ET-REMOTE DISTANCE

ET-REMOTE DISTANCE is Distance Measurement Module by Ultrasonic Waves; it consists of 2 important parts. Firstly, it is the part of Board Ultrasonic (HC-SR04) that includes sender and receiver of Signal Ultrasonic. When the sender sends out Signal Ultrasonic in the air and the Signal touched any object, Signal Ultrasonic is reflected and sent back to the receiver; next, the Receiver converts the received Signal into the format of Width Pulse (positive). Secondly, it is the part of Board Controller that controls the interval of reading Signal Width Pulse from Board Ultrasonic and it converts the received Signal Pulse into the value of distance (centimeter) and shows the result on 7-Segment Display. Moreover, it can be interfaced with user through RS232; so, user can send Command to this Board Controller to read the distance measurement, including setup arbitrary distance for measuring and send out the distance measurement to user in the format of Logic TTL.

1. SPECIFICATION OF MODULE ET-REMOTE DISTANCE

- Use +5V Power Supply

- Distance measurement is in the range of 3 cm. – 210 cm. approximately

- Show measuring result by 7-Segment as 4-Digit; 3-Digit for integer and 1-Digit for decimal system

- A unit of measurement is centimeter with 1-Digit decimal system for display the measuring result

- ON/OFF 7-Segment Display by either SW. on board or by Command through RS232. The latest status of ON/OFF 7-Segment will be saved in E2PROM and it still remained although the Module Power-OFF.

- There are 2 Operation Modes. Firstly, this mode detects distance; it can setup any arbitrary distance to detect and it outputs the value as Logic TTL. Secondly, this mode reads the distance; it sends Command for reading distance through RS232. In this case, user can choose the format of reading distance either to be ASCII or HEX; either to be integer or decimal (depending on Command type for reading distance).

- Both Operation Modes auto operate together; so, user can connect Cable from any Port that is related to the Operation Mode instantly.

- For Distance Detection Mode, if the measuring distance is less than the defined Distance Detection, the Logic at Port OUTPUT becomes Logic 0 (TTL) and the Buzzer create long Beep sound. The interval of Beep sound is frequently when value of the measured distance (comes near and nearer the object) is less than the defined Distance Detection higher than 50%.

- ON/OFF Buzzer by setting JUMPER

- Use SW. on Module to set 1 value of distance each time for detection. This setting value will be stored in E2P and it still remains although Module Power-OFF. The old setting value of distance is overwritten every time user sets any new value of distance.

- Can set 2 Baud Rates by Jumper to send Command to read distance through RS232; 9600 and 57600.

- ON/OFF Filter (Default) for reading distance in order to display result on 7-Segment Display. If ON Filter, the 7-Segment Display is stable while showing the measured distance; but speed of reading value to show on Display is slower. If the range of distance for detection is highly different, it should set OFF Filter because speed of reading distance is more rapid; it is suitable for distance detection but the result that is shown on 7-Segment Display is unstable.

2. OPERATION OF MODULE ET-REMOTE DISTANCE

Initially, Module starts running when internal program reads value of Jumper BR to setup Baud Rate for RS232 Communication between user and Module; there are 2 Baud Rates that can be setup; 9600 and 57600. Secondly, read value of distance Detector, status of Filter, and status of ON/OFF 7-Segment that user has already setup and stored in E2PROM. Thirdly, check status of pressing SW.SAVE; pressing this SW.SAVE at the beginning while Power-ON, it setup the value of ON/OFF Filter (Normally, Default value is Filter = OFF and 7-Segment = ON). Finally, type the message Ascii "*@OK*" and followed by *0x0D*, it is totally 4 Byte and it is sent out through RS232; it means that Module is ready to run. The value that is sent out can check if the Module is ready to run when Start or Power-ON; or, it may use Delay instead because it makes the operation of MCU and Module run smoothly.

Next, program loops to read Pulse value from Board Ultrasonic in order to calculate the received value to be distance and show the result on 7-Segment Display (when 7-Segent Display is setup as ON). Meanwhile, it compares the value of distance that is read with the value from Detector; if the reading value is less than the defined value from Detector, the program externally sends Logic 0 to Connector Output (default = 1) and it creates Beep sound. This Logic 0 and Beep sound continues creating until the current value that is reading is higher than the value from Detector. While reading and showing value of distance, including checking distance Detector, Module still waits for receiving Command from user to read the distance through RS232 automatically. The operation of Module continues running repeatedly.

The measuring distance of Module is in the range of 3 cm – 210 cm; the measuring distance between object and Module is shown in the picture below. For highly accurate measurement, the Module should be parallel to the floor and be upright to the object that will be shot by Ultrasonic because the reflection of wave is also high accuracy.
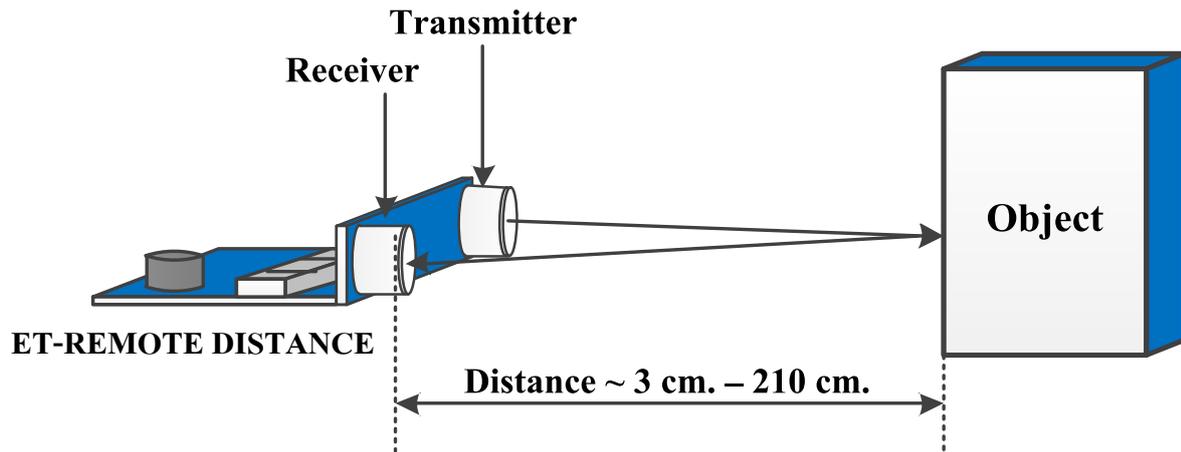
Figure 2.1 shows distance measurement of Module ET-REMOTE DISTANCE.

