR_PIN,LOW); //CLEAR SIGNALS
43   pinMode(SENSOR_PIN,INPUT); //NOW WAITING FOR INPUT, PING BACK
44   microseconds=pulseIn(SENSOR_PIN,HIGH); //READING RECIEVED
45   distance=microseconds/29/2; //TIME INTO DISTANCE
46
```

```
44   microseconds=pulseIn(SENSOR_PIN,HIGH); //READING RECIEVED
45   distance=microseconds/29/2; //TIME INTO DISTANCE
46
47   Serial.println(distance); //SEE DISTANCE IN PRINT
48
49   if (distance <= 200 && distance >1) //IF PARAMETER IS BREACHED
50   {
51     digitalWrite(LED, HIGH); //RED LIGHT COME ON
52     digitalWrite(LEDGRN, LOW); //GREEN LIGHT OFF
53     lcd.clear ();
54
55     lcd.print("Intruder");
56     lcd.setCursor(0, 1);
57     lcd.print("Detected");
58
59     tone (BUZZER, NOTE_C4); //BUZZER SOUNDS
60     delay (1000);
61     tone (BUZZER, NOTE_A3);
62     delay(1000);
63   }
64   else
65   {
66     digitalWrite(LED, LOW); //RED LIGHT COME OFF
67     digitalWrite(LEDGRN, HIGH); //GREEN LIGHT ON
68     lcd.clear ();
69
70     lcd.print("Two Metre Area");
71     lcd.setCursor(0, 1);
72     lcd.print("All Clear");
73
74     noTone (BUZZER); //BUZZER NO SOUND
75     delay (1000); //ADDING THESE DELAYS SEEMED TO..
76     noTone (BUZZER);
77     delay(1000); //...HELP LCD STOP BLINKING
78
79
```

With the code, we first see I have included from the library the LCD function. Next I label pins and name them as definitions, 'inst' or 'const int'. Next is set up, the code that runs just one time on initial energising. Then we get to the loop which runs constantly. The loop includes the code sent to the scanner to continuously run and our 'IF' and 'ELSE' statements.



Task 5 – Discussion, Evaluation and Conclusions

Discuss your results and comment on the power and usefulness of the C programming language used in embedded systems. What modifications could be made to make this system more useful? Provide a conclusion to your report stating whether you have met the requirements of the system.

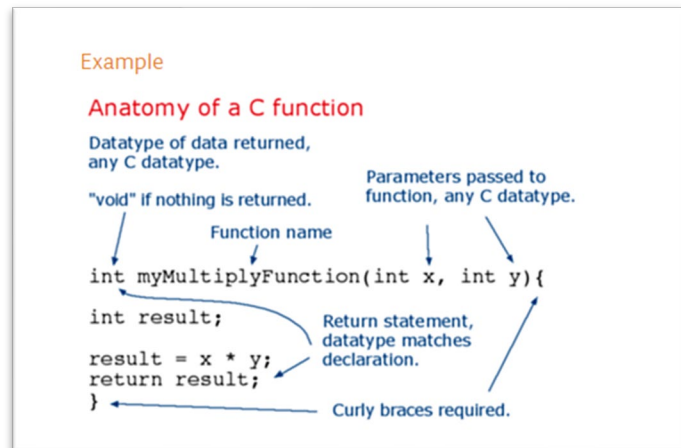
Overall, I am happy with the functionality of the system and it clearly does as intended and outlined in the scope for the project. This is a very simple design and the application I chose to model it on reflected that. If this was for example a house alarm, we would need more stored information, counters etc. At this point the code would become more complicated and need additions such as a keypad for extra security. I feel the C programming worked very well for what we have here and I must say I am happy with how it turned out. The testing process was pretty much continuous as you don't know the code is going to work until you have written it and simulated. Then if something is wrong in the code, it was often unclear and I would have to go through every line, often missing the mistakes over and over. I feel this is something that would decrease with more time and experience using it, however.

If I was to start again I would make various modifications. I would ensure everything came through some kind of switch which could be used as an on/off. If I had included this straight away it could have fitted in well. Going back to place one at the end of the project proved complicated in terms of both wiring and coding. So this is definitely something I would incorporate if I were to start again. Another thing I would utilise is the use of functions. Segmenting code into functions allows a programmer to create modular pieces of code that

perform a defined task and then return to the area of code from which the function was "called". The typical case for creating a function is when one needs to perform the same action multiple times in a program.

I would like to also incorporate a keypad. Obviously for the application I modelled the design on it is not necessary but if the alarm system was to be used for something else, such as a building alarm, I feel I could build on the code and components I already have for a more complex, effective system. This would include entering a code to activate and a code to reset/deactivate. There may also be a call for a battery back up in the event of the primary supply failing.

Due to the alarm being portable by nature the supply would come from a battery cell rather than plugging in as shown in the simulation. Due to the supply being low, I don't foresee this being an issue.



List of Appendices

1. Arduino Uno 3 datasheet
2. RGB LED datasheet
3. Resistor datasheets
4. Piezo-buzzer datasheet
5. LCD 16x2 datasheet
6. Ultra-sonic distance sensor datasheet
7. Flowchart
8. Test results sheet

Bibliography

Store.arduino.cc. 2021. Arduino Uno Rev3 | Arduino Official Store. [online] Available at: <<https://store.arduino.cc/arduino-uno-rev3>> [Accessed 11 May 2021].

R3, A., 2021. Arduino Uno - R3 - DEV-11021 - SparkFun Electronics. [online] Sparkfun.com. Available at: <<https://www.sparkfun.com/products/11021>> [Accessed 11 May 2021].

Instructables.com. 2021. [online] Available at: <<https://www.instructables.com/How-to-use-a-Buzzer-Arduino-Tutorial/>> [Accessed 11 May 2021].

Uk.rs-online.com. 2021. Sounder, Buzzer & Microphone Components | RS Components. [online] Available at: <<https://uk.rs-online.com/web/c/passive-components/sounder-buzzer-microphone-components/piezo-buzzer>> [Accessed 11 May 2021].

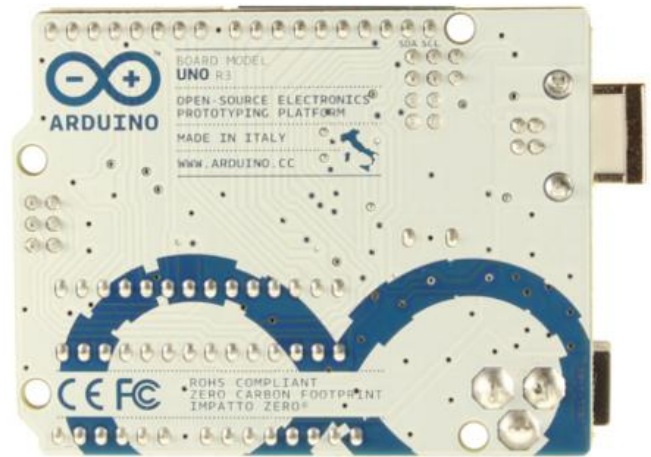
Merriam-webster.com. 2021. Definition of COMPONENT. [online] Available at: <<https://www.merriam-webster.com/dictionary/component>> [Accessed 11 May 2021].

MaxBotix Inc. 2021. Understanding How Ultrasonic Sensors Work | MaxBotix Inc.. [online] Available at: <<https://www.maxbotix.com/articles/how-ultrasonic-sensors-work.htm#:~:text=An%20ultrasonic%20sensor%20is%20an,information%20about%20an%20object's%20proximity.>> [Accessed 11 May 2021].

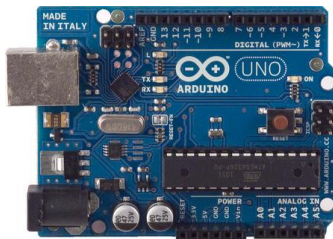
Arduino Uno



Arduino Uno R3 Front



Arduino Uno R3 Back



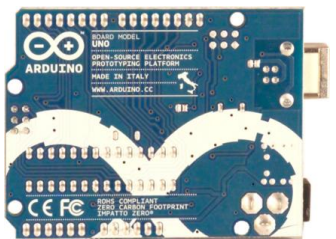
Arduino Uno R2 Front



Arduino Uno SMD



Arduino Uno Front



Arduino Uno Back

Overview

The Arduino Uno is a microcontroller board based on the ATmega328 ([datasheet](#)). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

| [Revision 2](#) of the Uno board has a resistor pulling the 8U2 HWB line to ground, making it easier to put into [DFU mode](#).

| [Revision 3](#) of the board has the following new features:

- 1.0 pinout: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible both with the board that use the AVR, which operate with 5V and with the Arduino Due that operate with 3.3V. The second one is a not connected pin, that is reserved for future purposes.
- Stronger RESET circuit.
- Atmega 16U2 replace the 8U2.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the [index of Arduino boards](#).

Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V

Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

Schematic & Reference Design

EAGLE files: [arduino-uno-Rev3-reference-design.zip](#) (NOTE: works with Eagle 6.0 and newer)

Schematic: [arduino-uno-Rev3-schematic.pdf](#)

Note: The Arduino reference design can use an Atmega8, 168, or 328, Current models use an ATmega328, but an Atmega8 is shown in the schematic for reference. The pin configuration is identical on all three processors.

Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

Memory

The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.

- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication using the [SPI library](#).
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the [analogReference\(\)](#) function. Additionally, some pins have specialized functionality:

- **TWI: A4 or SDA pin and A5 or SCL pin.** Support TWI communication using the [Wire library](#).

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with [analogReference\(\)](#).
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the [mapping between Arduino pins and ATmega328 ports](#). The mapping for the Atmega8, 168, and 328 is identical.

Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, [on Windows, a .inf file is required](#). The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Uno's digital pins.

The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the [documentation](#) for details. For SPI communication, use the [SPI library](#).

Programming

The Arduino Uno can be programmed with the Arduino software ([download](#)). Select "Arduino Uno from the **Tools > Board** menu (according to the microcontroller on your board). For details, see the [reference](#) and [tutorials](#).

The ATmega328 on the Arduino Uno comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available . The ATmega16U2/8U2 is loaded with a DFU bootloader, which can be activated by:

- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

You can then use [Atmel's FLIP software](#) (Windows) or the [DFU programmer](#) (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader). See [this user-contributed tutorial](#) for more information.

Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2/16U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload. This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see [this forum thread](#) for details.

USB Overcurrent Protection

The Arduino Uno has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Four screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.



ATTENTION

**OBSERVE PRECAUTIONS
FOR HANDLING
ELECTROSTATIC
DISCHARGE
SENSITIVE
DEVICES**