__NOTE__! For highly accurate measurement when measuring distance by this Module, the Module should be parallel to the floor and be upright to the object that will be shot by Ultrasonic because the reflection of wave is also high accuracy. Moreover, the distance between the Module and floor should be suitable; if the measurement is wider, the height of the Module from the floor should be higher because Ultrasonic wave that is shot is like a ray of a flashlight. If the distance between the Module and floor is too close, the ray that is shot will reflect the floor before reflect the defined object; so, the value of measurement is inaccurate.
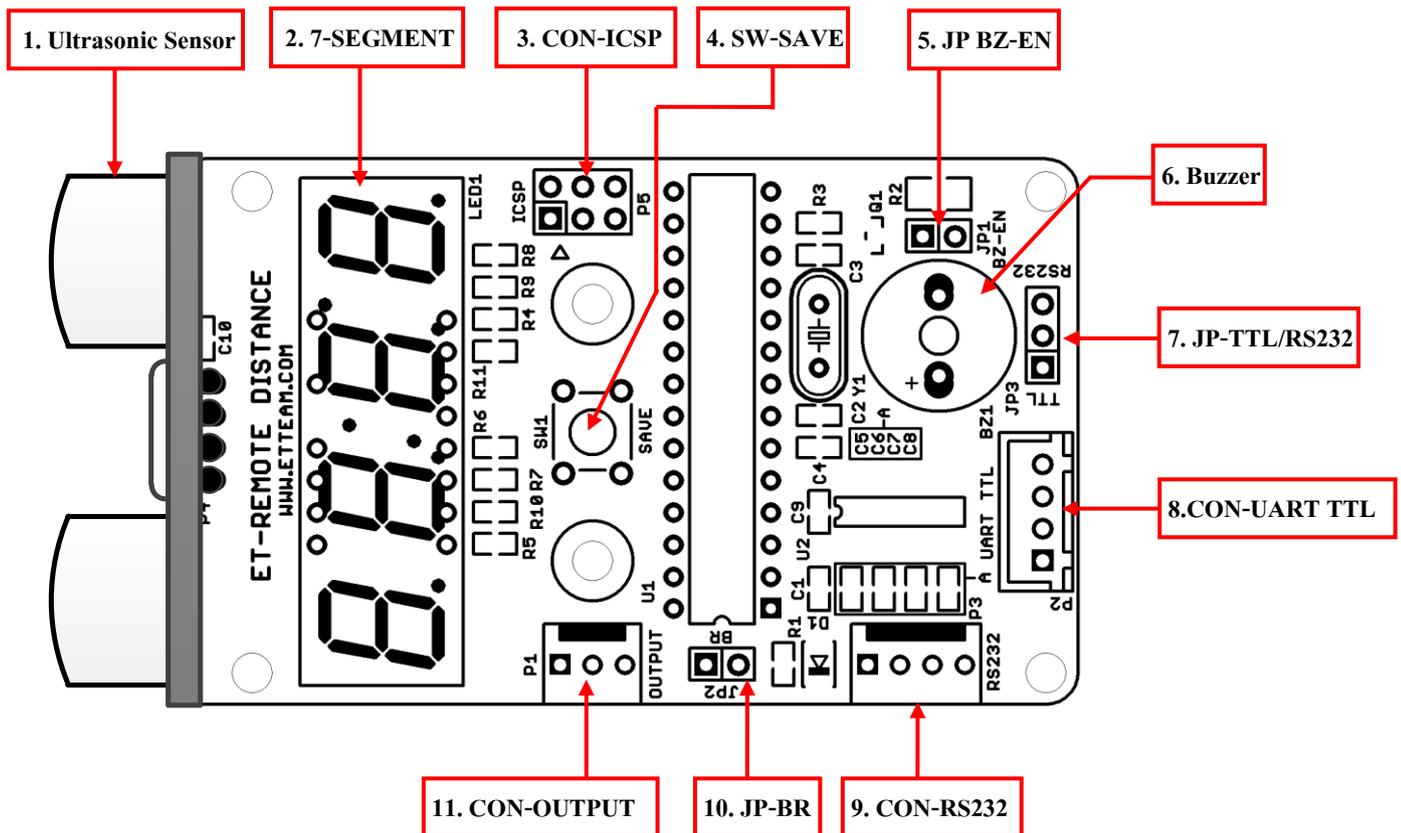
3. STRUCTURE OF MODULEET-REMOTE DISTANCE

Figure 3.1 shows structure of Module ET-REMOTE DISTANCE.

1. Ultrasonic Sensor: This is Board Ultrasonic Sensor to measure distance in the range of 3cm-210cm.

2. 7-Segment: This is 4-Digit Display to show the measuring value with the unit of measurement of *centimeter*; 3-Digit for numeric value and 1-Digit for decimal value. It can enable/disable (ON/OFF) 7-Segment Display by Command through RS232 or press and hold SW.SAVE (No.4) for a while until 7-Segment becomes OFF or ON as required (Note: The operation of ON/OFF 8-Segment is as Toggle. The status of ON/OFF will be stored in E2PROM and it will be called when Power-ON).

3. CON-ICSP: This Connector downloads Firmware into MCU; it is not related to any use of user.

4. SW-SAVE: There are 3 functions of this Switch as follows;

      1) ON/OFF Filter, please read more information in 5.1

      2) ON/OFF 7-Segment Display, please read more information in 5.2

      3) Save the preferable distance to detect and it is save in E2PROM to use in Detection Mode, please read more information in 4.1

5. JP BZ-EN: This Jumper sets ON/OFF Buzzer. If removed this Jumper, it is OFF-Buzzer; but, if it inserted this Jumper, it is ON-Buzzer.

6. Buzzer: This Buzzer creates Beep sound; it runs according to the setting value of Firmware.

7. *JP-TTL/ RS232*: This Jumper chooses Connector RS232. If this Jumper is set to the position of TTL, it chooses the Connector UART TTL (No.8); or, if this Jumper is set to the position of RS232, it chooses the Connector RS232 (No.9).

TTL ▬●● RS232   TTL ●●▬ RS232

เลือกใช้งานขั้วต่อ Uart TTL(8)   เลือกใช้งานขั้วต่อ RS232(9)

Figure 3.2 shows how to set Jumper when choosing Connector 232.

NOTE! When receiving-transmitting Command to read Distance, it has to choose either one of Connector RS232 only; and set Jumper JP-TTL/RS232 correctly. When choosing any Connector, it depends on type of Board MCU that is connected in order to send Command and read Distance. If the connective Board has already been connected IC Line Driver 3232 or user requires connecting to Port 232 of PC, it should choose Connector RS232 (9); but, if user requires connecting it with Pin UART of MCU directly, it should choose Connector UART TTL (8).