Part Number: L-154A4SURKQBDZGC

Hyper Red
Blue
Green

Features

- Uniform light output.
- Low power consumption.
- Long life-solid state reliability.
- RoHS compliant.

Description

The Hyper Red source color devices are made with Al-GaN/P on GaAs substrate Light Emitting Diode.

The Blue source color devices are made with InGaN Light Emitting Diode.

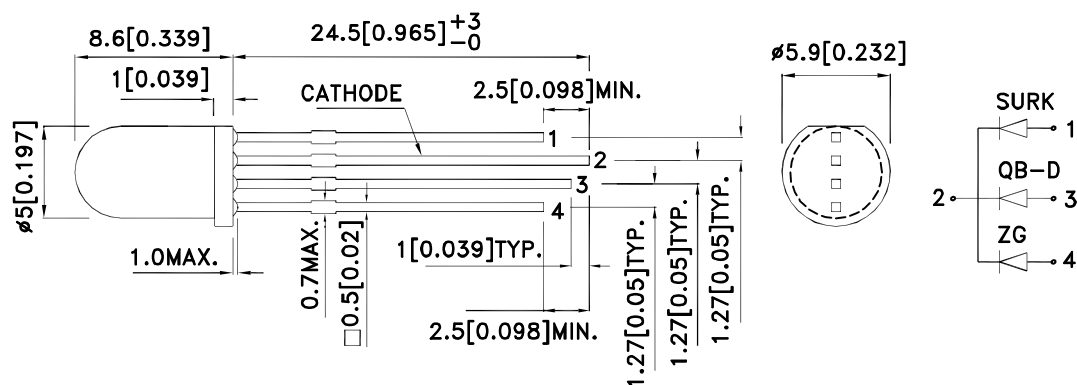
The Green source color devices are made with InGaN on Sapphire Light Emitting Diode.

Static electricity and surge damage the LEDS.

It is recommended to use a wrist band or anti-electrostatic glove when handling the LEDs.

All devices, equipment and machinery must be electrically grounded.

Package Dimensions



Notes:

1. All dimensions are in millimeters (inches).
2. Tolerance is $\pm 0.25(0.01")$ unless otherwise noted.
3. Lead spacing is measured where the leads emerge from the package.
4. The specifications, characteristics and technical data described in the datasheet are subject to change without prior notice.



Selection Guide

Part No.	Dice	Lens Type	Iv (mcd) [2] @ 20mA		Viewing Angle [1]
			Min.	Typ.	2θ1/2
L-154A4SURKQBDZGC	Hyper Red (AlGaInP)	Water Clear	600	1200	50°
	Blue (InGaN)		450	800	
	Green (InGaN)		1000	1700	

Notes:

1. θ1/2 is the angle from optical centerline where the luminous intensity is 1/2 of the optical peak value.
2. Luminous intensity/ luminous Flux: +/-15%.

Electrical / Optical Characteristics at TA=25°C

Symbol	Parameter	Device	Typ.	Max.	Units	Test Conditions
λpeak	Peak Wavelength	Hyper Red Blue Green	650 468 515		nm	IF=20mA
λD [1]	Dominant Wavelength	Hyper Red Blue Green	630 470 525		nm	IF=20mA
Δλ1/2	Spectral Line Half-width	Hyper Red Blue Green	28 25 30		nm	IF=20mA
C	Capacitance	Hyper Red Blue Green	35 100 45		pF	VF=0V;f=1MHz
VF [2]	Forward Voltage	Hyper Red Blue Green	1.95 3.3 3.3	2.5 4 4.1	V	IF=20mA
IR	Reverse Current	Hyper Red Blue Green		10 50 50	uA	VR=5V

Notes:

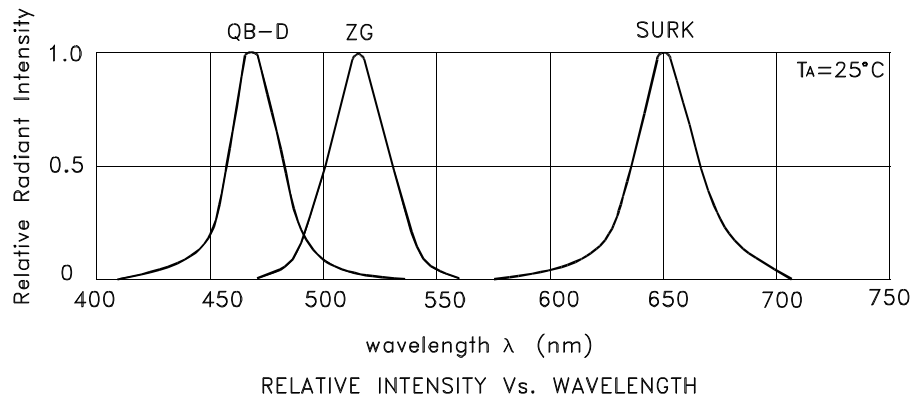
- 1.Wavelength: +/-1nm.
2. Forward Voltage: +/-0.1V.

Absolute Maximum Ratings at TA=25°C

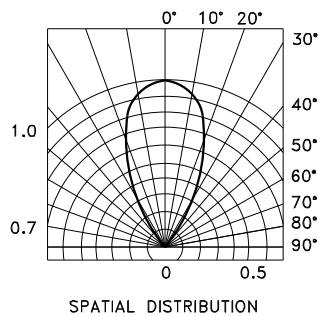
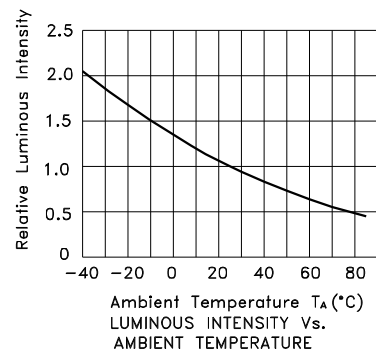
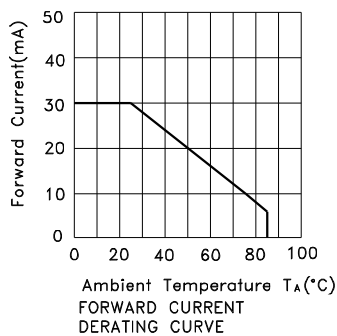
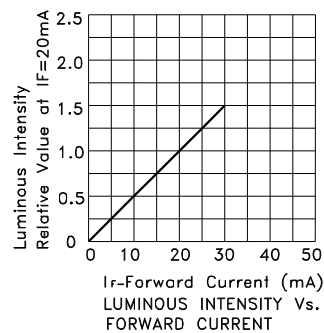
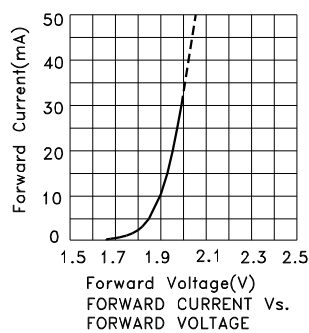
Parameter	Hyper Red	Blue	Green	Units
Power dissipation	75	120	102.5	mW
DC Forward Current	30	30	25	mA
Peak Forward Current [1]	185	150	150	mA
Reverse Voltage	5			V
Operating/Storage Temperature	-40°C To +85°C			
Lead Solder Temperature [2]	260°C For 3 Seconds			
Lead Solder Temperature [3]	260°C For 5 Seconds			

Notes:

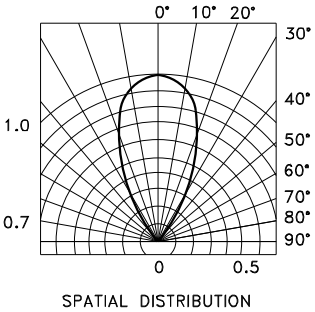
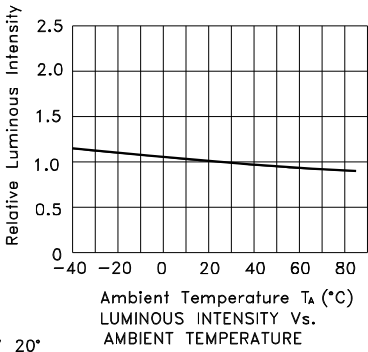
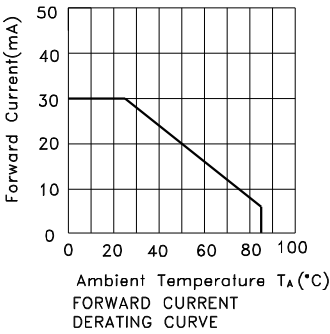
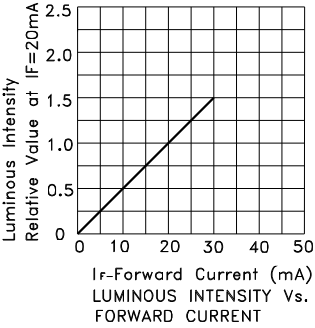
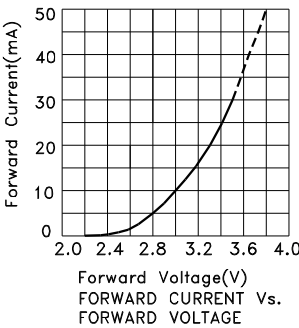
1. 1/10 Duty Cycle, 0.1ms Pulse Width.
2. 2mm below package base.
3. 5mm below package base.



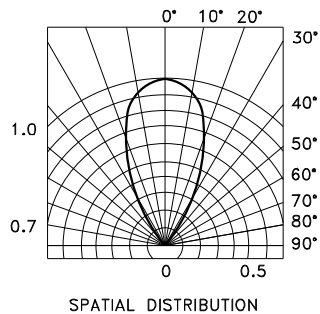
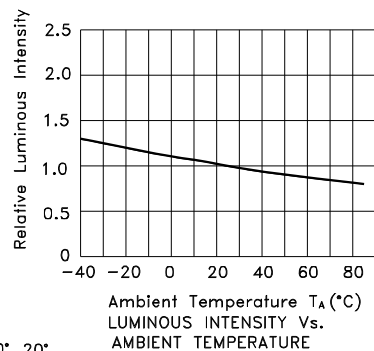
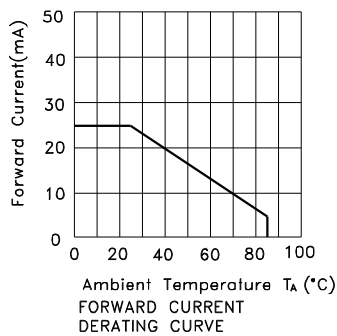
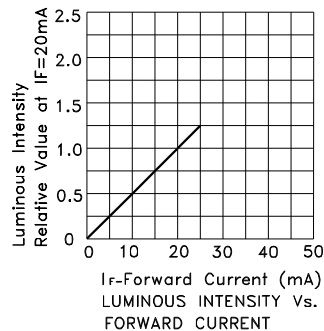
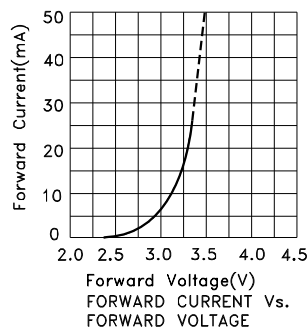
L-154A4SURKQBDZGC
Hyper Red



Blue

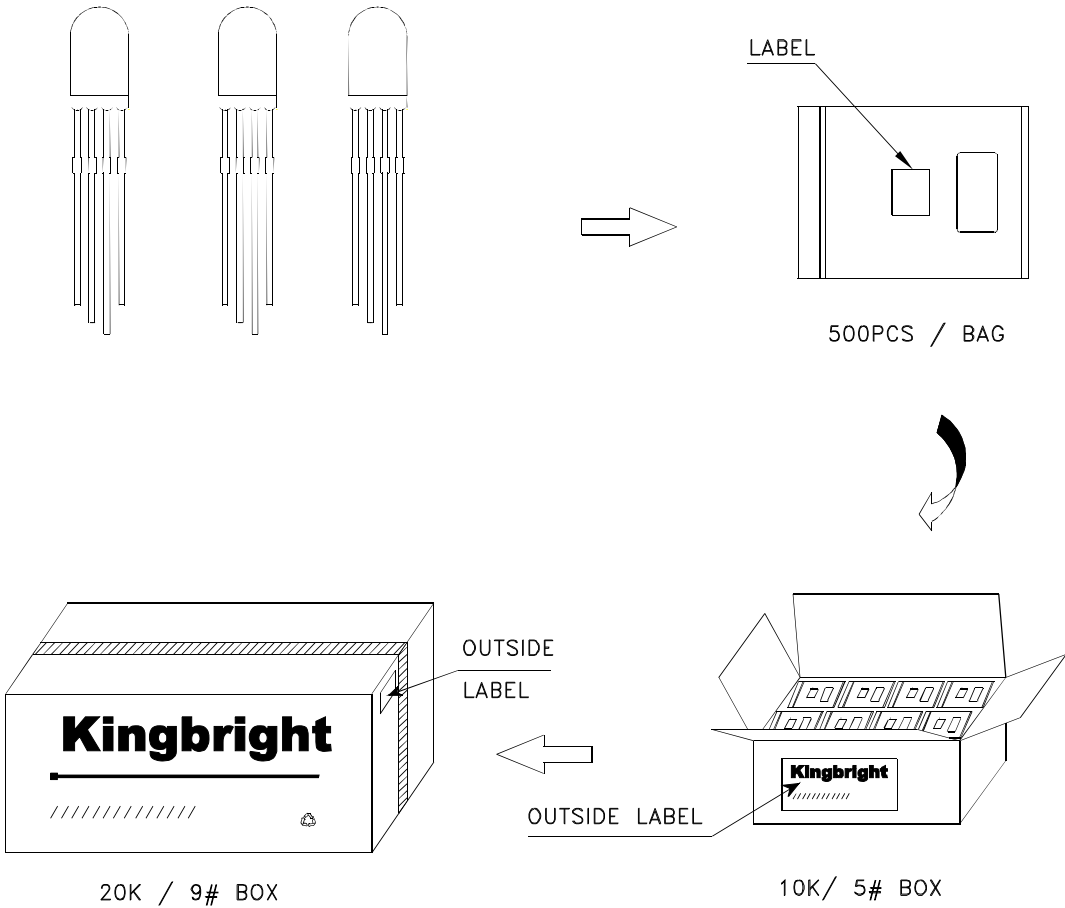


Green



PACKING & LABEL SPECIFICATIONS

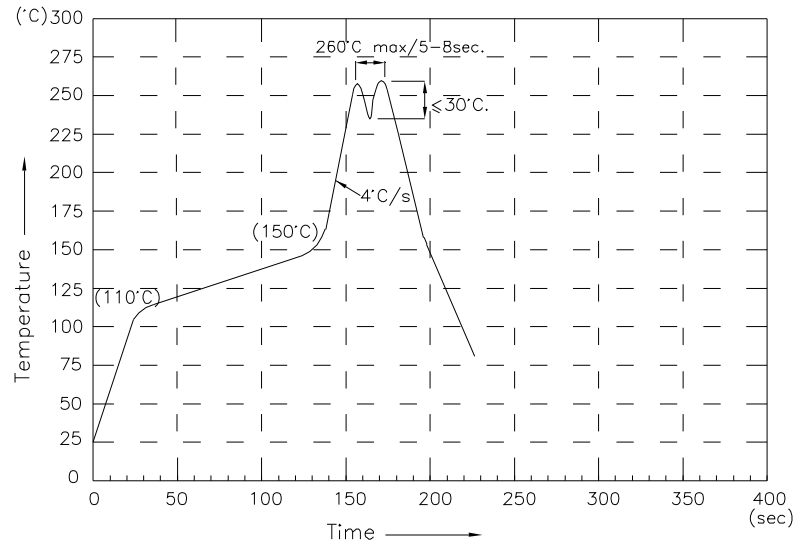
L-154A4SURKQBDZGC



Kingbright	
P/NO: L-154A4xxx	
QTY: 500 pcs	Q.C. Q C xx xx xxxx PASSED
S/N: XXXX	
CODE: XXX	
LOT NO:	
 xxxxxxxxxxxxxxxxxxxxxx	
RoHS Compliant	

L-154A4SURKQBDZGC

Wave Soldering Profile For Lead-free Through-hole LED.



NOTES:

1. Recommend the wave temperature 245°C~260°C. The maximum soldering temperature should be less than 260°C.
2. Do not apply stress on epoxy resins when temperature is over 85 degree°C.
3. The soldering profile apply to the lead free soldering (Sn/Cu/Ag alloy).
4. No more than once.

LED MOUNTING METHOD

1. The lead pitch of the LED must match the pitch of the mounting holes on the PCB during component placement. Lead-forming may be required to insure the lead pitch matches the hole pitch. Refer to the figure below for proper lead forming procedures.

(Fig. 1)

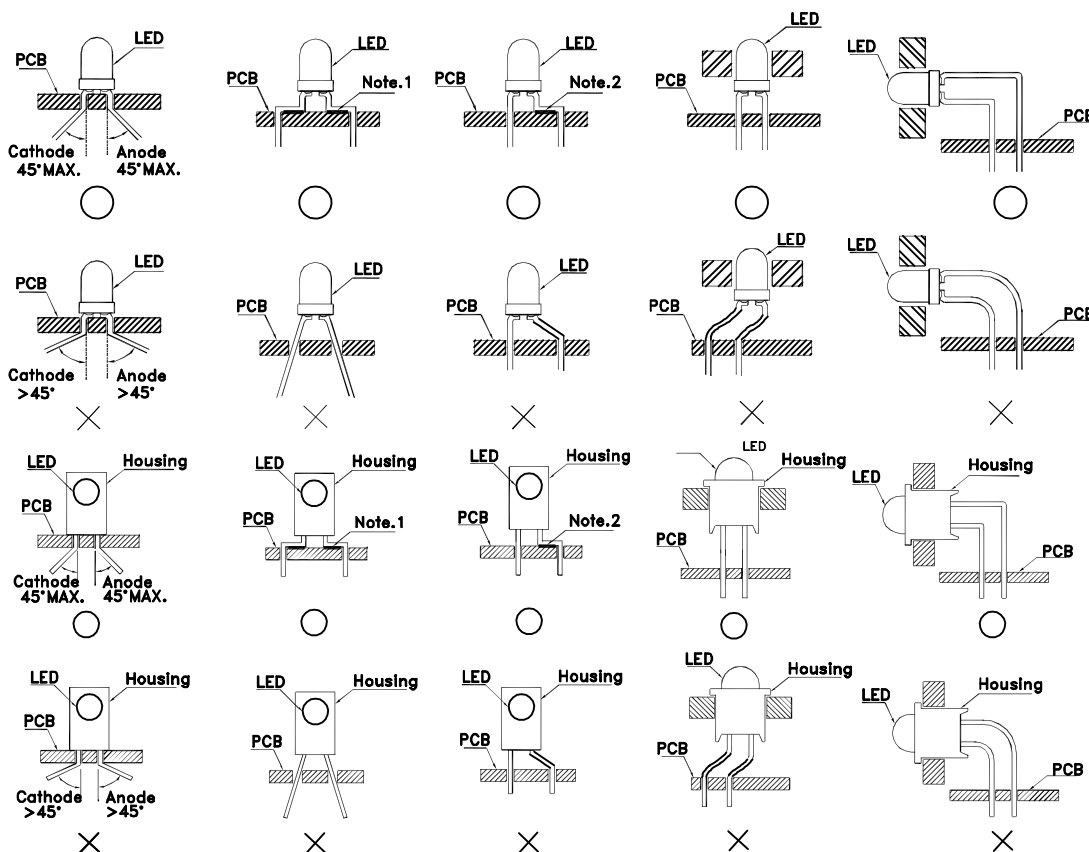


Fig.1

“○” Correct mounting method “×” Incorrect mounting method
 Note 1-2 : Do not route PCB trace in the contact area between the leadframe and the PCB to prevent short-circuits.

2. When soldering wire to the LED, use individual heat-shrink tubing to insulate the exposed leads to prevent accidental contact short-circuit.
 (Fig. 2)

3. Use stand-offs (Fig. 3) or spacers (Fig. 4) to securely position the LED above the PCB.

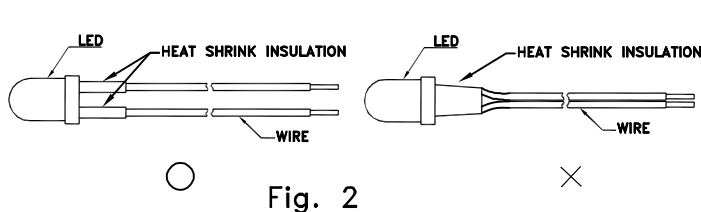


Fig. 2

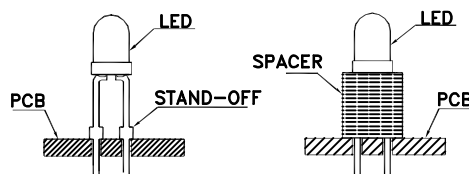


Fig. 3

Fig. 4

LEAD FORMING PROCEDURES

1. Maintain a minimum of 2mm clearance between the base of the LED lens and the first lead bend. (Fig. 5 and 6)

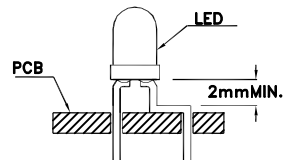


Fig. 5

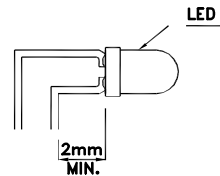


Fig. 6

2. Lead forming or bending must be performed before soldering, never during or after Soldering.
3. Do not stress the LED lens during lead-forming in order to fractures in the lens epoxy and damage the internal structures.
4. During lead forming, use tools or jigs to hold the leads securely so that the bending force will not be transmitted to the LED lens and its internal structures. Do not perform lead forming once the component has been mounted onto the PCB. (Fig. 7)
5. Do not bend the leads more than twice. (Fig. 8)
6. After soldering or other high-temperature assembly, allow the LED to cool down to 50°C before applying outside force (Fig. 9). In general, avoid placing excess force on the LED to avoid damage. For any questions please consult with Kingbright representative for proper handling procedures.

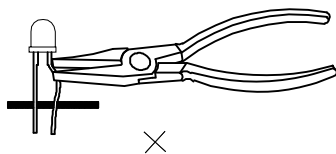


Fig. 7

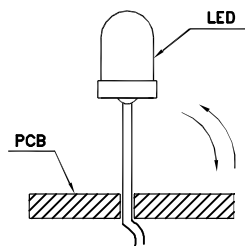


Fig. 8

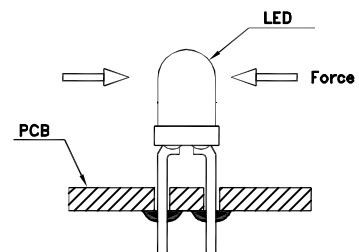
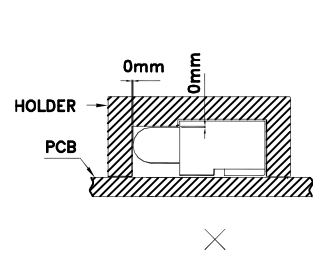
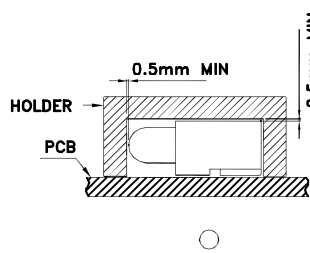
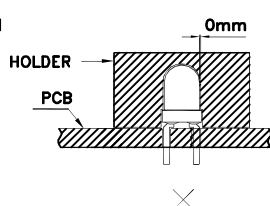
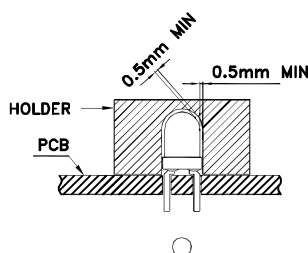
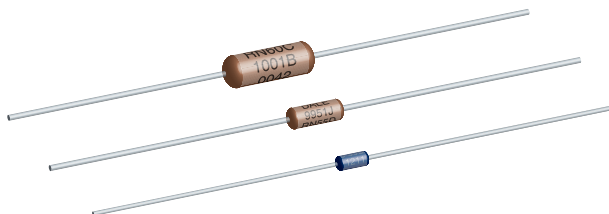


Fig. 9

7. No stress shall be applied on the LED during soldering to prevent damage.



Metal Film Resistors, Military, MIL-R-10509 Qualified, Precision, Type RN and MIL-PRF-22684 Qualified, Type RL



FEATURES

- Very low noise (- 40 dB)
- Very low voltage coefficient (5 ppm/V)
- Controlled temperature coefficient
- Flame retardant epoxy coating
- Commercial alternatives to military styles are available with higher power ratings. See CMF Industrial data sheet: www.vishay.com/doc?31018

STANDARD ELECTRICAL SPECIFICATIONS

GLOBAL MODEL	MIL STYLE	MIL SPEC. SHEET	POWER RATING $P_{70^{\circ}\text{C}}$ W	POWER RATING $P_{125^{\circ}\text{C}}$ W	MAX. WORKING VOLTAGE ⁽¹⁾ V	RESISTANCE RANGE Ω MIL-R-10509 $\pm 100 \text{ ppm}/^{\circ}\text{C}$ (D)	RESISTANCE RANGE Ω MIL-R-10509 $\pm 50 \text{ ppm}/^{\circ}\text{C}$ (C)	RESISTANCE RANGE Ω MIL-R-10509 $\pm 25 \text{ ppm}/^{\circ}\text{C}$ (E)	RESISTANCE RANGE Ω MIL-PRF-22684	TOL. ⁽³⁾ $\pm \%$	DIELECTRIC STRENGTH V_{AC}
CMF50	RN50	08	-	0.05	200	-	10 to 100K	10 to 100K	-	0.1, 0.25, 0.5, 1	450
CMF55	RN55	07	0.125	0.10	200	10 to 301K	49.9 to 100K	49.9 to 100K	-	0.1, 0.25, 0.5, 1	450
CMF60	RN60	01	0.25	0.125	300	10 to 1M	49.9 to 499K	49.9 to 499K	-	0.1, 0.25, 0.5, 1	500
CMF65	RN65	02	0.50	0.25	350	10 to 2M	49.9 to 1M	49.9 to 1M	-	0.1, 0.25, 0.5, 1	900
CMF70	RN70	03	0.75 ⁽²⁾	0.50	500	10 to 2.49M	24.9 to 1M	24.9 to 1M	-	0.1, 0.25, 0.5, 1	900
CMF07	RL07	01	0.25	-	250	-	-	-	51 to 150K	2, 5	450
CMF20	RL20	02	0.50	-	350	-	-	-	4.3 to 470K	2, 5	700

Notes

- (1) Continuous working voltage shall be $\sqrt{P \times R}$ or maximum working voltage, whichever is less.
(2) Formerly rated at 1 W and is the direct replacement for RN70 of MIL-R-10509 rev. D.
(3) Tolerances of $\pm 0.1 \%$, $\pm 0.25 \%$ and $\pm 0.5 \%$ are not applicable to characteristic D.

TECHNICAL SPECIFICATIONS

PARAMETER	UNIT	CONDITION
Voltage Coefficient	ppm/V	5 when measured between 10 % and full rated voltage
Insulation Resistance	Ω	$\geq 10^{10}$ min. dry; $\geq 10^8$ min. after moisture test
Operating Temperature Range	$^{\circ}\text{C}$	- 65/+ 175 (see derating curves for military range)
Terminal Strength	lb	5 pound pull test for RL07/RL20; 2 pound pull test for all others
Solderability		Continuous satisfactory coverage when tested in accordance with MIL-R-10509 and MIL-PRF-22684



GLOBAL PART NUMBER INFORMATION

New Global Part Numbering: RN60D3483FR36 (preferred part numbering format)

R	N	6	0	D	3	4	8	3	F	R	3	6			
MIL STYLE	CHARACTERISTIC		RESISTANCE VALUE			TOLERANCE CODE			PACKAGING			SPECIAL			
RN50 RN55 RN60 RN65 RN70	E = 25 ppm C = 50 ppm D = 100 ppm		3 digit significant figure, followed by a multiplier Use "R" for values < 100 Ω 10R0 = 10 Ω 2152 = 21.5 kΩ 2494 = 2.49 MΩ			B = ± 0.1 % C = ± 0.25 % D = ± 0.5 % F = ± 1 %			B14 = Tin/lead, bulk BSL = Tin/lead, bulk, single lot date code R36 = Tin/lead, T/R (full) RE6 = Tin/lead, T/R (1000 pieces) RSL = Tin/lead, T/R, single lot date code			Blank = Standard (Dash number) 88 = Hot solder dip 143 = Non-magnetic			

Historical Part Number example: RN60D3483F (will continue to be accepted)

RN60	D	3483	F	R36
MIL STYLE	CHARACTERISTIC	RESISTANCE VALUE	TOLERANCE CODE	PACKAGING

New Global Part Numbering: RL07S471JR36 (preferred part numbering format)

R	L	0	7	S	4	7	1	J	R	3	6			
MIL STYLE	LEAD MATERIAL		RESISTANCE VALUE			TOLERANCE CODE			PACKAGING			SPECIAL		