8. *CON-UART TTL*: This Connector RS232 is TTL Level that can be connected with Pin UART (RX,TX) of MCU directly. While connecting, it has to cross the Cables between RX, TX. The operation of this Connector is the same as Connector RS232 (No.9), but the level of signal is less, so it can be connected with Pin MCU directly.
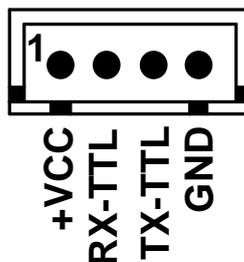

Figure 3.3 shows Connector UART TTL to send Command and read Distance as ASCII or Hex CODE.

Details of each PIN

+VCC,GND = Connect with Power Supply DC +5V, Ground

RX-TTL   = Receive user's Command ON/OFF 7-Segment and Command for reading Distance

TX-TTL   = Send value of distance measurement in the format of ASCII or HEX to user

When using this Connector UART TLL, it has to set Jumper TTL/RS232(7) to the position of TTL.

9. CON- RS232: This Connector RS232 has Line Driver completely; it is externally connected through IC Line Driver 3232 on board to this Connector. User can connect this Connector with PORT RS232 of PC or connects with PORT RS232 of External MCU that has already been connected through IC Line Driver RS232. Remember, while connecting this Connector, it has to cross Cables RX, TX. The operation of this Connector is the same as Connector UART TTL (No.8) but the level of signal is different.
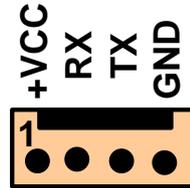

Figure 3.4 shows Connector RS232 Line Driver to send Command and read Distance as ASCII or Hex CODE.

Details of each PIN

+VCC , GND = Connect Module with Power Supply DC +5V, Ground

RX = Receive user's Command ON/OFF 7-Segment and Command for reading distance

TX = Send value of distance measurement in the format of ASCII or HEX to user

When using this Connector RS232, it has to set Jumper TTL/RS232 (7) to the position of RS232.

10. JP-BR: This Jumper chooses Baud Rate for RS232 Communication when using Connector UART TTL or RS232. There are 2 values for Baud Rate as follows;   Insert Jumper   = Choose Baud Rate 57600 bit/sec

Remove Jumper = Choose Baud Rate 9600 bit/sec

After switched the position of Jumper completely, it has to reset Module to update the new Baud Rate.

11. CON-OUTPUT: This Connector sends Signal of distance detection as TTL to user. If the value of distance that is currently reading is higher than the defined value of Detection, the status at Pin P1 of Connector OUTPUT becomes Logic '1' (5V). The status still is Logic '1' until the value of the distance that is currently reading is less than the defined value of Detection and the status at Pin P1 of Connector OUTPUT becomes Logic '0' (0V). It alternates between the status of Logic '0' (0V) and Logic '1' (5V) repeatedly. Before using this Connector, user has to set the required range of distance to detect (Distance Detection) for Module first, please read more information in 4.1.
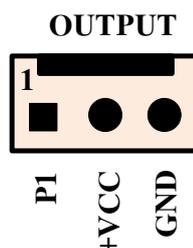

Figure 3.6 shows PIN arrangement of Connector OUTPUT.

> *Note! For the part of Power Supply of Module, user has to choose either one of Connector UART TTL (8) or RS232 (9) or OUTPUT (11) only to connect at PIN +VCC, GND.*

4. HOW TO USE MODULEET-REMOTE DISTANCE

There are 2 operation modes for this Module ET-REMOTE DISTANCE and both operation modes auto operate together; in this case, user only chooses Cable to interface to Connector of the preferable operation mode. Now, it describes how to use each operation mode as follows;

The operation of each mode is mentioned below;

4.1) DETECTION MODE

This operation mode is to read any incoming value of distance by Module to check if it is less or higher than the defined value of distance in E2PROM. If the reading value is higher than the defined value of user, it sends the Logic '1' to Connector OUTPUT Pin P1. Or, if the reading value is less than the defined value of user, it sends out Logic '0' to Connector OUTPUT Pin P1. In this case, user only reads Logic at Connector OUTPUT P1.
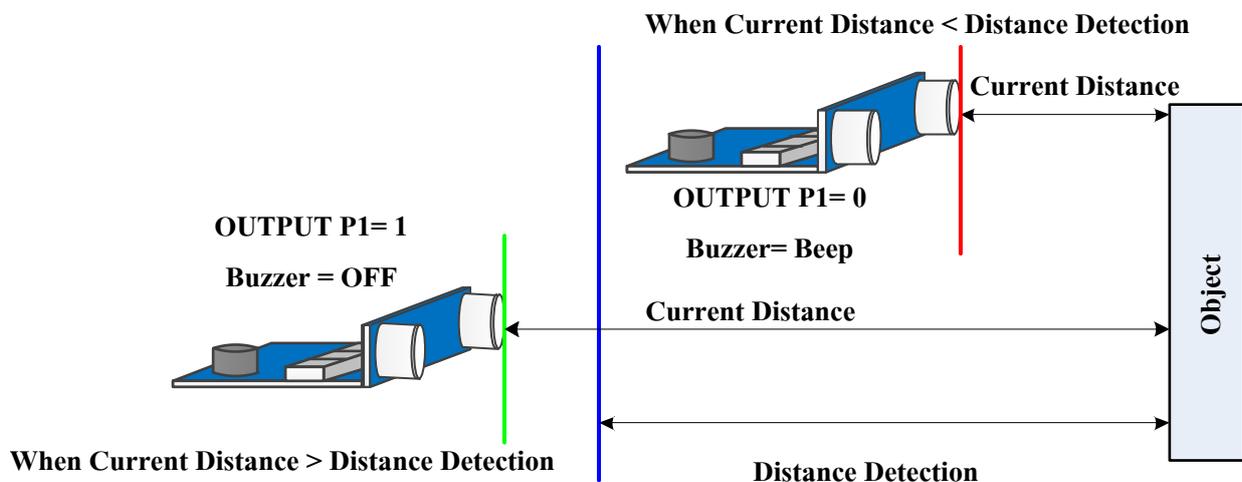

When Current Distance < Distance Detection

Current Distance

OUTPUT P1= 0

Buzzer= Beep

Current Distance

OUTPUT P1= 1

Buzzer = OFF

When Current Distance > Distance Detection

Distance Detection

Object

Figure 4.1 shows status of OUTPUT P1 when the current distance is higher or less than the defined Detection of user.

Summarize how to use Detection Mode

1.) Provide Power Supply +5V, GND into Module through either one of Connectors as mentioned above; in this case, it suggests providing Power Supply at Connector OUTPUT (11). For this operation mode, it uses Pin P1 for reading Signal Detection; so, it is more convenient to use. After provided Power Supply into the Module completely, 7-Segment should be enabled (ON). If 7-Segment is disabled (OFF), please check if 7-Segment is turned off. For more information about ON/OFF 7-Segment, please read the topic 5.2.