RL07 RL20	S = Solderable		2 digit significant figure, followed by a multiplier Use "R" for values < 10 Ω 4R3 = 4.3 Ω 202 = 2.0 k Ω 474 = 470 k Ω			G = $\pm 2\%$ J = $\pm 5\%$			B14 = Tin/lead, bulk BSL = Tin/lead, bulk, single lot date code R36 = Tin/lead, T/R (full) RE6 = Tin/lead, T/R (1000 pieces) RSL = Tin/lead, T/R, single lot date code			Blank = Standard (Dash number) 88 = Hot solder dip 143 = Non-magnetic		

Historical Part Number example: RL07S471J (will continue to be accepted)

RL07	S	471	J	R36
MIL STYLE	LEAD MATERIAL	RESISTANCE VALUE	TOLERANCE CODE	PACKAGING

Note

- For additional information on packaging, refer to the Through Hole Resistor Packaging document (www.vishay.com/doc?31544).

MATERIAL SPECIFICATIONS

Element	Nickel-chrome alloy
Coating	Flame retardant epoxy, formulated for superior moisture protection
Core	Fire-cleaned high purity ceramic
Termination	Standard lead material is solder-coated copper. Solderable and weldable.

APPLICABLE MIL-SPECS

MIL-R-10509 and MIL-PRF-22684: The CMF models meet or exceed the electrical, environmental and dimensional requirements of MIL-R-10509 and MIL-PRF-22684.

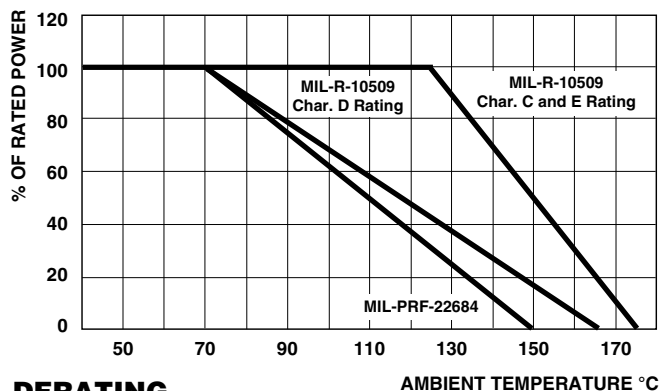
Noise: Vishay Dale metal film resistors have exceptionally low noise level. Average for standard resistance range is 0.10 μV per V over a decade of frequency, with low and intermediate resistance values typically below 0.05 μV per V.

CAGE CODE: 91637

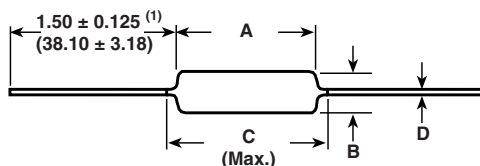
ENVIRONMENTAL SPECIFICATIONS

General	Environmental performance is shown in the Environmental Performance table. Test methods are those specified in MIL-R-10509 and MIL-PRF-22684.
Shelf Life	Resistance shifts due to storage at room temperature are negligible.

Vishay Dale CMF resistors have an operating temperature range of - 65 °C to + 175 °C. They must be derated according to the following curves:


DERATING

DIMENSIONS in inches (millimeters)



VISHAY DALE MODEL	A	B	C (MAX.)	D
CMF50	0.150 ± 0.020 (3.81 ± 0.51)	0.065 ± 0.015 (1.65 ± 0.38)	0.244 (6.20)	0.016 ± 0.002 (0.41 ± 0.05)
CMF55	0.240 ± 0.020 (6.10 ± 0.51)	0.090 ± 0.008 (2.29 ± 0.20)	0.290 (7.37)	0.025 ± 0.002 (0.64 ± 0.05)
CMF60	0.344 ± 0.031 (8.74 ± 0.79)	0.145 ± 0.015 (3.68 ± 0.38)	0.425 (10.80)	0.025 ± 0.002 (0.64 ± 0.05)
CMF65	0.562 ± 0.031 (14.27 ± 0.79)	0.180 ± 0.015 (4.57 ± 0.38)	0.687 (17.45)	0.025 ± 0.002 (0.64 ± 0.05)
CMF70	0.562 ± 0.031 (14.27 ± 0.79)	0.180 ± 0.015 (4.57 ± 0.38)	0.687 (17.45)	0.032 ± 0.002 (0.81 ± 0.05)
CMF07	0.240 ± 0.020 (6.10 ± 0.51)	0.090 ± 0.008 (2.29 ± 0.20)	0.290 (7.37)	0.025 ± 0.002 (0.64 ± 0.05)
CMF20	0.375 ± 0.040 (9.53 ± 1.02)	0.145 ± 0.015 (3.68 ± 0.38)	0.425 (10.80)	0.032 ± 0.002 (0.81 ± 0.05)

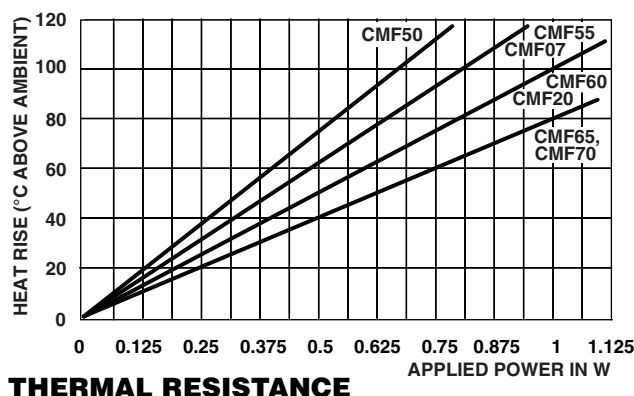
Notes

- (1) Lead length for product in bulk pack. For product supplied in Tape and Reel, the actual lead length would be based on the body size, tape spacing and lead trim.

MILITARY POWER RATING			
WATTAGE	MILITARY QUALIFIED		
	MIL-R-10509		MIL-PRF-22684
	AT + 70 °C (D)	AT + 125 °C (C and E)	AT + 70 °C
0.05	-	RN50	-
0.10	-	RN55	-
0.125	RN55	RN60	-
0.25	RN60	RN65	RL07
0.50	RN65	RN70	RL20
0.75 (2)	RN70	-	-

Notes

- Commercial equivalents of military styles are available with higher power ratings. Consult factory.
- (2) Formerly rated at 1 W and is the direct replacement for RN70 of MIL-R-10509 rev. D.

**MARKING** (per MIL-PRF-10509)

Characteristics: D = 100 ppm, C = 50 ppm, E = 25 ppm
 Tolerance: F = 1 %, D = 0.5 %, C = 0.25 %, B = 0.1 %
 Value = Three significant figures and multiplier
 J = JAN (Joint Army - Navy) brand

RN50: (3 lines)

J50D JAN, type, characteristic
 1211 Value
 F137 Tolerance and 3 digit date code

RN55, RN60, RN65, RN70 (4 lines)

DALE Company logo
 0137J 4 digit date code and JAN brand
 RN55D Type and characteristic
 1211F Value and Tolerance

Note

- RL series are color banded per MIL-PRF-22684.

PERFORMANCE

REQUIREMENT	MIL-R-10509			MIL-PRF-22684
	CHARACTERISTIC D	CHARACTERISTIC C	CHARACTERISTIC E	
MIL Temperature Coefficient	+ 200 ppm/°C - 500 ppm/°C	± 50 ppm/°C	± 25 ppm/°C	± 200 ppm/°C
Applicable Vishay Dale Temperature Coefficient	± 100 ppm/°C	± 50 ppm/°C	± 25 ppm/°C	± 200 ppm/°C
TEST	MIL_{max.}	MIL_{max.}	MIL_{max.}	MIL_{max.}
Thermal Shock	± 0.50 % ΔR	± 0.25 % ΔR	± 0.25 % ΔR	± 1.00 % ΔR
Short Time Overload	± 0.50 % ΔR	± 0.25 % ΔR	± 0.25 % ΔR	± 0.50 % ΔR
Low Temperature Operation	± 0.50 % ΔR	± 0.25 % ΔR	± 0.25 % ΔR	± 0.50 % ΔR
Moisture Resistance	± 1.50 % ΔR	± 0.50 % ΔR	± 0.50 % ΔR	± 1.50 % ΔR
Shock	± 0.50 % ΔR	± 0.25 % ΔR	± 0.25 % ΔR	± 0.50 % ΔR
Vibration	± 0.50 % ΔR	± 0.25 % ΔR	± 0.25 % ΔR	± 0.50 % ΔR
Load Life	± 1.00 % ΔR	± 0.50 % ΔR	± 0.50 % ΔR	± 2.00 % ΔR
Dielectric Withstanding Voltage	± 0.50 % ΔR	± 0.25 % ΔR	± 0.25 % ΔR	± 0.50 % ΔR
Effect of Solder	± 0.50 % ΔR	± 0.10 % ΔR	± 0.10 % ΔR	± 0.50 % ΔR



Disclaimer

ALL PRODUCT, PRODUCT SPECIFICATIONS AND DATA ARE SUBJECT TO CHANGE WITHOUT NOTICE TO IMPROVE RELIABILITY, FUNCTION OR DESIGN OR OTHERWISE.

Vishay Intertechnology, Inc., its affiliates, agents, and employees, and all persons acting on its or their behalf (collectively, "Vishay"), disclaim any and all liability for any errors, inaccuracies or incompleteness contained in any datasheet or in any other disclosure relating to any product.

Vishay makes no warranty, representation or guarantee regarding the suitability of the products for any particular purpose or the continuing production of any product. To the maximum extent permitted by applicable law, Vishay disclaims (i) any and all liability arising out of the application or use of any product, (ii) any and all liability, including without limitation special, consequential or incidental damages, and (iii) any and all implied warranties, including warranties of fitness for particular purpose, non-infringement and merchantability.

Statements regarding the suitability of products for certain types of applications are based on Vishay's knowledge of typical requirements that are often placed on Vishay products in generic applications. Such statements are not binding statements about the suitability of products for a particular application. It is the customer's responsibility to validate that a particular product with the properties described in the product specification is suitable for use in a particular application. Parameters provided in datasheets and/or specifications may vary in different applications and performance may vary over time. All operating parameters, including typical parameters, must be validated for each customer application by the customer's technical experts. Product specifications do not expand or otherwise modify Vishay's terms and conditions of purchase, including but not limited to the warranty expressed therein.

Except as expressly indicated in writing, Vishay products are not designed for use in medical, life-saving, or life-sustaining applications or for any other application in which the failure of the Vishay product could result in personal injury or death. Customers using or selling Vishay products not expressly indicated for use in such applications do so at their own risk. Please contact authorized Vishay personnel to obtain written terms and conditions regarding products designed for such applications.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document or by any conduct of Vishay. Product names and markings noted herein may be trademarks of their respective owners.

Material Category Policy

Vishay Intertechnology, Inc. hereby certifies that all its products that are identified as RoHS-Compliant fulfill the definitions and restrictions defined under Directive 2011/65/EU of The European Parliament and of the Council of June 8, 2011 on the restriction of the use of certain hazardous substances in electrical and electronic equipment (EEE) - recast, unless otherwise specified as non-compliant.

Please note that some Vishay documentation may still make reference to RoHS Directive 2002/95/EC. We confirm that all the products identified as being compliant to Directive 2002/95/EC conform to Directive 2011/65/EU.

Vishay Intertechnology, Inc. hereby certifies that all its products that are identified as Halogen-Free follow Halogen-Free requirements as per JEDEC JS709A standards. Please note that some Vishay documentation may still make reference to the IEC 61249-2-21 definition. We confirm that all the products identified as being compliant to IEC 61249-2-21 conform to JEDEC JS709A standards.



Features

- Black in colour
- With internal drive circuit
- Sealed structure
- Wave solderable and washable
- Housing material: Noryl

**RoHS
Compliant**

Applications

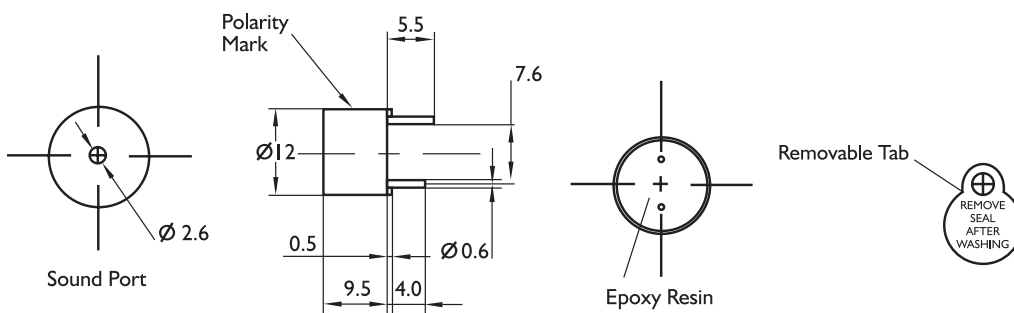
- Computer and peripherals
- Communications equipment
- Portable equipment
- Automobile electronics
- POS system
- Electronic cash register

Specifications:

Rated Voltage	: 6V DC
Operating Voltage	: 4 to 8V DC
Rated Current*	: ≤30mA
Sound Output at 10cm*	: ≥85dB
Resonant Frequency	: 2300 ±300Hz
Tone	: Continuous
Operating Temperature	: -25°C to +80°C
Storage Temperature	: -30°C to +85°C
Weight	: 2g

*Value applying at rated voltage (DC)

Diagram



Dimensions : Millimetres
Tolerance : ±0.5mm

Part Number Table

Description	Part Number
Buzzer, Electromech, 6V DC	ABI-009-RC

Important Notice : This data sheet and its contents (the "Information") belong to the members of the Premier Farnell group of companies (the "Group") or are licensed to it. No licence is granted for the use of it other than for information purposes in connection with the products to which it relates. No licence of any intellectual property rights is granted. The Information is subject to change without notice and replaces all data sheets previously supplied. The Information supplied is believed to be accurate but the Group assumes no responsibility for its accuracy or completeness, any error in or omission from it or for any use made of it. Users of this data sheet should check for themselves the Information and the suitability of the products for their purpose and not make any assumptions based on information included or omitted. Liability for loss or damage resulting from any reliance on the Information or use of it (including liability resulting from negligence or where the Group was aware of the possibility of such loss or damage arising) is excluded. This will not operate to limit or restrict the Group's liability for death or personal injury resulting from its negligence. pro-SIGNAL is the registered trademark of the Group. © Premier Farnell plc 2012.

XIAMEN AMOTEC DISPLAY CO.,LTD

SPECIFICATIONS OF LCD MODULE

MODULE NO : ADM1602K-NSW-FBS/3.3V

DOC.REVISION: 00

	SIGNATURE	DATE
PREPARED BY (RD ENGINEER)	QIU	2008-10-29
CHECKED BY	<i>Chen</i>	2008-10-29
APPROVED BY	<i>yfe</i>	2008-10-29

DOCUMENT REVISION HISTORY

VERSINO	DATE	DESCRIPTION	CHANGED BY
00	Oct-29-2008	First issue	

CONTENTS

Item	Page
Functions & Features	3
Mechanical specifications	3
Dimensional Outline	4
Absolute maximum ratings	5
Block diagram	5
Pin description	5
Contrast adjust	6
Optical characteristics	6
Electrical characteristics	6
Timing Characteristics	7-8
Instruction description	9-12
Display character address code:	12
character pattern	13
Quality Specifications	14--21

1. Features

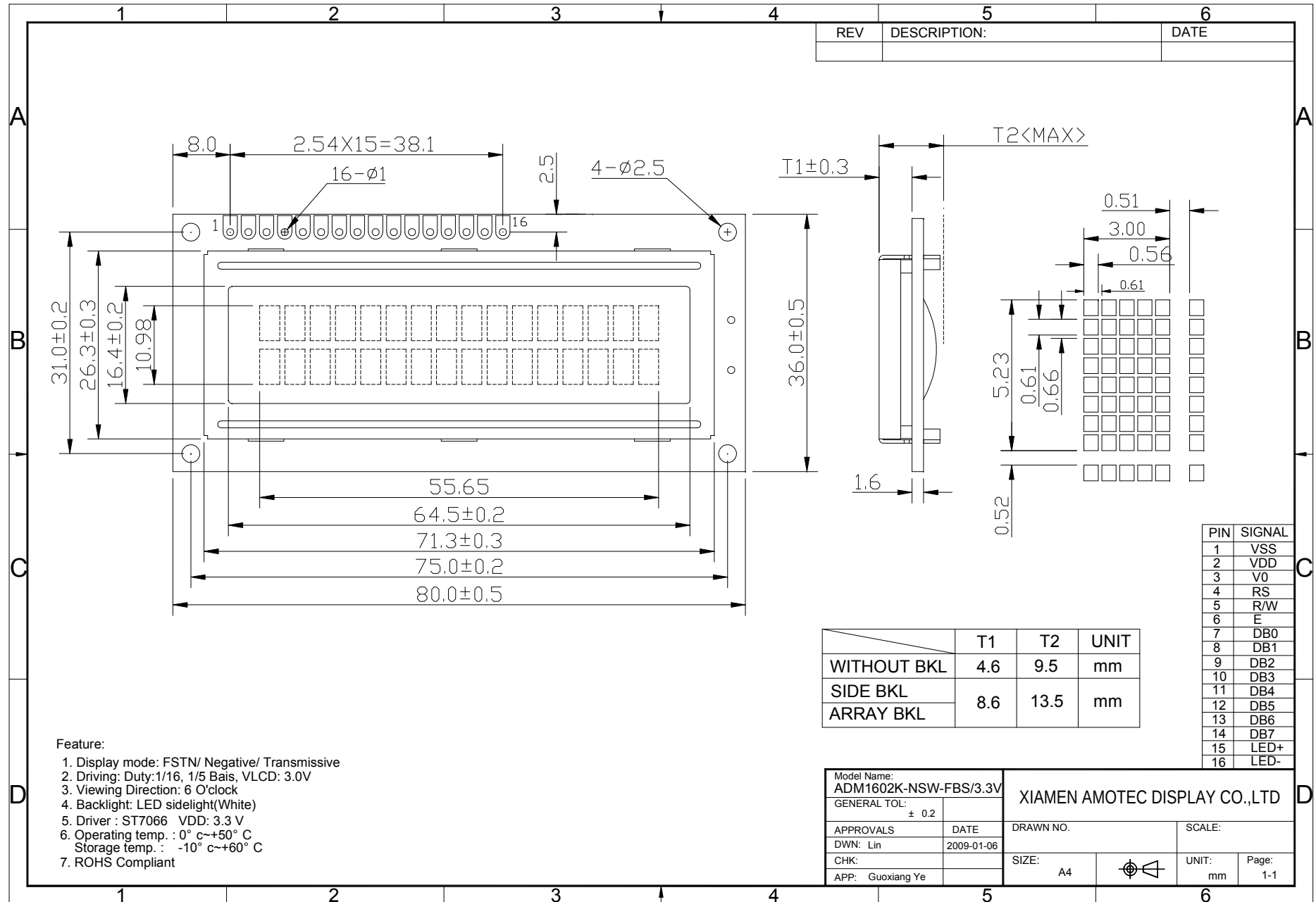
1. 5x8 dots with cursor
2. 16characters *2lines display
3. 4-bit or 8-bit MPU interfaces
4. Built-in controller (ST7066 or equivalent)
5. Display Mode & Backlight Variations
6. ROHS Compliant

LCD type	☐ TN			
	☐ FSTN	☒ FSTN Negative		
	☐ STN Yellow Green		☐ STN Gray	☐ STN Blue Negative
View direction	☒ 6 O'clock		☐ 12 O'clock	
Rear Polarizer	☐ Reflective		☐ Transflective	☒ Transmissive
Backlight Type	☒ LED	☐ EL	☐ Internal Power	☒ 3.3V Input
		☐ CCFL	☒ External Power	☐ 5.0V Input
Backlight Color	☒ White	☐ Blue	☐ Amber	☐ Yellow-Green
Temperature Range	☒ Normal		☐ Wide	☐ Super Wide
DC to DC circuit	☐ Build-in			☒ Not Build-in
Touch screen	☐ With			☒ Without
Font type	☒ English-Japanese		☐ English-Europen	☐ English-Russian
	☐ Other			

2. MECHANICAL SPECIFICATIONS

Module size	80.0mm(L)*36.0mm(W)* Max13.5(H)mm
Viewing area	64.5mm(L)*16.4mm(W)
Character size	3.00mm(L)*5.23mm(W)
Character pitch	3.51mm(L)*5.75mm(W)
Weight	Approx.

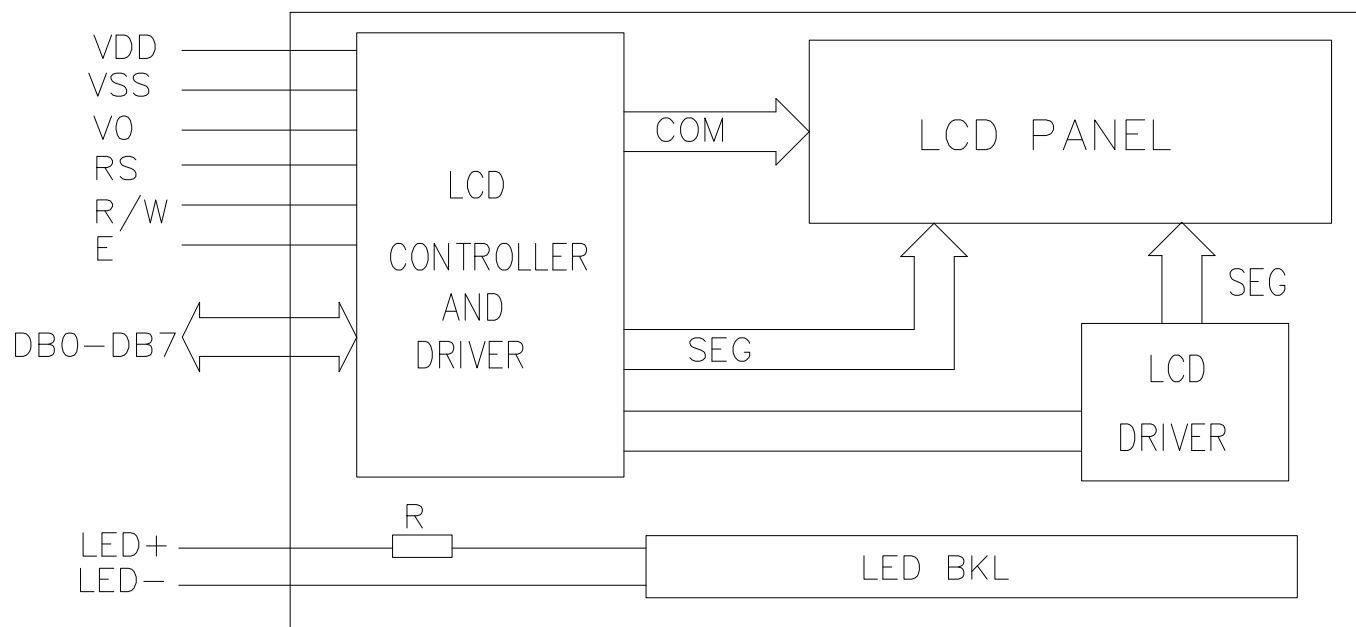
3. Outline dimension



4. Absolute maximum ratings

Item	Symbol	Standard			Unit
Power voltage	$V_{DD}-V_{SS}$	0	-	7.0	V
Input voltage	V_{IN}	VSS	-	VDD	
Operating temperature range	V_{OP}	0	-	+50	°C
Storage temperature range	V_{ST}	-10	-	+60	

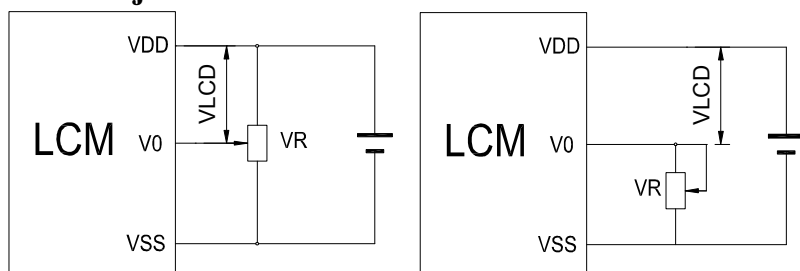
5. Block diagram



6. Interface pin description

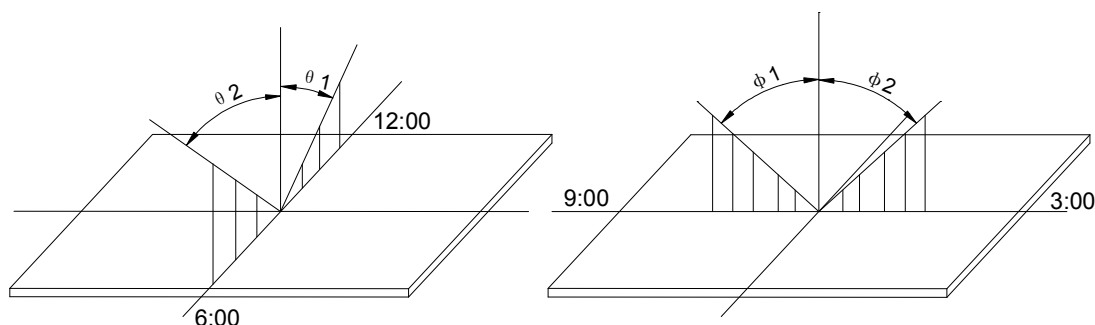
Pin no.	Symbol	External connection	Function
1	VSS	Power supply	Signal ground for LCM
2	V_{DD}		Power supply for logic for LCM
3	V_0		Contrast adjust
4	RS	MPU	Register select signal
5	R/W	MPU	Read/write select signal
6	E	MPU	Operation (data read/write) enable signal
7~10	DB0~DB3	MPU	Four low order bi-directional three-state data bus lines. Used for data transfer between the MPU and the LCM. These four are not used during 4-bit operation.
11~14	DB4~DB7	MPU	Four high order bi-directional three-state data bus lines. Used for data transfer between the MPU
15	LED+	LED BKL power supply	Power supply for BKL
16	LED-		Power supply for BKL

7. Contrast adjust



$V_{DD}-V_0$: LCD Driving voltage VR: 10k~20k

8. Optical characteristics



STN type display module ($T_a=25^{\circ}\text{C}$, $V_{DD}=3.3\text{V}$)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Viewing angle	$\theta 1$	$C_r \geq 3$		20		deg
	$\theta 2$			40		
	$\Phi 1$			35		
	$\Phi 2$			35		
Contrast ratio	C_r		-	10	-	-
Response time (rise)	T_r	-	-	200	250	ms
Response time (fall)	T_r	-	-	300	350	

9. Electrical characteristics

DC characteristics

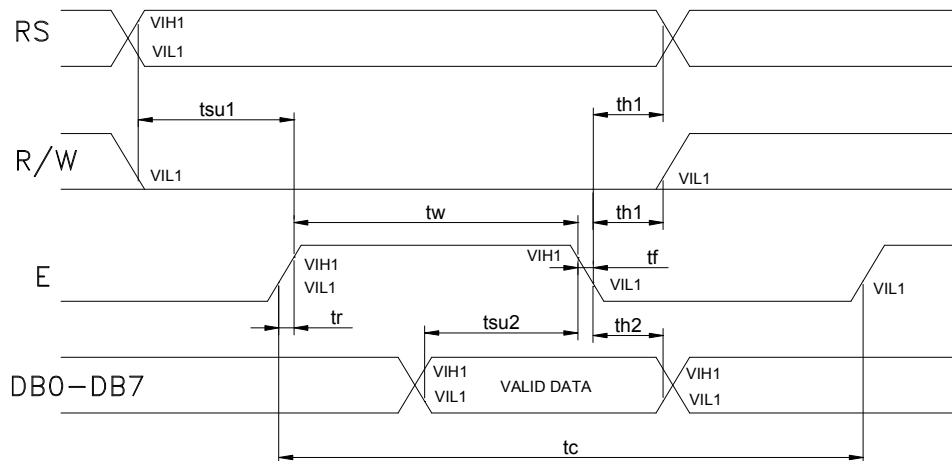
Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Supply voltage for LCD	$V_{DD}-V_0$	$T_a = 25^{\circ}\text{C}$	-	3.0	-	V
Input voltage	V_{DD}		3.1	3.3	3.5	
Supply current	I_{DD}	$T_a=25^{\circ}\text{C}$, $V_{DD}=3.3\text{V}$	-	1.5	2.5	mA
Input leakage current	I_{LKG}		-	-	1.0	μA
“H” level input voltage	V_{IH}		2.2	-	V_{DD}	V
“L” level input voltage	V_{IL}	Twice initial value or less	0	-	0.6	
“H” level output voltage	V_{OH}	$LOH=-0.25\text{mA}$	2.4	-	-	
“L” level output voltage	V_{OL}	$LOH=1.6\text{mA}$	-	-	0.4	
Backlight supply voltage	V_F		-	3.0		
Backlight supply current	I_{LED}	$V_{LED}=3.3\text{V}$ $R=25\Omega$			16	mA

10. Timing Characteristics

Write cycle ($T_a=25^{\circ}\text{C}$, $V_{DD}=3.3\text{V}$)

Parameter	Symbol	Test pin	Min.	Typ.	Max.	Unit
Enable cycle time	t_c	E	500	-	-	ns
Enable pulse width	t_w		300	-	-	
Enable rise/fall time	t_r, t_f		-	-	25	
RS; R/W setup time	t_{su1}	RS; R/W RS; R/W	100	-	-	
RS; R/W address hold time	t_{h1}		10	-	-	
Read data output delay	t_{su2}	DB0~DB7	60	-	-	
Read data hold time	t_{h2}		10	-	-	

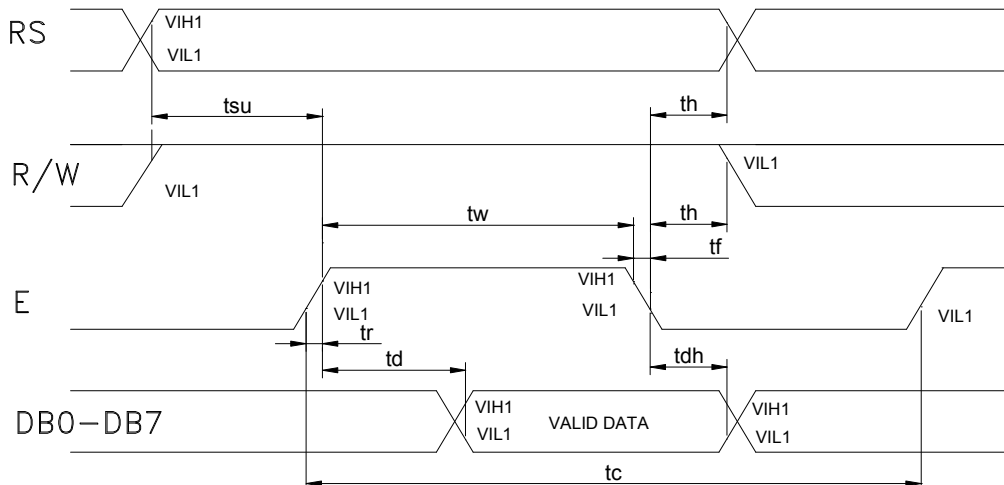
Write mode timing diagram



Read cycle ($T_a=25^{\circ}\text{C}$, $V_{DD}=3.3\text{V}$)

Parameter	Symbol	Test pin	Min.	Typ.	Max.	Unit
Enable cycle time	t_c	E	500	-	-	ns
Enable pulse width	t_w		300	-	-	
Enable rise/fall time	t_r, t_f		-	-	25	
RS; R/W setup time	t_{su}	RS; R/W RS; R/W	100	-	-	
RS; R/W address hold time	t_h		10	-	-	
Read data output delay	t_d	DB0~DB7	60	-	90	
Read data hold time	t_{dh}		20	-	-	

Read mode timing diagram



11. FUNCTION DESCRIPTION

11.1 System Interface

This chip has all two kinds of interface type with MPU : 4-bit bus and 8-bit bus. 4-bit bus and 8-bit bus is selected by DL bit in the instruction register.

11.2 Busy Flag (BF)

When BF = "High", it indicates that the internal operation is being processed. So during this time the next instruction cannot be accepted. BF can be read, when RS = Low and R/W = High (Read Instruction Operation), through DB7 port. Before executing the next instruction, be sure that BF is not high.

11.3 Address Counter (AC)

Address Counter (AC) stores DDRAM/CGRAM address, transferred from IR. After writing into (reading from) DDRAM/CGRAM, AC is automatically increased (decreased) by 1. When RS = "Low" and R/W = "High", AC can be read through DB0 – DB6 ports.

11.4 Display Data RAM (DDRAM)

DDRAM stores display data of maximum 80 x 8 bits (80 characters). DDRAM address is set in the address counter (AC) as a hexadecimal number.

Display position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
DDRAM address	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
DDRAM address	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F

11.5 CGROM (Character Generator ROM)

CGROM has a 5 x 8 dots 204 characters pattern and a 5 x 10 dots 32 characters pattern. CGROM has 204 character patterns of 5 x 8 dots.

11.6 CGRAM (Character Generator RAM)

CGRAM has up to 5 , 8 dot, 8 characters. By writing font data to CGRAM, user defined characters can be used.

Character Code (DDRAM Data)								CGRAM Address							Character Patterns (CGRAM Data)							
b8	b7	b6	b5	b4	b3	b2	b1	b0	b5	b4	b3	b2	b1	b0	b7	b6	b5	b4	b3	b2	b1	b0
0	0	0	0	0	-	0	0	0	0	0	0	0	0	0	-	-	-	1	1	1	1	1
						0	0	0				0	0	1				0	0	1	0	0
						0	0	0				0	1	0				0	0	1	0	0
						0	0	0				0	1	1				0	0	1	0	0
						0	0	0				1	0	0				0	0	1	0	0
						0	0	0				1	0	1				0	0	1	0	0
						0	0	0				1	1	0				0	0	1	0	0
						0	0	0				1	1	1				0	0	0	0	0
0	0	0	0	0	-	0	0	1	0	0	1	0	0	0	-	-	-	1	1	1	1	0
						0	0	1				0	0	1				1	0	0	0	1
						0	0	1				0	1	0				1	0	0	0	1
						0	0	1				0	1	1				1	1	1	1	0
						0	0	1				1	0	0				1	0	1	0	0
						0	0	1				1	0	1				1	0	0	1	0
						0	0	1				1	1	0				1	0	0	0	1
						0	0	1				1	1	1				0	0	0	0	0

Relationship between CGRAM Addresses, Character Codes (DDRAM) and Character patterns (CGRAM Data)

Notes:

1. Character code bits 0 to 2 correspond to CGRAM address bits 3 to 5 (3 bits: 8 types).
2. CGRAM address bits 0 to 2 designate the character pattern line position. The 8th line is the cursor position

and its display is formed by a logical OR with the cursor. Maintain the 8th line data, corresponding to the cursor display position, at 0 as the cursor display. If the 8th line data is 1, 1 bit will light up the 8th line regardless of the cursor presence.

3. Character pattern row positions correspond to CGRAM data bits 0 to 4 (bit 4 being at the left).

4. As shown Table, CGRAM character patterns are selected when character code bits 4 to 7 are all 0. However, since character code bit 3 has no effect, the R display example above can be selected by either character code 00H or 08H.

5. 1 for CGRAM data corresponds to display selection and 0 to non-selection.

“-“: Indicates no effect.

11.7 Cursor/Blink Control Circuit

It controls cursor/blink ON/OFF at cursor position.

11.8 Outline

To overcome the speed difference between the internal clock of ST7066 and the MPU clock, ST7066 performs internal operations by storing control in formations to IR or DR. The internal operation is determined according to the signal from MPU, composed of read/write and data bus (Refer to Table7).