2.) Setup arbitrary distance to detect (Distance Detection) as mentioned below;

- After 7-Sement ON completely, turn the front of Module towards the upright object or wall as shown in the picture 4.1.

- Slightly move Module in-out from the wall until it reaches the defined distance to detect; user can see the distance measurement on 7-Segment Display with the measurement unit of centimeter. If 7-Segment Display shows the message "Err.", it means that the distance measurement exceeds the defined range that the Module can measure (3-210 cm.).

- After got the preferable Distance Detection completely, it should press SW.SAVE on Module and release the switch within 1 second, value of the Distance Detection will be saved in E2PROM. If pressed SW.SAVE while 7-Segment Display is showing the message "Err", there no any effect the operation; user should moves Module and adjust the distance to save the new distance detection again. The value of the distance that is saved is the value of Distance Detection; Program compares this saved value with the latest value of distance that is read. After released SW.SAVE, user can hear long Beep sound twice; it means that the value of the Distance Detection has already been saved. User can hear Beep sound periodically because the Module is in the defined range of Distance Detection that the Module can detect. If user moves the Module far from the defined range of the Distance Detection, the Beep sound stops.

> **Note**_! The Distance Detection that is setup by user will be saved in E2PROM and it will always be called when Module Power-ON. If user requires changing new value of Distance Detection, please do the same process as mentioned above again. Remember, the new value of the Distance Detection that is setup always overwrites the old one. If the new value of the Distance Detection that is setup is less than the old one, user will hear Beep sound all the time while setting new value but there is no any effect on setting the new value; user can ignore it and keep setting the new value as normal._

3.) After setup the new Distance Detection completely, it has to connect Cable Pin1 at Connector OUTPUT to MCU to read status of Detection as shown in the figure 4.2 .If the current value of distance that is reading is higher than the defined value of Distance Detection, it means that the Module exceeds the defined range of detection; so, the Module sends out Logic '1' to user through Connector OUTPUT Pin P1. The Logic '1'still remained until the current value that is reading is less than the defined value of the Distance Detection. When the Module is in the defined range of distance detection, the Module sends out Logic '0' to user through Connector OUTPUT Pin P1 and the Logic '0' still remained. Moreover, user can hear Beep sound periodically when the Module is still in the defined range of detection. If the Module is in the range of detection less than 50% of the Distance Detection, user can hear Beep sound often.
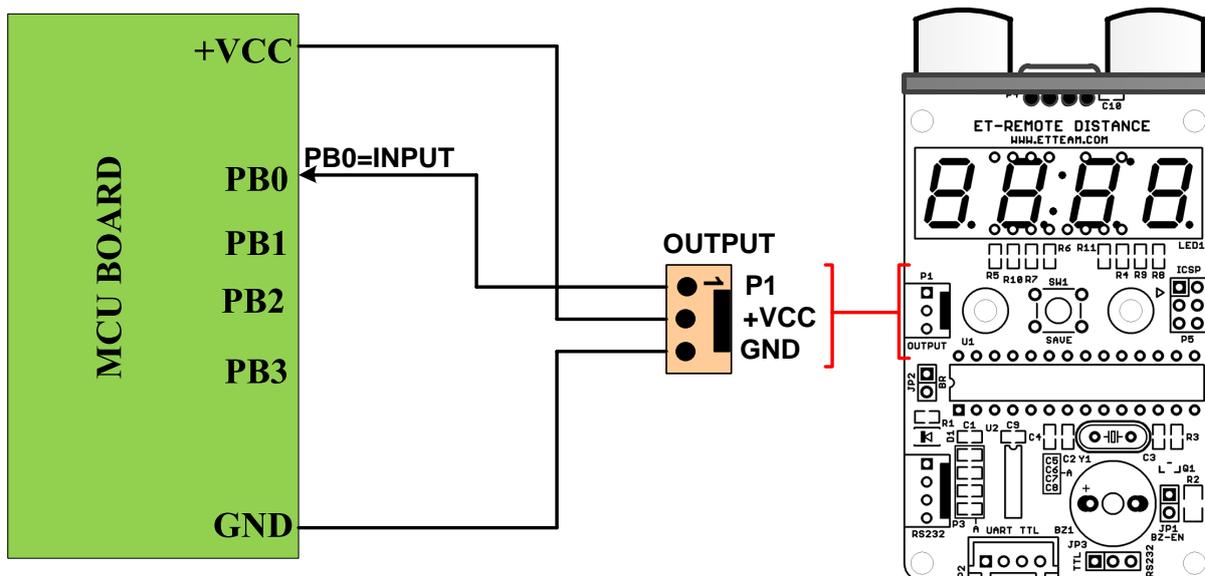
Figure 4.2 shows how to connect Cable of Detection Mode for reading status of Detection.

**Note**! While detecting distance by the Detection Mode and it shows the message "Err." on 7-Segment Display, it means that it exceeds the defined range of distance that the Module can detect (3-210 cm.). The Module assumes that the value of "Err" is always higher than the defined value of Detection by user; so, it is always Logic '1' at OUTPUT P1.

4.2) READ DISTANCE MODE

When using this operation mode, it sends Command to read distance measurement by the Module. User can send Command and read the distance measurement that is sent back from either Connector "RS232 (9) or UART TTL (8)". It sets Jumper "JP-BR" (10) to choose Baud Rates for the Communication; in this case, there are 2 values that are 9600 bit/s or 57600 bit/s.

Summarize how to use Read Distance Mode

1.) Set Jumper "JP-BR" (10) to choose Baud Rate for receiving-sending data as follows;

When setting Jumper   = 57600 (Default); or,

When removing Jumper   = 9600.

The Baud Rate that is chosen by user will be updated when Module Power-ON.

2.) Set Jumper "TTL/RS232" (7) to choose Connector 232. If interfacing 232 to Pin MCU directly (TTL), it has to set Jumper to the position of TTL and then interface Connector "UART TTL" (8). Or, if interfacing 232 to PORT 232 of PC or PORT 232 of Board MCU that has IC Line Driver232 in the part of Pin Rx,Tx of MCU, it has to set Jumper to the position of RS232 and then interface Connector "RS232" (9).