Instructions can be divided largely into four groups:

- 1) ST7066 function set instructions (set display methods, set data length, etc.)
- 2) Address set instructions to internal RAM
- 3) Data transfer instructions with internal RAM
- 4) Others

The address of the internal RAM is automatically increased or decreased by 1.

Note: during internal operation, busy flag (DB7) is read “High”.

Busy flag check must be preceded by the next instruction.

11.9 Instruction Table

Instruction	Instruction code										Description	Execution time (fosc= 270 KHZ)
	RS	R/M	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
Clear Display	0	0	0	0	0	0	0	0	0	1	Write "20H" to DDRA and set DDRAM address to "00H" from AC	1.53ms
Return Home	0	0	0	0	0	0	0	0	1	-	Set DDRAM address to "00H" From AC and return cursor to its original position if shifted. The contents of DDRAM are not changed.	1.53ms
Entry mode Set	0	0	0	0	0	0	0	1	I/D	SH	Assign cursor moving direction And blinking of entire display	39us
Display ON/OFF control	0	0	0	0	0	0	1	D	C	B	Set display (D), cursor (C), and Blinking of cursor (B) on/off Control bit.	
Cursor or Display shift	0	0	0	0	0	1	S/C	R/L	-	-	Set cursor moving and display Shift control bit, and the Direction, without changing of DDRAM data.	39us
Function set	0	0	0	0	1	DL	N	F	-	-	Set interface data length (DL: 8-Bit/4-bit), numbers of display Line (N: =2-line/1-line) and, Display font type (F: 5x11/5x8)	39us
Set CGRAM Address	0	0	0	1	AC5	AC4	AC3	AC2	AC1	AC0	Set CGRAM address in address Counter.	39us
Set DDRAM Address	0	0	1	AC6	AC5	AC4	AC3	AC2	AC1	AC0	Set DDRAM address in address Counter.	39us
Read busy Flag and Address	0	1	BF	AC6	AC5	AC4	AC3	AC2	AC1	AC0	Whether during internal Operation or not can be known By reading BF. The contents of Address counter can also be read.	0us
Write data to Address	1	0	D7	D6	D5	D4	D3	D2	D1	D0	Write data into internal RAM (DDRAM/CGRAM).	43us
Read data From RAM	1	1	D7	D6	D5	D4	D3	D2	D1	D0	Read data from internal RAM (DDRAM/CGRAM).	43us

NOTE:

When an MPU program with checking the busy flag (DB7) is made, it must be necessary 1/2fosc is necessary for executing the next instruction by the falling edge of the "E" signal after the busy flag (DB7) goes to "Low".

11.3Contents

1) Clear display

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	0	0	1

Clear all the display data by writing "20H" (space code) to all DDRAM address, and set DDRAM address to "00H" into AC (address counter).

Return cursor to the original status, namely, bring the cursor to the left edge on the first line of the display. Make the entry mode increment (I/D="High").

2) Return home

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	0	1	-

Return home is cursor return home instruction.

Set DDRAM address to "00H" into the address counter.
 Return cursor to its original site and return display to its original status, if shifted.
 Contents of DDRAM does not change.

3) Entry mode set

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	1	I/D	SH

Set the moving direction of cursor and display.

I/D: increment / decrement of DDRAM address (cursor or blink)

When I/D="high", cursor/blink moves to right and DDRAM address is increased by 1.

When I/D="Low", cursor/blink moves to left and DDRAM address is increased by 1.

*CGRAM operates the same way as DDRAM, when reading from or writing to CGRAM.

SH: shift of entire display

When DDRAM read (CGRAM read/write) operation or SH="Low", shifting of entire display is not performed. If SH="High" and DDRAM write operation, shift of entire display is performed according to I/D value. (I/D="high". shift left, I/D="Low". Shift right).

4) Display ON/OFF control

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	1	D	C	B

Control display/cursor/blink ON/OFF 1 bit register.

D: Display ON/OFF control bit

When D="High", entire display is turned on.

When D="Low", display is turned off, but display data remains in DDRAM.

C: cursor ON/OFF control bit

When D="High", cursor is turned on.

When D="Low", cursor is disappeared in current display, but I/D register preserves its data.

B: Cursor blink ON/OFF control bit

When B="High", cursor blink is on, which performs alternately between all the "High" data and display characters at the cursor position.

When B="Low", blink is off.

5) Cursor or display shift

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	1	S/C	R/L	-	-

Shifting of right/left cursor position or display without writing or reading of display data.

This instruction is used to correct or search display data.

During 2-line mode display, cursor moves to the 2nd line after the 40th digit of the 1st line.

Note that display shift is performed simultaneously in all the lines.

When display data is shifted repeatedly, each line is shifted individually.

When display shift is performed, the contents of the address counter are not changed.

Shift patterns according to S/C and R/L bits

S/C	R/L	Operation
0	0	Shift cursor to the left, AC is decreased by 1
0	1	Shift cursor to the right, AC is increased by 1
1	0	Shift all the display to the left, cursor moves according to the display
1	1	Shift all the display to the right, cursor moves according to the display

6) Function set

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	1	DL	N	F	-	-

DL: Interface data length control bit

When DL="High", it means 8-bit bus mode with MPU.

When DL="Low", it means 4-bit bus mode with MPU. Hence, DL is a signal to select 8-bit or 4-bit bus mode.

When 4-bit bus mode, it needs to transfer 4-bit data twice.

N: Display line number control bit

When N="Low", 1-line display mode is set.

When N="High", 2-line display mode is set.

F: Display line number control bit

When F="Low", 5x8 dots format display mode is set.

When F="High", 5x11 dots format display mode.

7) Set CGRAM address

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	1	AC5	AC4	AC3	AC2	AC1	AC0

Set CGRAM address to AC.

The instruction makes CGRAM data available from MPU.

8) Set DDRAM address

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	1	AC6	AC5	AC4	AC3	AC2	AC1	AC0

Set DDRAM address to AC.

This instruction makes DDRAM data available from MPU.

When 1-line display mode (N=LOW), DDRAM address is from "00H" to "4FH". In 2-line display mode (N=High), DDRAM address in the 1st line is from "00H" to "27H", and DDRAM address in the 2nd line is from "40H" to "67H".

9) Read busy flag & address

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	1	BF	AC6	AC5	AC4	AC3	AC2	AC1	AC0

This instruction shows whether SPLC780D is in internal operation or not.

If the resultant BF is "High", internal operation is in progress and should wait BF is to be LOW, which by then the next instruction can be performed. In this instruction you can also read the value of the address counter.

10) Write data to RAM

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
1	0	D7	D6	D5	D4	D3	D2	D1	D0

Write binary 8-bit data to DDRAM/CGRAM.

The selection of RAM from DDRAM, and CGRAM, is set by the previous address set instruction (DDRAM address set, CGRAM address set).

RAM set instruction can also determine the AC direction to RAM.

After write operation. The address is automatically increased/decreased by 1, according to the entry mode.

11) Read data from RAM

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
1	1	D7	D6	D5	D4	D3	D2	D1	D0

Read binary 8-bit data from DDRAM/CGRAM.

The selection of RAM is set by the previous address set instruction. If the address set instruction of RAM is not performed before this instruction, the data that has been read first is invalid, as the direction of AC is not yet determined. If RAM data is read several times without RAM address instructions set before, read operation, the correct RAM data can be obtained from the second. But the first data would be incorrect, as there is no time margin to transfer RAM data.

In case of DDRAM read operation, cursor shift instruction plays the same role as DDRAM address set

instruction, it also transfers RAM data to output data register.

After read operation, address counter is automatically increased/decreased by 1 according to the entry mode.

After CGRAM read operation, display shift may not be executed correctly.

NOTE: In case of RAM write operation, AC is increased/decreased by 1 as in read operation.

At this time, AC indicates next address position, but only the previous data can be read by the read instruction.

12. Standard character pattern

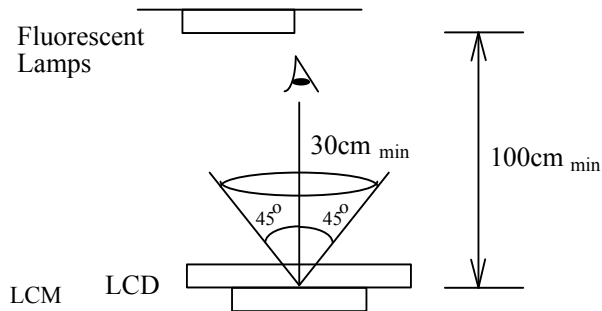
Upper 4bit Lower 4bit		LLLL	LLH	LLHL	LLHH	LHLL	LHLH	LHHL	LHHH	HLLL	HLLH	HLHL	HLHH	HHLL	HHLH	HHHL	HHHH
LLLL	CG RAM (1)																
LLH	(2)																
LLHL	(3)																
LLHH	(4)																
LHLL	(5)																
LHLH	(6)																
LHHL	(7)																
LHHH	(8)																
HLLL	(1)																
HLLH	(2)																
HLHL	(3)																
HLHH	(4)																
HHLL	(5)																
HHLH	(6)																
HHHL	(7)																
HHHH	(8)																

13. QUALITY SPECIFICATIONS

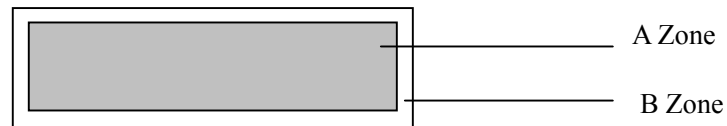
13.1 Standard of the product appearance test

Manner of appearance test: The inspection should be performed in using 20W x 2 fluorescent lamps. Distance between LCM and fluorescent lamps should be 100 cm or more. Distance between LCM and inspector eyes should be 30 cm or more.

Viewing direction for inspection is 45° from vertical against LCM.



Definition of zone:



A Zone: Active display area (minimum viewing area).

B Zone: Non-active display area (outside viewing area).

13.2 Specification of quality assurance

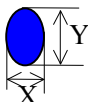
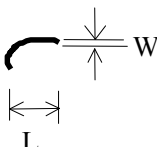
AQL inspection standard

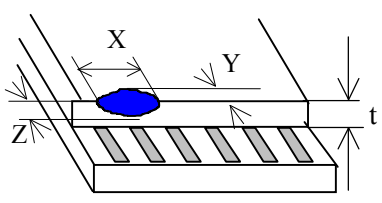
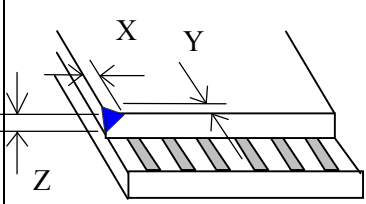
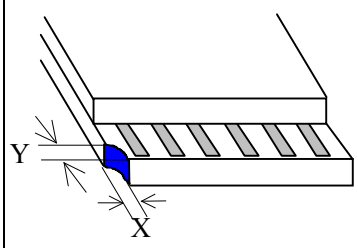
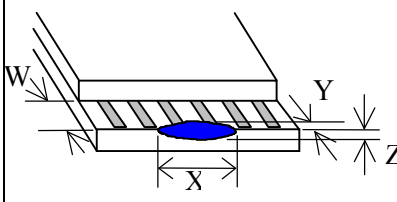
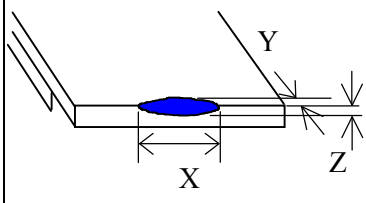
Sampling method: MIL-STD-105E, Level II, single sampling

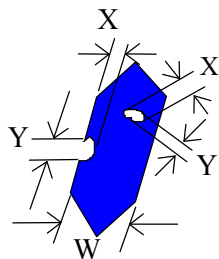
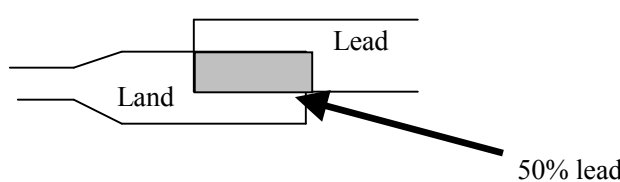
Defect classification **(Note: * is not including)**

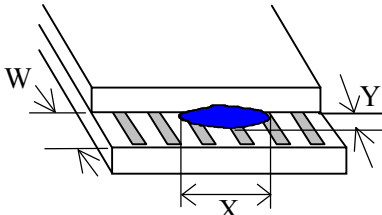
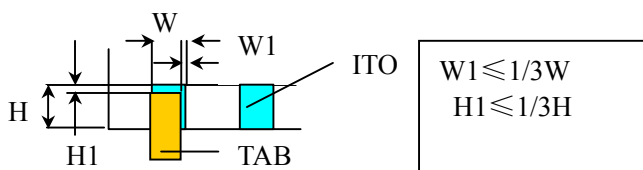
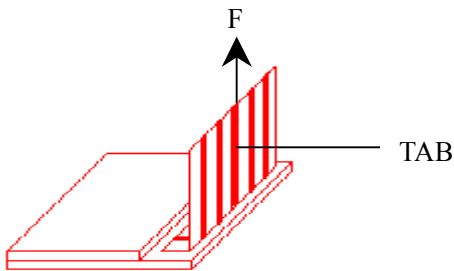
Classify	Item		Note	AQL
Major	Display state	Short or open circuit	1	0.65
		LC leakage		
		Flickering		
		No display		
		Wrong viewing direction		
		Contrast defect (dim, ghost)	2	
		Back-light	1,8	
	Non-display	Flat cable or pin reverse	10	
		Wrong or missing component	11	
Minor	Display state	Background color deviation	2	1.0
		Black spot and dust	3	
		Line defect, Scratch	4	
		Rainbow	5	
		Chip	6	
		Pin hole	7	
	Polarizer	Protruded	12	
		Bubble and foreign material	3	
	Soldering	Poor connection	9	
	Wire	Poor connection	10	
	TAB	Position, Bonding strength	13	

Note on defect classification

No.	Item	Criterion																				
1	Short or open circuit	Not allow																				
	LC leakage																					
	Flickering																					
	No display																					
	Wrong viewing direction																					
	Wrong Back-light																					
2	Contrast defect	Refer to approval sample																				
	Background color deviation																					
3	Point defect, Black spot, dust (including Polarizer) $\phi = (X+Y)/2$	 <table><tr><th>Point Size</th><th>Acceptable Qty.</th></tr><tr><td>$\phi \leq 0.10$</td><td>Disregard</td></tr><tr><td>$0.10 < \phi \leq 0.20$</td><td>3</td></tr><tr><td>$0.20 < \phi \leq 0.25$</td><td>2</td></tr><tr><td>$0.25 < \phi \leq 0.30$</td><td>1</td></tr><tr><td>$\phi > 0.30$</td><td>0</td></tr></table> Unit: mm	Point Size	Acceptable Qty.	$\phi \leq 0.10$	Disregard	$0.10 < \phi \leq 0.20$	3	$0.20 < \phi \leq 0.25$	2	$0.25 < \phi \leq 0.30$	1	$\phi > 0.30$	0								
Point Size	Acceptable Qty.																					
$\phi \leq 0.10$	Disregard																					
$0.10 < \phi \leq 0.20$	3																					
$0.20 < \phi \leq 0.25$	2																					
$0.25 < \phi \leq 0.30$	1																					
$\phi > 0.30$	0																					
4	Line defect, Scratch	 <table><tr><th colspan="2">Line</th><th>Acceptable Qty.</th></tr><tr><th>L</th><th>W</th><th></th></tr><tr><td>---</td><td>$0.015 \geq W$</td><td>Disregard</td></tr><tr><td>$3.0 \geq L$</td><td>$0.03 \geq W$</td><td rowspan="2">2</td></tr><tr><td>$2.0 \geq L$</td><td>$0.05 \geq W$</td></tr><tr><td>$1.0 \geq L$</td><td>$0.1 > W$</td><td>1</td></tr><tr><td>---</td><td>$0.05 < W$</td><td>Applied as point defect</td></tr></table> Unit: mm	Line		Acceptable Qty.	L	W		---	$0.015 \geq W$	Disregard	$3.0 \geq L$	$0.03 \geq W$	2	$2.0 \geq L$	$0.05 \geq W$	$1.0 \geq L$	$0.1 > W$	1	---	$0.05 < W$	Applied as point defect
Line		Acceptable Qty.																				
L	W																					
---	$0.015 \geq W$	Disregard																				
$3.0 \geq L$	$0.03 \geq W$	2																				
$2.0 \geq L$	$0.05 \geq W$																					
$1.0 \geq L$	$0.1 > W$	1																				
---	$0.05 < W$	Applied as point defect																				
5	Rainbow	Not more than two color changes across the viewing area.																				

No	Item	Criterion																																
6	Chip Remark: X: Length direction Y: Short direction Z: Thickness direction t: Glass thickness W: Terminal Width	<table><caption>Acceptable criterion</caption><tr><th>X</th><th>Y</th><th>Z</th></tr><tr><td>≤2</td><td>0.5mm</td><td>≤t/2</td></tr></table> <table><caption>Acceptable criterion</caption><tr><th>X</th><th>Y</th><th>Z</th></tr><tr><td>≤2</td><td>0.5mm</td><td>≤t</td></tr></table> <table><caption>Acceptable criterion</caption><tr><th>X</th><th>Y</th><th>Z</th></tr><tr><td>≤3</td><td>≤2</td><td rowspan="2">≤t</td></tr><tr><td colspan="2">shall not reach to ITO</td></tr></table> <table><caption>Acceptable criterion</caption><tr><th>X</th><th>Y</th><th>Z</th></tr><tr><td>Disregard</td><td>≤0.2</td><td>≤t</td></tr></table> <table><caption>Acceptable criterion</caption><tr><th>X</th><th>Y</th><th>Z</th></tr><tr><td>≤5</td><td>≤2</td><td>≤t/3</td></tr></table>	X	Y	Z	≤2	0.5mm	≤t/2	X	Y	Z	≤2	0.5mm	≤t	X	Y	Z	≤3	≤2	≤t	shall not reach to ITO		X	Y	Z	Disregard	≤0.2	≤t	X	Y	Z	≤5	≤2	≤t/3
X	Y	Z																																
≤2	0.5mm	≤t/2																																
X	Y	Z																																
≤2	0.5mm	≤t																																
X	Y	Z																																
≤3	≤2	≤t																																
shall not reach to ITO																																		
X	Y	Z																																
Disregard	≤0.2	≤t																																
X	Y	Z																																
≤5	≤2	≤t/3																																

No.	Item	Criterion								
7	Segment pattern W = Segment width $\phi = (X+Y)/2$	(1) Pin hole $\phi < 0.10\text{mm}$ is acceptable.  <table><tr><th>Point Size</th><th>Acceptable Qty</th></tr><tr><td>$\phi \leq 1/4W$</td><td>Disregard</td></tr><tr><td>$1/4W < \phi \leq 1/2W$</td><td>1</td></tr><tr><td>$\phi > 1/2W$</td><td>0</td></tr></table> Unit: mm	Point Size	Acceptable Qty	$\phi \leq 1/4W$	Disregard	$1/4W < \phi \leq 1/2W$	1	$\phi > 1/2W$	0
Point Size	Acceptable Qty									
$\phi \leq 1/4W$	Disregard									
$1/4W < \phi \leq 1/2W$	1									
$\phi > 1/2W$	0									
8	Back-light	(1) The color of backlight should correspond its specification. (2) Not allow flickering								
9	Soldering	(1) Not allow heavy dirty and solder ball on PCB. (The size of dirty refer to point and dust defect) (2) Over 50% of lead should be solderedon Land. 								
10	Wire	(1) Copper wire should not be rusted (2) Not allow crack on copper wire connection. (3) Not allow reversing the position of the flat cable. (4) Not allow exposed copper wire inside the flat cable.								
11*	PCB	(1) Not allow screw rust or damage. (2) Not allow missing or wrong putting of component.								

No	Item	Criterion
12	Protruded W: Terminal Width	 Acceptable criteria: $Y \leq 0.4$
13	TAB	1. Position  2. TAB bonding strength test  $P (=F/\text{TAB bonding width}) \geq 650\text{gf/cm}$,(speed rate: 1mm/min) 5pcs per SOA (shipment)
14	Total no. of acceptable Defect	A. Zone Maximum 2 minor non-conformities per one unit. Defect distance: each point to be separated over 10mm B. Zone It is acceptable when it is no trouble for quality and assembly in customer's end product.

13.3 Reliability of LCM

Reliability test condition:

Item	Condition	Time (hrs)	Assessment
High temp. Storage	80°C	48	No abnormalities in functions and appearance
High temp. Operating	70°C	48	
Low temp. Storage	-30°C	48	
Low temp. Operating	-20°C	48	
Humidity	40°C/ 90%RH	48	
Temp. Cycle	0°C ← 25°C → 50°C (30 min ← 5 min → 30min)	10cycles	

Recovery time should be 24 hours minimum. Moreover, functions, performance and appearance shall be free from remarkable deterioration within 50,000 hours under ordinary operating and storage conditions room temperature (20±8°C), normal humidity (below 65% RH), and in the area not exposed to direct sun light.

13.4 Precaution for using LCD/LCM

LCD/LCM is assembled and adjusted with a high degree of precision. Do not attempt to make any alteration or modification. The followings should be noted.

General Precautions:

1. LCD panel is made of glass. Avoid excessive mechanical shock or applying strong pressure onto the surface of display area.
2. The polarizer used on the display surface is easily scratched and damaged. Extreme care should be taken when handling. To clean dust or dirt off the display surface, wipe gently with cotton, or other soft material soaked with isopropyl alcohol, ethyl alcohol or trichlorotrifluoroethane, do not use water, ketone or aromatics and never scrub hard.
3. Do not tamper in any way with the tabs on the metal frame.
4. Do not make any modification on the PCB without consulting AMOTEC
5. When mounting a LCM, make sure that the PCB is not under any stress such as bending or twisting. Elastomer contacts are very delicate and missing pixels could result from slight dislocation of any of the elements.
6. Avoid pressing on the metal bezel, otherwise the elastomer connector could be deformed and lose contact, resulting in missing pixels and also cause rainbow on the display.
7. Be careful not to touch or swallow liquid crystal that might leak from a damaged cell. Any liquid crystal adheres to skin or clothes, wash it off immediately with soap and water.

Static Electricity Precautions:

1. CMOS LSI is used for the module circuit; therefore operators should be grounded whenever he/she comes into contact with the module.
2. Do not touch any of the conductive parts such as the LSI pads; the copper leads on the PCB and the interface terminals with any parts of the human body.

-
3. Do not touch the connection terminals of the display with bare hand; it will cause disconnection or defective insulation of terminals.