3.) Interface Cable at Connector "RS232" (9) or "UART TTL" (8) according to setting Jumper in step 2; next, provide

Power Supply into Module (Power-On Module) as shown in the picture 4.3 below;
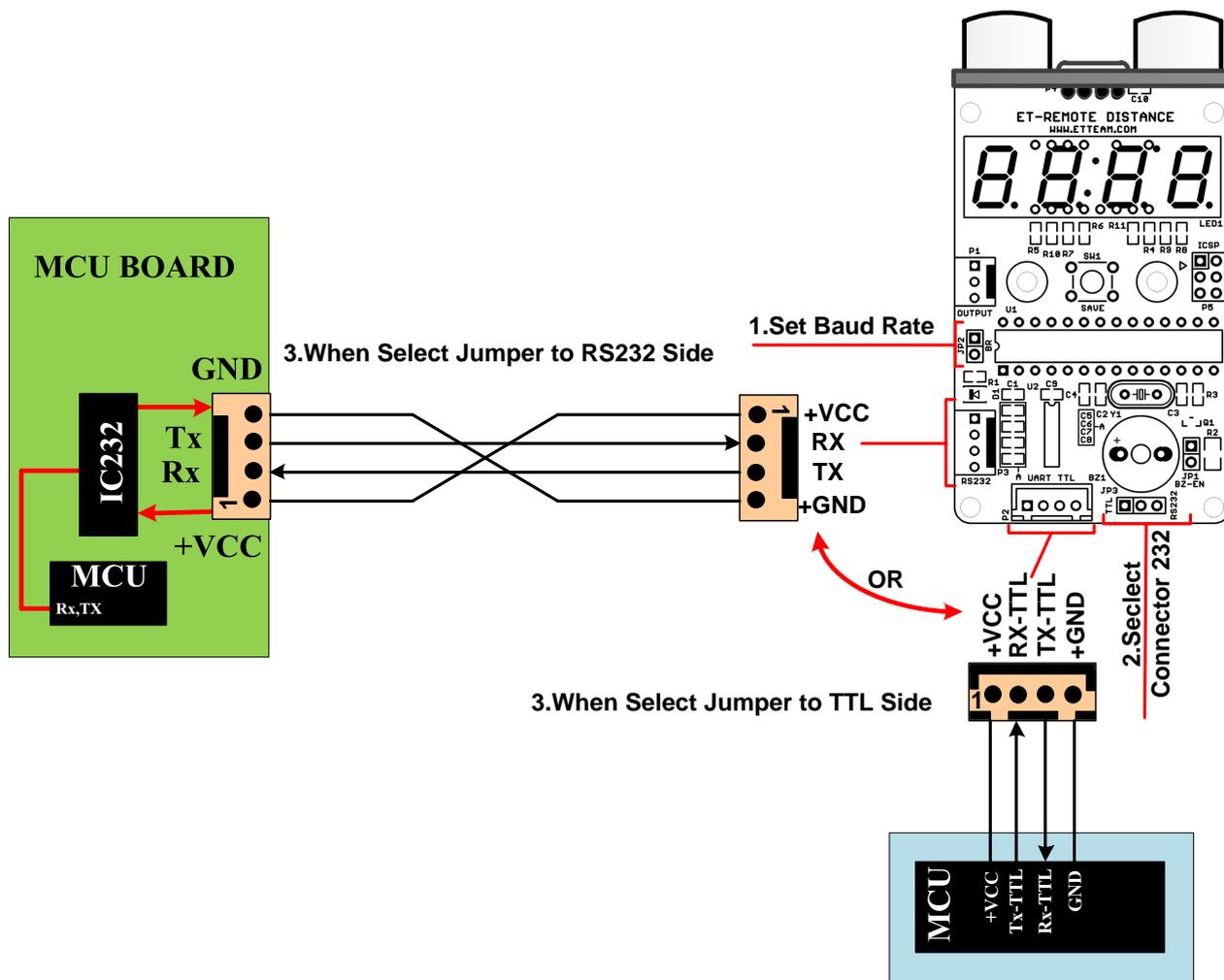

Figure 4.3 shows how to connect Cable for Read Distance Mode.

4.) Write Program to send Command in the format of ASCII Code for reading the distance measurement as required.

Each Command has 5 Byte; the last Byte of Command is Enter that has no any symbol, so it must be replaced by

ASCII Code = 0x0D. The value of distance that is sent back to user is in the format of ASCII Code and Hex Code,

depending on the Command type that user has sent out to read. The format of Command is mentioned as follows;

- 1. Command 'RA' (Read-ASCII Integer)

This Command reads the distance measurement that has the result as integer. The Module returns 5

Byte Data in the format of ASCII Code to user; the value of distance measurement is in the 2nd - 4th Byte (ASCII

3Byte), the value is in the range of "003-210" with the measurement unit of centimeter. If the value of distance

measurement is not in the range, the Data in the 2nd – 4th Byte that is sent back from the Module is "Err". The

format of Command is;

Start	Command	Mark#1	END
byte1	byte2-3	byte4	byte5
∗	RA	=	Enter (0x0D)
Respond Distance			
Start	Distance = cm.(Ascii 3Byte)		END
byte1	byte2-4		byte5
#	003-210		0x0D
#	Err		0x0D

→ **When Distance Between 3-210 cm.**

→ **When Distance Not Between 3-210 cm.**

This Command is suitable for reading the distance measurement as integer to show the result on LCD Display because the result is in the ASCII Code.

- 2. Command 'RH' (Read-Hex Integer)

This Command reads the distance measurement that has the result as integer. The Module returns 3 Byte Data in the format of Hex Code to user; the value of the distance measurement is in the 2nd Byte (Hex 1 Byte), the value is in the range of 0x03-0xD2 (3-210) with the measurement unit of centimeter. If the value of the distance measurement is not in the range, the Data in the 2nd Byte that is sent back from the Module is 0xFE that means the range of distance measurement is Error. The format of Command is;

Start	Command	Mark#1	END
byte1	byte2-3	byte4	byte5
∗	RH	=	Enter (0x0D)
Respond Distance			
Start	Distance = cm.(Hex 1Byte)		END
byte1	byte2		byte3
0x23 ('#')	0x03-0xD2		0x0D
0x23 ('#')	0xFE		0x0D

→ **When Distance Between 3-210 cm.**

→ **When Distance Not Between 3-210 cm.**

This Command is suitable for reading the distance measurement as integer for mathematical calculation or comparison. The value that is read is the actual value in the format of hexadecimal number; so, it is unnecessary to convert any value before use.