 4. The modules should be kept in anti-static bags or other containers resistant to static for storage.
 5. Only properly grounded soldering irons should be used.
 6. If an electric screwdriver is used, it should be grounded and shielded to prevent sparks.
 7. The normal static prevention measures should be observed for work clothes and working benches.
 8. Since dry air is inductive to static, a relative humidity of 50-60% is recommended.

Soldering Precautions:

1. Soldering should be performed only on the I/O terminals.
2. Use soldering irons with proper grounding and no leakage.
3. Soldering temperature: $280^{\circ}\text{C} \pm 10^{\circ}\text{C}$
4. Soldering time: 3 to 4 second.
5. Use eutectic solder with resin flux filling.
6. If flux is used, the LCD surface should be protected to avoid spattering flux.
7. Flux residue should be removed.

Operation Precautions:

1. The viewing angle can be adjusted by varying the LCD driving voltage V_o .
2. Since applied DC voltage causes electro-chemical reactions, which deteriorate the display, the applied pulse waveform should be a symmetric waveform such that no DC component remains. Be sure to use the specified operating voltage.
3. Driving voltage should be kept within specified range; excess voltage will shorten display life.
4. Response time increases with decrease in temperature.
5. Display color may be affected at temperatures above its operational range.
6. Keep the temperature within the specified range usage and storage. Excessive temperature and humidity could cause polarization degradation, polarizer peel-off or generate bubbles.
7. For long-term storage over 40 °C is required, the relative humidity should be kept below 60%, and avoid direct sunlight.

Limited Warranty

AMOTEC LCDs and modules are not consumer products, but may be incorporated by AMOTEC 's customers into consumer products or components thereof, AMOTEC does not warrant that its LCDs and components are fit for any such particular purpose.

1. The liability of AMOTEC is limited to repair or replacement on the terms set forth below. AMOTEC will not be responsible for any subsequent or consequential events or injury or damage to any personnel or user including third party personnel and/or user. Unless otherwise agreed in writing between AMOTEC and the customer, AMOTEC will only replace or repair any of its LCD which is found defective electrically or visually when inspected in accordance with AMOTEC general LCD inspection standard . (Copies available on request)
2. No warranty can be granted if any of the precautions state in handling liquid crystal display above has been disregarded. Broken glass, scratches on polarizer mechanical damages as well as defects that are caused accelerated environment tests are excluded from warranty.
3. In returning the LCD/LCM, they must be properly packaged; there should be detailed description of the failures or defect.

PING)))™ Ultrasonic Distance Sensor (#28015)

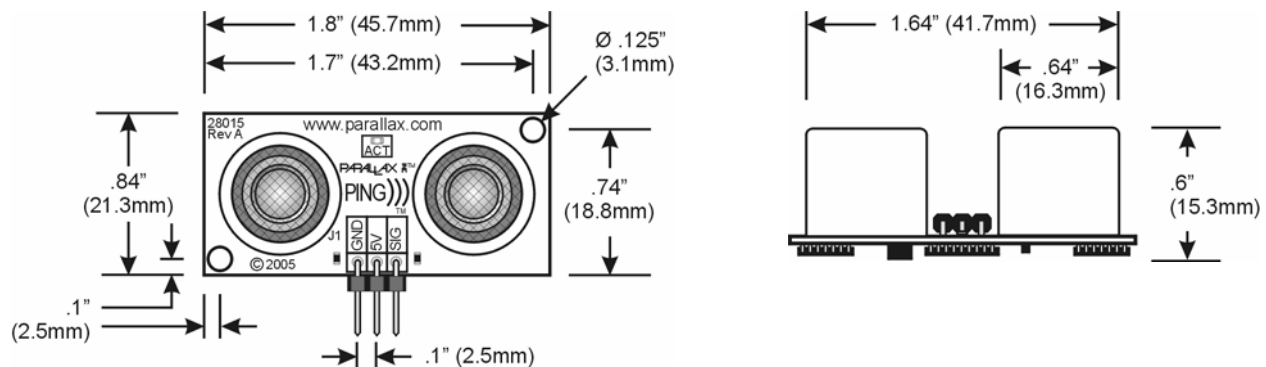
The Parallax PING))) ultrasonic distance sensor provides precise, non-contact distance measurements from about 2 cm (0.8 inches) to 3 meters (3.3 yards). It is very easy to connect to BASIC Stamp® or Javelin Stamp microcontrollers, requiring only one I/O pin.

The PING))) sensor works by transmitting an ultrasonic (well above human hearing range) burst and providing an output pulse that corresponds to the time required for the burst echo to return to the sensor. By measuring the echo pulse width the distance to target can easily be calculated.

Features

- Supply Voltage – 5 VDC
- Supply Current – 30 mA typ; 35 mA max
- Range – 2 cm to 3 m (0.8 in to 3.3 yds)
- Input Trigger – positive TTL pulse, 2 μ s min, 5 μ s typ.
- Echo Pulse – positive TTL pulse, 115 μ s to 18.5 ms
- Echo Hold-off – 750 μ s from fall of Trigger pulse
- Burst Frequency – 40 kHz for 200 μ s
- Burst Indicator LED shows sensor activity
- Delay before next measurement – 200 μ s
- Size – 22 mm H x 46 mm W x 16 mm D (0.84 in x 1.8 in x 0.6 in)

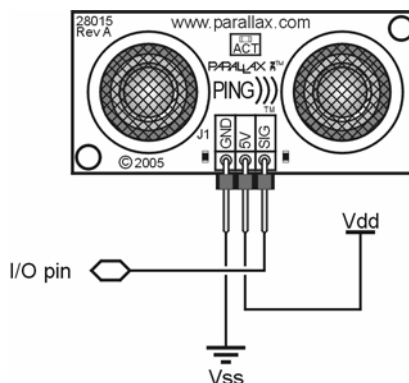
Dimensions



Pin Definitions

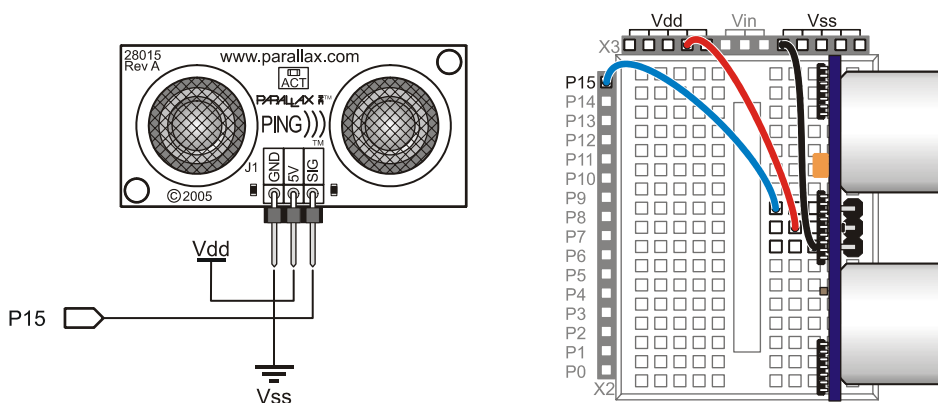
GND	Ground (Vss)
5 V	5 VDC (Vdd)
SIG	Signal (I/O pin)

The PING))) sensor has a male 3-pin header used to supply power (5 VDC), ground, and signal. The header allows the sensor to be plugged into a solderless breadboard, or to be located remotely through the use of a standard servo extender cable (Parallax part #805-00002). Standard connections are shown in the diagram to the right.



Quick-Start Circuit

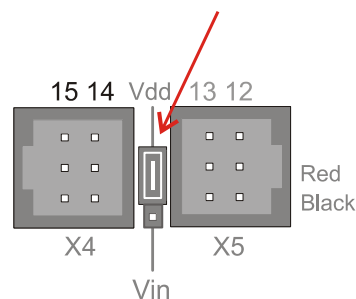
This circuit allows you to quickly connect your PING))) sensor to a BASIC Stamp® 2 via the Board of Education® breadboard area. The PING))) module's GND pin connects to Vss, the 5 V pin connects to Vdd, and the SIG pin connects to I/O pin P15. This circuit will work with the example program Ping_Demo.BS2 listed on page 7.



Servo Cable and Port Cautions

If you want to connect your PING))) sensor to a Board of Education using a servo extension cable, follow these steps:

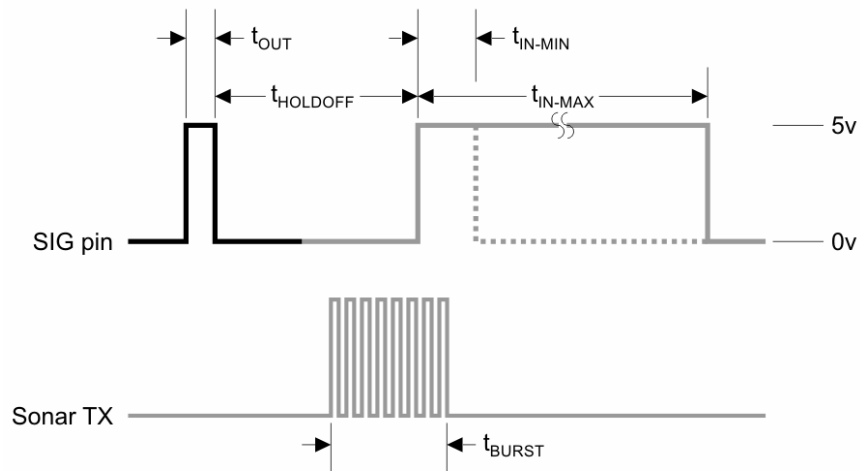
1. When plugging the cable onto the PING))) sensor, connect Black to GND, Red to 5 V, and White to SIG.
2. Check to see if your Board of Education servo ports have a jumper, as shown at right.
3. If your Board of Education servo ports have a jumper, set it to Vdd as shown.
4. If your Board of Education servo ports do not have a jumper, do not use them with the PING))) sensor. These ports only provide Vin, not Vdd, and this may damage your PING))) sensor. Go to the next step.
5. Connect the servo cable directly to the breadboard with a 3-pin header. Then, use jumper wires to connect Black to Vss, Red to Vdd, and White to I/O pin P15.



Board of Education Servo Port Jumper, Set to Vdd

Theory of Operation

The PING))) sensor detects objects by emitting a short ultrasonic burst and then "listening" for the echo. Under control of a host microcontroller (trigger pulse), the sensor emits a short 40 kHz (ultrasonic) burst. This burst travels through the air at about 1130 feet per second, hits an object and then bounces back to the sensor. The PING))) sensor provides an output pulse to the host that will terminate when the echo is detected, hence the width of this pulse corresponds to the distance to the target.



— HOST	t_{OUT}	2 μ S (min), 5 μ S typical
— PING	$t_{HOLDOFF}$	750 μ S
	t_{BURST}	200 μ S @ 40 kHz
	t_{IN-MIN}	115 μ S
	t_{IN-MAX}	18.5 mS

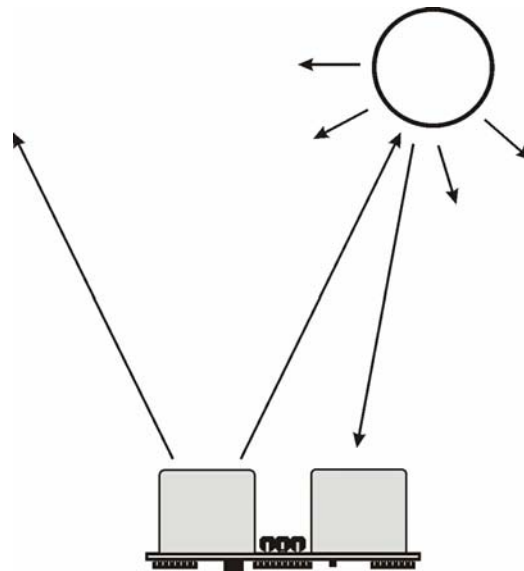
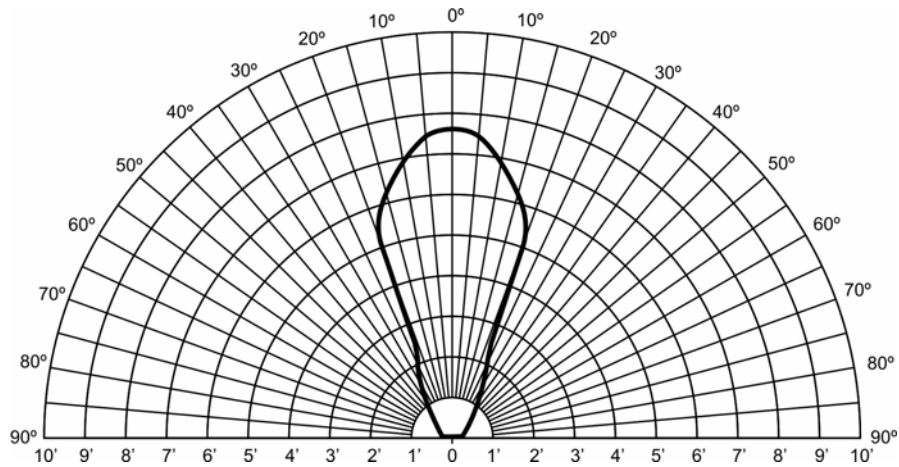
Test Data

The test data on the following pages is based on the PING))) sensor, tested in the Parallax lab, while connected to a BASIC Stamp microcontroller module. The test surface was a linoleum floor, so the sensor was elevated to minimize floor reflections in the data. All tests were conducted at room temperature, indoors, in a protected environment. The target was always centered at the same elevation as the PING))) sensor.

Test 1

Sensor Elevation: 40 in. (101.6 cm)

Target: 3.5 in. (8.9 cm) diameter cylinder, 4 ft. (121.9 cm) tall – vertical orientation

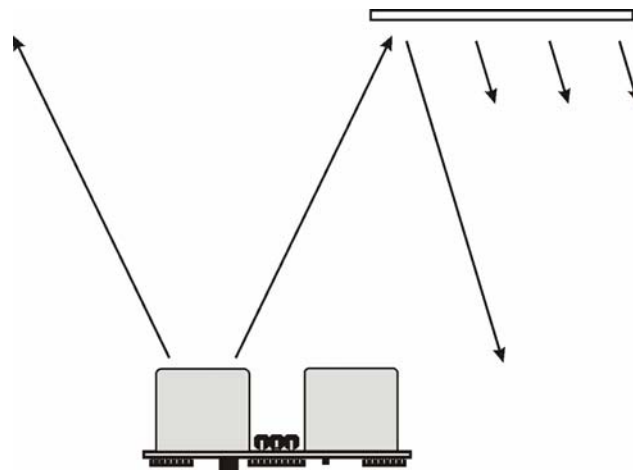
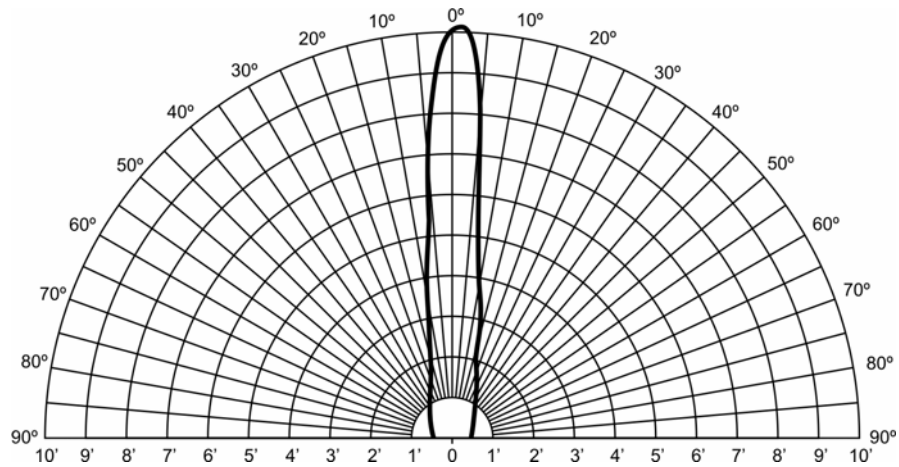


Test 2

Sensor Elevation: 40 in. (101.6 cm)

Target: 12 in. x 12 in. (30.5 cm x 30.5 cm) cardboard, mounted on 1 in. (2.5 cm) pole

- target positioned parallel to backplane of sensor



Program Example: BASIC Stamp 2 Microcontroller

The following program demonstrates the use of the PING))) sensor with the BASIC Stamp 2 microcontroller. Any model of BASIC Stamp 2 module will work with this program as conditional compilation techniques are used to make adjustments based on the module that is connected.

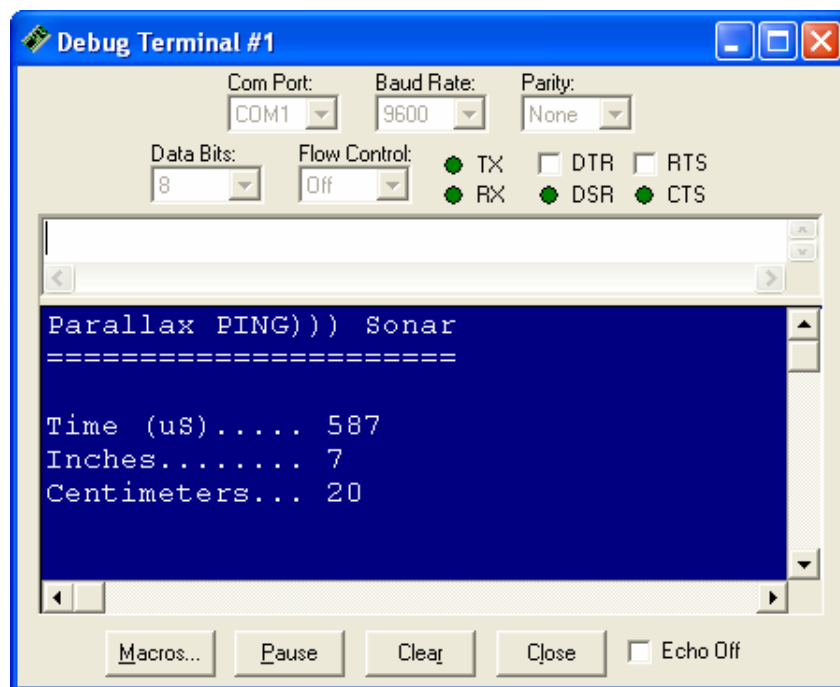
The heart of the program is the **Get_Sonar** subroutine. This routine starts by making the output bit of the selected IO pin zero – this will cause the successive **PULSOUT** to be low-high-low as required for triggering the PING))) sensor. After the trigger pulse falls the sensor will wait about 200 microseconds before transmitting the ultrasonic burst. This allows the BS2 to load and prepare the next instruction. That instruction, **PULSIN**, is used to measure the high-going pulse that corresponds to the distance to the target object.

The raw return value from **PULSIN** must be scaled due to resolution differences between the various members of the BS2 family. After the raw value is converted to microseconds, it is divided by two in order to remove the "return trip" of the echo pulse. The value now held in *rawDist* is the distance to the target in microseconds.

Conversion from microseconds to inches (or centimeters) is now a simple matter of math. The generally-accepted value for the speed-of-sound is 1130 feet per second. This works out to 13,560 inches per second or one inch in 73.746 microseconds. The question becomes, how do we divide our pulse measurement value by the floating-point number 73.746?

Another way to divide by 73.746 is to multiply by 0.01356. For new BASIC Stamp users this may seem a dilemma but in fact there is a special operator, ******, that allows us to do just that. The ****** operator has the affect of multiplying a value by units of 1/65,536. To find the parameter for ****** then, we simply multiply 0.01356 by 65,536; the result is 888.668 (we'll round up to 889).

Conversion to centimeters uses the same process and the result of the program is shown below:



```

' =====
'
' File..... Ping_Demo.BS2
' Purpose.... Demo Code for Parallax PING))) Sonar Sensor
' Author..... Parallax, Inc.
' E-mail..... support@parallax.com
' Started....
' Updated.... 08 JUN 2005
'
'   {$STAMP BS2}
'   {$PBASIC 2.5}
'
' =====

' -----[ Program Description ]-----
'
' This program demonstrates the use of the Parallax PING))) sensor and then
' converting the raw measurement to English (inches) and Metric (cm) units.
'
' Sonar Math:
'
' At sea level sound travels through air at 1130 feet per second. This
' equates to 1 inch in 73.746 uS, or 1 cm in 29.034 uS).
'
' Since the PING))) sensor measures the time required for the sound wave to
' travel from the sensor and back. The result -- after conversion to
' microseconds for the BASIC Stamp module in use -- is divided by two to
' remove the return portion of the echo pulse. The final raw result is
' the duration from the front of the sensor to the target in microseconds.

' -----[ I/O Definitions ]-----

Ping          PIN      15

' -----[ Constants ]-----

#SELECT $STAMP
#CASE BS2, BS2E
  Trigger      CON      5              ' trigger pulse = 10 uS
  Scale        CON      $200          ' raw x 2.00 = uS
#CASE BS2SX, BS2P, BS2PX
  Trigger      CON      13
  Scale        CON      $0CD          ' raw x 0.80 = uS
#CASE BS2PE
  Trigger      CON      5
  Scale        CON      $1E1          ' raw x 1.88 = uS
#ENDSELECT

RawToIn        CON      889            ' 1 / 73.746 (with **)
RawToCm        CON      2257          ' 1 / 29.034 (with **)

IsHigh         CON      1              ' for PULSOUT
IsLow          CON      0

```

```

' -----[ Variables ]-----
rawDist      VAR      Word      ' raw measurement
inches       VAR      Word
cm           VAR      Word

' -----[ Initialization ]-----

Reset:
  DEBUG CLS,
    "Parallax PING))) Sonar", CR,      ' setup report screen
    "=====", CR,
    CR,
    "Time (uS).....", CR,
    "Inches.....", CR,
    "Centimeters..."

' -----[ Program Code ]-----

Main:
  DO
    GOSUB Get_Sonar      ' get sensor value
    inches = rawDist ** RawToIn  ' convert to inches
    cm = rawDist ** RawToCm      ' convert to centimeters

    DEBUG CRSRXY, 15, 3,      ' update report screen
      DEC rawDist, CLREOL,
      CRSRXY, 15, 4,
      DEC inches, CLREOL,
      CRSRXY, 15, 5,
      DEC cm, CLREOL

    PAUSE 100
  LOOP
END

' -----[ Subroutines ]-----

' This subroutine triggers the PING))) sonar sensor and measures
' the echo pulse. The raw value from the sensor is converted to
' microseconds based on the Stamp module in use. This value is
' divided by two to remove the return trip -- the result value is
' the distance from the sensor to the target in microseconds.

Get_Sonar:
  Ping = IsLow      ' make trigger 0-1-0
  PULSOUT Ping, Trigger  ' activate sensor
  PULSIN Ping, IsHigh, rawDist  ' measure echo pulse
  rawDist = rawDist * / Scale  ' convert to uS
  rawDist = rawDist / 2      ' remove return trip
  RETURN

```

Program Example: BASIC Stamp 1 Microcontroller

```
' =====
'
' File..... Ping_Demo.BS1
' Purpose.... Demo Code for Parallax PING))) Sonar Sensor
' Author..... Parallax, Inc.
' E-mail..... support@parallax.com
' Started....
' Updated.... 06 JUN 2006
'
' {$STAMP BS1}
' {$PBASIC 1.0}
' =====

' -----[ Program Description ]-----
'
' This program demonstrates the use of the Parallax PING))) sensor and then
' converting the raw measurement to English (inches) and Metric (cm) units.
'
' Sonar Math:
'
' At sea level sound travels through air at 1130 feet per second. This
' equates to 1 inch in 73.746 uS, or 1 cm in 29.034 uS).
'
' Since the PING))) sensor measures the time required for the sound wave to
' travel from the sensor and back. The result -- after conversion to
' microseconds for the BASIC Stamp module in use -- is divided by two to
' remove the return portion of the echo pulse. The final raw result is
' the duration from the front of the sensor to the target in microseconds.

' -----[ I/O Definitions ]-----

SYMBOL Ping          = 7

' -----[ Constants ]-----

SYMBOL Trigger       = 1           ' 10 uS trigger pulse
SYMBOL Scale         = 10          ' raw x 10.00 = uS

SYMBOL RawToIn       = 889         ' 1 / 73.746 (with **)
SYMBOL RawToCm       = 2257        ' 1 / 29.034 (with **)

SYMBOL IsHigh        = 1           ' for PULSOUT
SYMBOL IsLow         = 0

' -----[ Variables ]-----

SYMBOL rawDist       = W1           ' raw measurement
SYMBOL inches        = W2
SYMBOL cm            = W3
```

```

' -----[ Program Code ]-----
Main:
  GOSUB Get_Sonar          ' get sensor value
  inches = rawDist ** RawToIn  ' convert to inches
  cm = rawDist ** RawToCm     ' convert to centimeters

  DEBUG CLS               ' report
  DEBUG "Time (uS)..... ", #rawDist, CR
  DEBUG "Inches..... ", #inches, CR
  DEBUG "Centimeters... ", #cm

  PAUSE 500
  GOTO Main

END

' -----[ Subroutines ]-----

' This subroutine triggers the PING))) sonar sensor and measures
' the echo pulse. The raw value from the sensor is converted to
' microseconds based on the Stamp module in use. This value is
' divided by two to remove the return trip -- the result value is
' the distance from the sensor to the target in microseconds.

Get_Sonar:
  LOW Ping                ' make trigger 0-1-0
  PULSOUT Ping, Trigger   ' activate sensor
  PULSIN Ping, IsHigh, rawDist ' measure echo pulse
  rawDist = rawDist * Scale ' convert to uS
  rawDist = rawDist / 2    ' remove return trip
  RETURN

```

Program Example: Javelin Stamp Microcontroller

This class file implements several methods for using the PING))) sensor:

```
package stamp.peripheral.sensor;

import stamp.core.*;

/**
 * This class provides an interface to the Parallax PING))) ultrasonic
 * range finder module.
 * <p>
 * <i>Usage:</i><br>
 * <code>
 *   Ping range = new Ping(CPU.pin0);           // trigger and echo on P0
 * </code>
 * <p>
 * Detailed documentation for the PING))) Sensor can be found at: <br>
 * http://www.parallax.com/detail.asp?product\_id=28015
 * <p>
 *
 * @version 1.0 03 FEB 2005
 */
public final class Ping {

    private int ioPin;

    /**
     * Creates PING))) range finder object
     *
     * @param ioPin PING))) trigger and echo return pin
     */
    public Ping (int ioPin) {
        this.ioPin = ioPin;
    }

    /**
     * Returns raw distance value from the PING))) sensor.
     *
     * @return Raw distance value from PING)))
     */
    public int getRaw() {

        int echoRaw = 0;

        CPU.writePin(ioPin, false);           // setup for high-going pulse
        CPU.pulseOut(1, ioPin);               // send trigger pulse
        echoRaw = CPU.pulseIn(2171, ioPin, true); // measure echo return

        // return echo pulse if in range; zero if out-of-range
        return (echoRaw < 2131) ? echoRaw : 0;
    }
}
```

```

/*
 * The PING))) returns a pulse width of 73.746 uS per inch. Since the
 * Javelin pulseIn() round-trip echo time is in 8.68 uS units, this is the
 * same as a one-way trip in 4.34 uS units. Dividing 73.746 by 4.34 we
 * get a time-per-inch conversion factor of 16.9922 (x 0.058851).
 *
 * Values to derive conversion factors are selected to prevent roll-over
 * past the 15-bit positive values of Javelin Stamp integers.
 */

/**
 * @return PING))) distance value in inches
 */
public int getIn() {
    return (getRaw() * 3 / 51);          // raw * 0.058824
}

/**
 * @return PING))) distance value in tenths of inches
 */
public int getIn10() {
    return (getRaw() * 3 / 5);          // raw / 1.6667
}

/*
 * The PING))) returns a pulse width of 29.033 uS per centimeter. As the
 * Javelin pulseIn() round-trip echo time is in 8.68 uS units, this is the
 * same as a one-way trip in 4.34 uS units. Dividing 29.033 by 4.34 we
 * get a time-per-centimeter conversion factor of 6.6896.
 *
 * Values to derive conversion factors are selected to prevent roll-over
 * past the 15-bit positive values of Javelin Stamp integers.
 */

/**
 * @return PING))) distance value in centimeters
 */
public int getCm() {
    return (getRaw() * 3 / 20);          // raw / 6.6667
}

/**
 * @return PING))) distance value in millimeters
 */
public int getMm() {
    return (getRaw() * 3 / 2);          // raw / 0.6667
}
}

```

This simple demo illustrates the use of the PING))) ultrasonic range finder class with the Javelin Stamp:

```

import stamp.core.*;
import stamp.peripheral.sensor.Ping;

public class testPing {

    public static final char HOME = 0x01;

    public static void main() {

        Ping range = new Ping(CPU.pin0);
        StringBuffer msg = new StringBuffer();

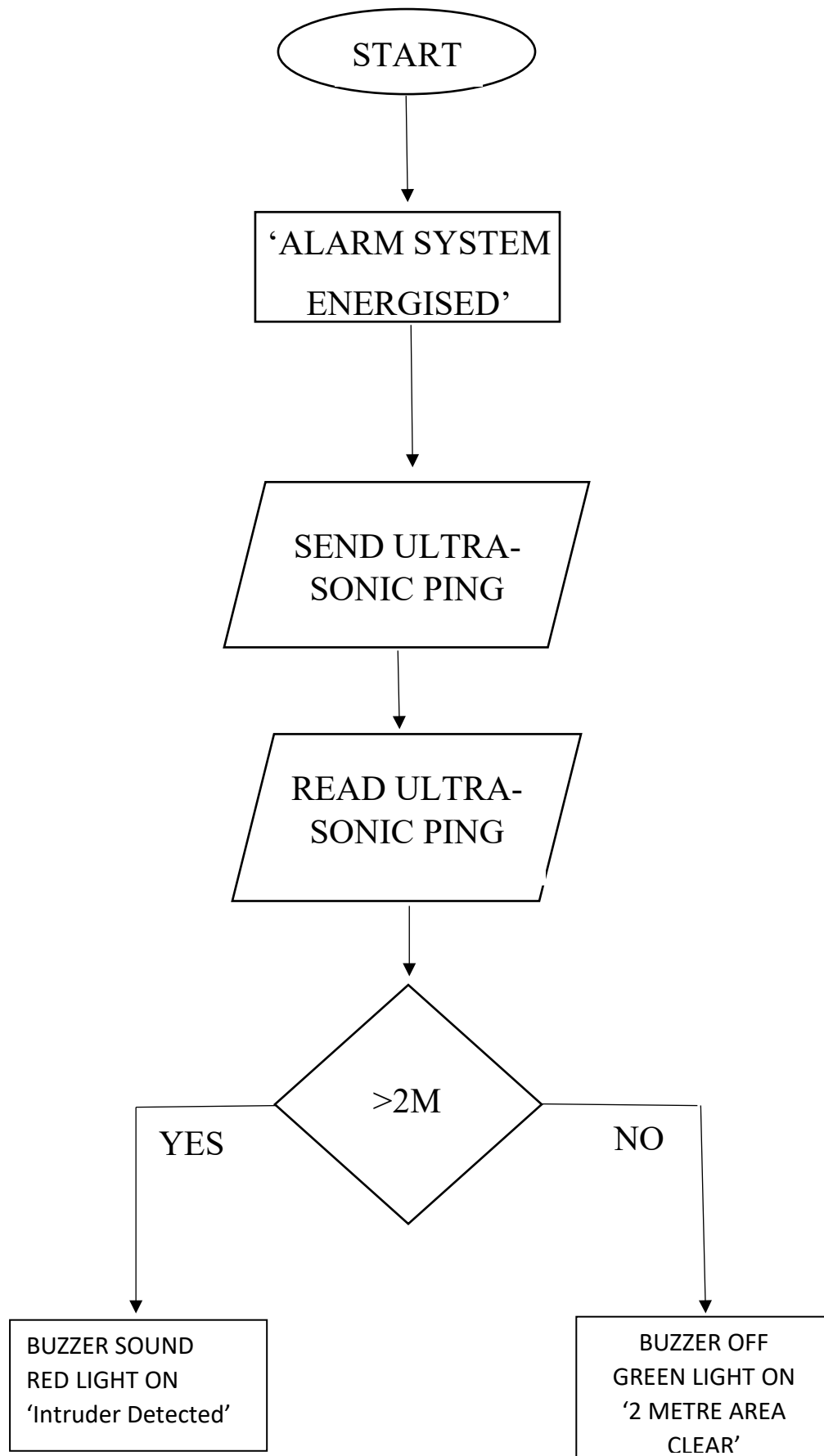
        int distance;

        while (true) {
            // measure distance to target in inches
            distance = range.getIn();

            // create and display measurement message
            msg.clear();
            msg.append(HOME);
            msg.append(distance);
            msg.append(" \"    \"\n");
            System.out.print(msg.toString());

            // wait 0.5 seconds between readings
            CPU.delay(5000);
        }
    }
}

```



Test Results sheet

	YES	NO	NOTES
Scope read and understood	Y		Proximity sensor
Hardware selected	Y		Compatible with Arduino
Software selected	Y		C+ Coding via Tinkercad
Buzzers function	Y		tone (BUZZER, NOTE_C4); //BUZZER SOUNDS delay (1000); tone (BUZZER, NOTE_A3); delay(1000);
LED functions	Y		
Resistors ok	Y		
Arduino Uno R3 ok	Y		
PING sensor ok	Y		
LCD ok	Y		
LCD Greeting message	Y		
No alarm when >2m's	Y		Green LED illuminated
Alarm when <2m's	Y		Buzzer and Red LED
LCD Screen changes accordingly	Y		
System auto-resets	Y		
Code error free	Y		

STUDENT FEEDBACK FORM

This section details the *extent* to which the assessment criteria are demonstrated by you, which in turn determines your mark. The marks available for each category of skill are shown. Lecturers will use the space provided to comment on the achievement of the task(s), including those areas in which you have performed well and areas that would benefit from development/improvement.

Generic Assessment Criteria	Marks available	Marks awarded
1. Engagement with Literature Skills		
	0	
2. Knowledge and Understanding Skills		
Hardware design 11/ 15 Excellent knowledge demonstrated of the hardware used with clear coherent reasons given for choices made. Software design 8/15 Flowchart uses correct symbols and is logically correct, but does not cover any setup functionality or imply a loop. It oversimplifies the code required for the system. You have really explored how the software operates the hardware, and the flowchart gives an overview of some of the logical steps of the program. However the software design should take place prior to the development of the code, your assignment implies this is written after the code was developed, citing lines where if /else statements occur. Consideration of some of the constants and variables required as part of this system would have improved your mark in this section.	30	19
3. Cognitive and Intellectual Skills		
Discussion, evaluation and conclusion Good evaluation of the underlying concepts. Some valid suggestions are made for improvement, and you have considered the effectiveness of your system against the original assignment brief. A consideration of the usefulness of the C programming language when working with embedded systems would have improved your mark further in this section, although you mentioned the usefulness of functions, these are not unique to the C programming language.	15	8
4. Practical Application Skills		
Implementation 24/40 Code is well written with self-annotating variables, use of constants where appropriate and comments. The hardware is utilised in a logical manner to give a functional system. Further marks in this section would have been gained by the use of functions, for example a function to turn the time taken for the ping to return into a distance.	50	30

testing 6/10 The inclusion of a test just outside the 2m range on the boundary, and just on the 2m would have been a useful addition here. Also, what would happen if the alarm is triggered at the 2m and then the intruder moves out of range? This should also be demonstrated. The testing table is a useful way to show what areas are being tested in a more compact fashion, and you have covered the majority of the tests required to show the effectiveness of the system.		
5. Transferable Skills for Life and Professional Practice		
Structure of report- Work is coherent, very fluent and is presented professionally. It includes datasheets for the components in the appendices. Code layout- code is indented on the most part correctly and makes good use of self-annotation techniques. In future ensure any images used are clearly referenced adhering to a standard referencing convention.	5	4

Assessment Mark (Assessment marks are subject to ratification at the Exam Board. These comments and marks are to give feedback on module work and are for guidance only until they are confirmed.)	Late Submission Penalties (tick if appropriate)		61%
	Up to 1 week late (40% Max)		
	Over 1 week late (0%)		