- 3. Command 'Ra' (Read-ASCII Integer and Point)

This Command reads the distance measurement as integer with 1 decimal point. The Module returns 7 Byte Data in the format of ASCII Code to user; the value of the distance measurement is in the 2nd – 6th Byte (ASCII 5 Byte), the value is in the range of "003.0-210.0" with the measurement unit of centimeter. If the value of the distance measurement is not in the range, the Data in the 2nd – 6th Byte that is sent back from the Module is "Error". The format of Command is mentioned below;

Start	Command	Mark#1	END
byte1	byte2-3	byte4	byte5
*	Ra	=	Enter (0x0D)

Respond Distance				
Start	**Distance = cm.(Ascii 5byte)**			**END**
byte1	byte2-4	byte5	byte6	byte7
#	003-210	.	0-9	0x0D
#	Error			0x0D

→ **When Distance Between 3-210 cm.**
→ **When Distance Not Between 3-210 cm.**

-Respond : byte2-4 = Integral value of Distance in the format of Ascii Code

byte6 = 1 Decimal point of Distance in the format of Ascii Code

This Command is suitable for reading the distance measurement that has decimal point to show on LCD Display because the result is in the format of ASCII.

- 4. Command 'Rh' (Read-Hex Integer and Point)

Start	Command	Mark#1	END
byte1	byte2-3	byte4	byte5
*	Rh	=	Enter (0x0D)

Respond Distance			
Start	**Distance = cm.(Hex 2byte)**		**END**
byte1	byte2	byte3	byte4
#	0x03-0xD2	0x00-0x09	0x0D
#	0xFE	0xFE	0x0D

→ **When Distance Between 3-210 cm.**
→ **When Distance Not Between 3-210 cm.**

-Respond : byte2 = Integral value of Distance in the format of Hex Code (3-210)

Byte3 = 1 Decimal point of Distance in the format of Hex Code (0-9)

This Command reads the distance measurement as integer with 1 decimal point. The Module returns 4 Byte Data in the format of Hex Code to user; the value of distance measurement is in the 2nd – 3rd Byte (Hex 2

Byte). In this case, the Byte 2 is integral value that is in the range of 0x03-0xD2; and the Byte 3 is the value of the decimal point that is in the range of 0x00-0x09 with the measurement unit of centimeter. If the value of the distance measurement is not in the range of 0x03-0xD3 (3-210), the Data in both Byte 2 and the Byte 3 that is sent back from the Module is 0xFE; it means that the distance measurement is Error. The format of Command is mentioned below;

This Command is suitable for reading the distance measurement that has decimal point for mathematical calculation or comparison and the value that is read is the actual value in the format of hexadecimal number. Both 2 Bytes of integral number and decimal number that are read are in separate part, so it has to combine both 2 Bytes together and store in the float Variable, and finally, user will get the value of the distance measurement that has decimal point before calculation.

For example: If the value of distance measurement that is read is 0x2A (integral number = 42) and 0x05 (decimal point = 5), the result is 42.5 cm when combined both parts together in the format of decimal number. The way to write program for combining the value is illustrated below;

```
Main()
{
    float  ds  ;  //Declare float variable to store distance measurement with
                    decimal point

    ds = 0x2A +(float)0x05/10 ;  //Include integral and decimal point
}
```

Sum of inclusion = Integral distance measurement + (Distance with decimal point/10)

If the format of Command (5Byte) that is sent out each time is incorrect (although only one byte is incorrect), the Command is not processed and the Module responds to user with the message " *#Command Fail !*" and followed by 0x0D.

Note! When sending Command to read value of distance measurement each time, user has to receive respond of the Command that is sent back from the Module first by checking the value of 0x0D completely and then send the next command. Otherwise, any new Command that is sent out or any data that is return to user may be incorrect because MCU is not ready to receive any Command and it has not yet finished the old Command.

In the initial stage of operation, it should provide Power Supply to Module and Board MCU; user has to wait until the Module is ready to receive Command first and user can send Command to read value. In this case, user may write proper Delay at the header of program or check the respond that is sent back by the Module. After Power-On and the Module is ready to receive Command, it sends ASCII "@OK" and followed by 0x0D (4byte) to user. User can read this value at the beginning of program to check if the Module is ready to receive Command.

5. HOW TO SETUP OPTIONS FOR MODULE ET-REMOTE DISTANCE

There are 2 values that can be setup for this Module; ON/OFF Filter and ON/OFF 7-Segment. Any ON/OFF status of each Option that user setup will be stored in E2PROM; so, every time Module is Power-On, this Option will always be called. The way to setup Option is mentioned below;

5.1) *Setup ON/OFF Filter*: This ON/OFF Filter has an effect on the operation of Module and the display of 7-Segment as described below;

- **ON-Filter**: The good point of ON-Filter is the stability of numbers that are shown on 7-Segment of Module. However, the weak point of ON-Filter is too slow to respond to the display of 7-Segment if the value of the distance measurement that changes from one value to another value is wide; moreover, it is also slow to read the distance measurement for Detection and it is too slow to send the value of distance measurement to user. So, this ON-Filter is suitable for any use that emphasizes stability of the display on 7-Segment, but it is not suitable for Detection Mode or Read Distance Mode through RS232 because it cannot respond the operation in time.

- **OFF-Filter**: The good point of OFF-Filter is quick to send the value of the distance measurement through RS232 and Signal Detection to user, but the weak point of this OFF-Filter is too difficult to see the value of the distance measurement that is shown on 7-Segment because the display is unstable. So, this OFF-Filter is suitable for Detection Mode and Read Detection Mode through RS232.

<u>How to setup ON/OFF Filter</u>

1) When it has not yet provided any Power Supply to Module, press SW.SAVE (4) on the Module and hold for a while

2) Connect Power Supply to the Module (SW.Save is still pressed); user can hear Beep sound once.

3) Look at the last digit of 7-Sgment (on the far right), check if it shows the letter 'd' or 'E'.

If 'E' = ON-Filter   or,   'd' = OFF-Filter

Figure 5.1A) ON-Filter   **Figure 5.1 B) OFF-Filter**

When setting ON/OFF Filter, the operation is in the format of Toggle. Every time user follows the step 1 and 2, the values are switched alternatively. The latest value that is shown to user is always stored in E2PROM and it will be

called when Module Power-ON.

4) Release SW.SAVE, the Module starts reading the distance measurement and display the value on 7-Segment as normal. When user requires changing or setting any new value, please return and follow the step 1 again.

5.2) *Setup ON/OFF 7-Segment*: There are 2 methods to setup ON/OFF 7-Segment as follows;

1)   Setup ON/OFF 7-Segment by Button Save on Module, please follows these steps;

- Provide Power Supply into Module; in this case, the status of 7-Segment may ON or OFF, it depends on the previous status of 7-Segment that is set by user.

- Press SW.SAVE on the Module and hold for a while until the status of 7-Segment becomes ON or OFF; user can hear Beep sound once when it set the status of ON/OFF for 7-Segment successfully.

- Release SW.SAVE and Module returns to read the distance measurement as normal.

- When pressing/releasing SW.SAVE each time, it switches ON/OFF of 7-Segment (Toggle); in this case, it has to press SW.SAVE and hold for a while until user can see the change on 7-Segemnt because it means that the setting value of ON/OFF successfully. If released the Switch too fast, it saves the value of Distance for Detector instead. This status of ON/OFF that is set is stored in E2PROM and it is always called when Module Power-ON.

2)   When setting ON/OFF 7-Segment by sending Command through RS232, it has to interface Cable 232, set Jumpers, and including choose the proper Baud Rate as mentioned in 4.2 first; next, user can send Command as required. The format of Command for setting ON/OFF of 7-Segment is described as follows;

- Command 'RL' : This Command is 5 Byte ASCII Command. When the Command is done, Module returns 4 Byte Respond to user that is "#OK" and then followed by 0x0D. The operation of this Command is in the format of Toggle ON/OFF of 7-Segment; when it sends out Command each time, it sets the status of 7-Segment either to be ON or OFF, depending on the current status of 7-Segment. If the current status of 7-Segment is ON, it becomes OFF instead; or, if the current status of 7-Segment is OFF, it becomes ON instead.

Start	Command	Mark#1	END
byte1	byte2-3	byte4	byte5
*	RL	=	Enter (0x0D)

Respond Command		
Start	Data	END
byte1	Byte2-3	Byte4
#	OK	0x0D

*- When sending out Command each time, it switches the status of ON/OFF of 7-Segment (Toggle).*When the status of 7-Segment is set to be ON/OFF by any way, the latest status that is set is always stored in E2PROM and it will be called when Module Power-ON.

> *<u>Note</u>! When Module Power-ON but user cannot see any change of distance measurement on 7-Segment, user may set OFF 7-Segment. In this case, user only switch the status to ON 7-Segment as mentioned above.*

6. EXAMPLE PROGRAMS

This example program illustrates how to use Command to read the distance measurement and read status of Detection. It is written by 3 series of MCU; if it is 89C51RE2, it uses Keil C51 Compiler; or, if it is AVR MEGA128, it uses C-WIN AVR; or, if it is PIC8722, it uses CCS Compiler. There are 4 example programs for each MCU as follows;

Ex1_CMM_RA_Ra: This example program illustrates how to send Command RA and Ra to read distance; the result is in the format of ASCII Code. It sets Baud Rate for sending the Command as 57600 bit/s. The value of the distance measurement that is read will be shown on LCD Display; the upper line shows integral (by Command RA) and the lower line shows the value with decimal point (by Command Ra) as shown in the picture 6.1 below.

Figure 6.1 shows the value of distance measurement on LCD Display.

From the example program above, after sent out the Command RA or Ra, user has to wait for receiving Data from Module completely according to the amount of Byte of each Command first. Next, user needs to take the received Data to check and setup condition of printing through the display.

Ex2_CMM_RH_Rh: This example program illustrates how to send Command RH and Rh to read distance; the result is in the format of Hex Code. It sets Baud Rate for sending the Command as 57600 bit/s. At the beginning of operation, it sends out the Command RH and it waits for receiving Data from the Module until amount of Byte of each Command has completed. Next, it takes the received distance that is read in the format of Hex Code to show on Port; in this case, user has to interface LED at Port to display the distance measurement in the format of Hex Code. After converted the Hex Code that is shown on LED Display to be decimal number completely, user will get the distance measurement (integral) according to the value that is shown on 7-Segment of Module. Next, Program sends out the Command Rh and it waits for receiving Data from the Module until amount of Byte of Command has completed. The received Data is the distance measurement in the format of Hex Code, it is 1-Byte integral and 1-Byte decimal point; Program combines both 2-Bytes together as mentioned above and the result will be in the same variable in the format of variable Float. Next, it compares the received value to check if the value that is read is in the defined range. If the reading value is in the range of 5.0-5.4cm, LED that is connected at Port Bit 0-3 turns on (ON); and if the reading value is in the range of 5.5-5.9cm, LED that is connected at Port Bit 4-7 turns on (ON); but, if the value exceeds this defined range, all LED turns off (OFF).

Ex3_CMM_RL_7Seg: This example program illustrates how to set ON/OFF for 7-Segment on Module by sending Command instead of pressing SW.SAVE on Module; it sets Baud Rate for sending Command as 57600 bit/sec. At the beginning of operation, it sends the Command RL to Module and it waits for receiving Respond from Module. When the Module returns the Respond 0x0D, it means that the Command that is sent out has already been done; next, it should set Delay for 2 seconds and user can send the old Command again. In this case, it makes 7-Segment on Module alternately turn on and turn off (Toggle) every 2 seconds consecutively.

Ex4_RdOut_Detec: This example program illustrates how to read status of Detection at Pin P1 of PORT OUTPUT. Before running program to test the operation, user has to set Detection value for Module completely. Referred to the example, it sets the Detection value as 10cm; interfaces LED to display the result of Detection at whichever Port of MCU; interface Pin P1 of Module with Pin I/O of MCU; and set the value as Input. When program starts running, it reads Logic at Pin P1 and check the status. If the reading value is 1, it means that the distance measurement is higher than 10cm. (normal); so, the Program commands LED that is interfaced at Port O/P of MCU to turn off (OFF). Or, if the reading value is 0, it means that the distance measurement is less than 10cm; so, the Program commands LED that is interfaced at Port O/P of MCU to turn on (ON). User slightly
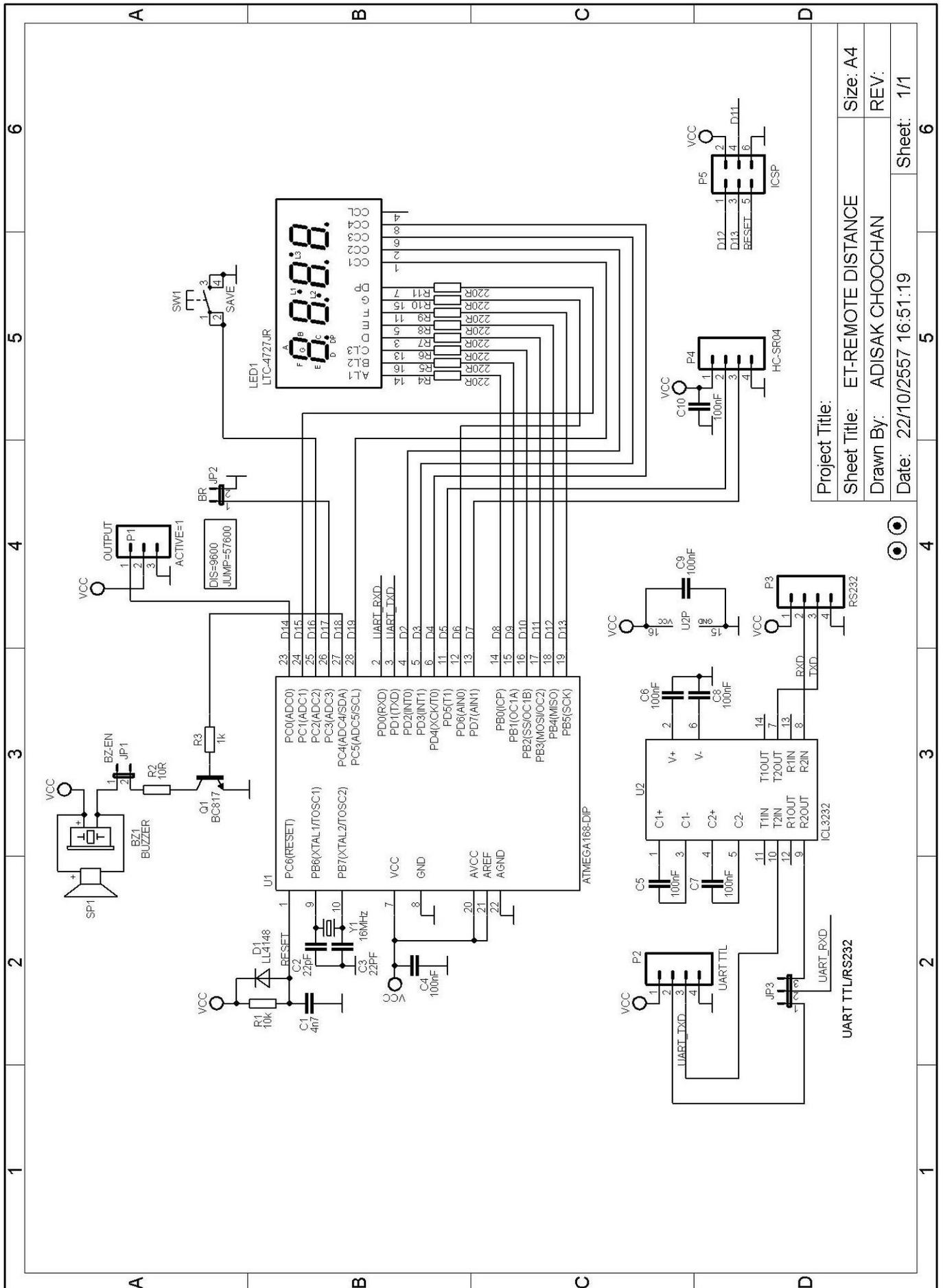
moves the Module until user gets the distance that is less and higher than 10cm, user can see LED ON/OFF according to the defined conditions.

For connective circuit for testing program, please read Comment at the header of each program; all programs are included in CD.
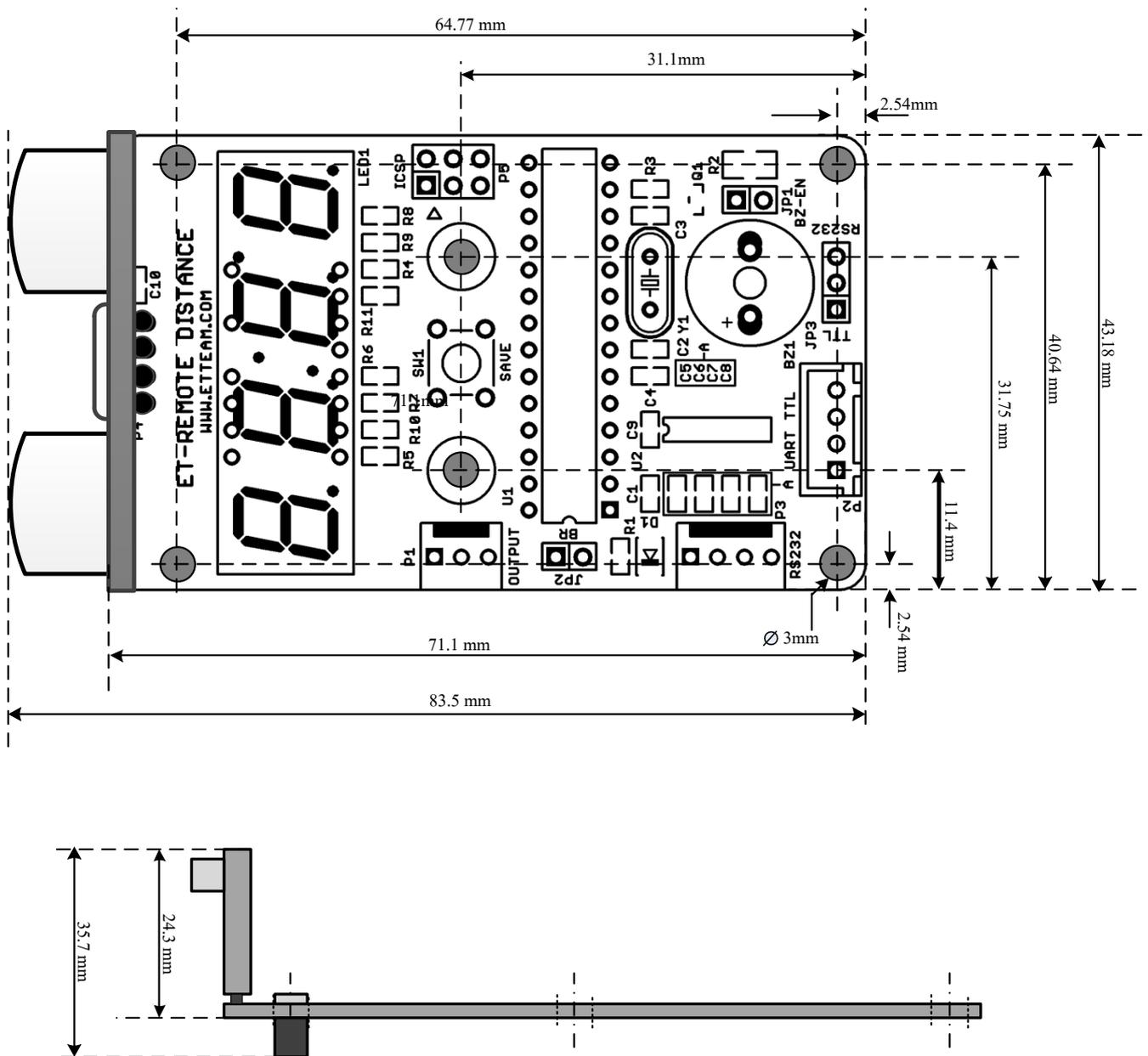
//---//

Size of ET-REMOTE DISTANCE

Circuit of ET-REMOTE DISTANCE