ET-MATRIX KEY DECODER is Module to decode state of pressing Key Pad Matrix 4x4 and 4x3. It sends out 2 types of the pressed Key to user; firstly, it is BCD 8421 type, including the state of pressing/releasing Key at Connector 8-Pin; and, secondly, it is ASCII Code type via Connector RS232. The Module has Buzzer and Connector to interface with LED that can show state of pressing/releasing Key. Remember, Key Pad that can be connected with must has the structure and pin arrangement of Matrix Key according to Connector Key_IN of Module only.

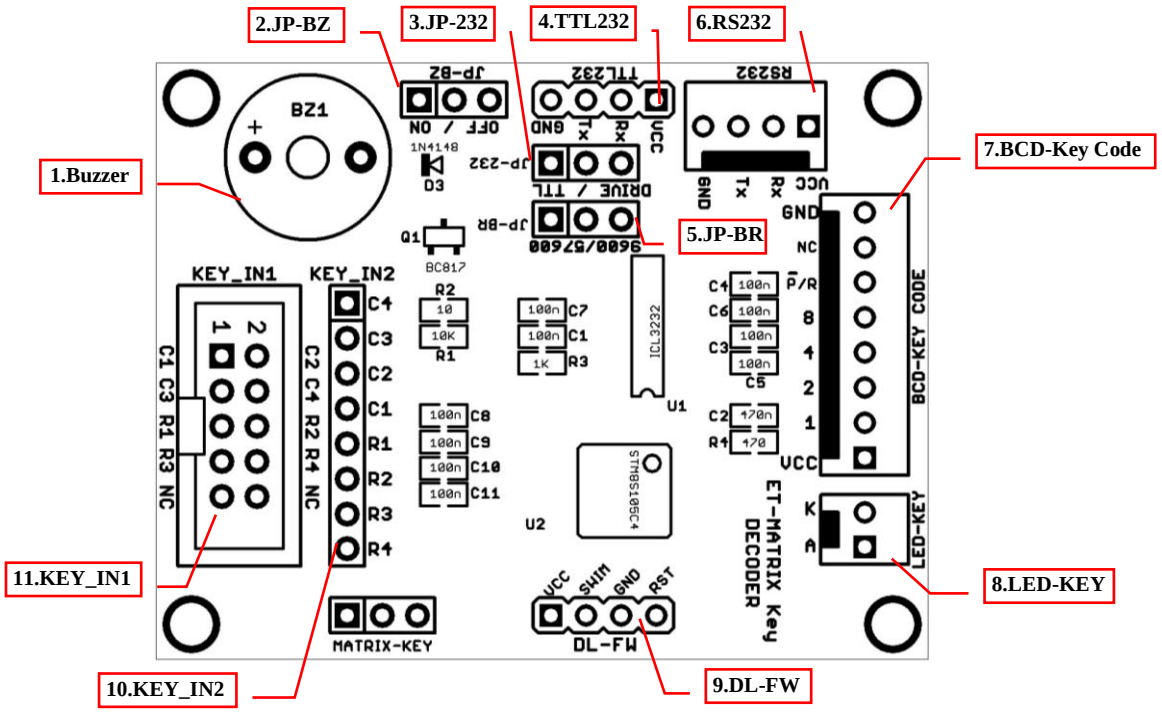
1. SPECIFICATIONS OF MODULE ET-MATRIX KEY DECODER

- Support Key Pad Matrix 4x4 and 4x3. User can read the feature of Key Pad that can be directly connected from the section “Structure of Module”. In case of Key Pad that has different pin arrangement unlike Connector Key_IN of the Module, user can re-arrange pin to wire Cable of Key Pad by self.
- This Module has 2 types of Connector Key_In that can support Key Pad; Block 10Pin and 8-Pin Signal Row. However, each Key_In has different arrangement of Pin Matrix, it has to choose proper Connector Key-IN according to structure of the connective Key Pad.
- Use Power Source +3.3V or +5V for Module.
- Show the state of pressing Key by Beep sound of Buzzer (Set ON/OFF by Jumper) and provide 1-CH Connector to interface LED to show state of pressing Key as well.
- Module shows the initial state (Power-ON) by Beep sound twice.
- Module runs as Single Key type; it can press only one key each time, it cannot press 2 keys at the same time.
- Have 2 types of Output Key Code of the pressed Key as follows;

    1) **Binary Code (BCD8421)**: Send Key Code to Connector 8-Pin; in this case, Pin $\overline{P}$/R notifies the state of pressing/releasing Key.

    2) **ASCII Code**: Send Key Code via Connector RS232 (TTL and Line Drive); it can set 2 Baud Rates; 9600 and 57600 bit/s. The Key Code that is sent out is prefixed by the letter ‘P’ or ‘R’ to notify the state of pressing/releasing Key, and then follows by ASCII value of the pressed Key.

2. STRUCTURE OF MODULE ET-MATRIX KEY DECODER



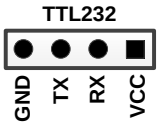
-1. ***Buzzer***: It produces Beep sound when Power-ON Module and after pressed Key.

-2. ***JP-BZ***: This Jumper sets ON/OFF Buzzer; it is effective after the change of setting Jumper.

-3. ***JP-232***: This Jumper is set to choose Connector 232. If Jumper is set to the side of TTL, it chooses Connector TTL232 (No.4); or, if Jumper is set to the side of Drive, it choses Connector RS232 (No.6) instead.

**When reading Key Code as ASCII Code, it has to choose only one type of Connector 232 and set Jumper JP-232 correctly according to the used Connector type. When choosing any Connector type, it depends on type of Board MCU that is connected with to read Key Code. If the Board has been connected with IC Line Driver 232 or user requires connecting to Port 232 of PC, it chooses Connector RS232 (No.6) and then sets Jumper JP-232 to the side of Drive. Or, if connecting with Pin Uart of MCU directly, it chooses Connector TTL232 (No.4) and sets Jumper JP-232 to the side of TTL instead.*

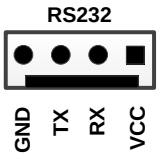
-4.***TTL232***: This Connector 232 is TTL type that can be connected with Pin Uart (Rx,Tx) of MCU directly; it has to cross Cables between Rx and Tx as pin arrangement below. Pin VCC and GND are connected with Power Supply 3.3V or 5V for the Module.



Show Pin Arrangement of Connector TTL232

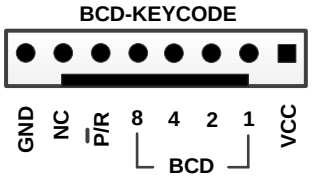
-5. ***JP-BR***: This Jumper sets Baud Rate for RS232 Communication; there are 2 values; 9600 and 57600 bit/s. When user requires using any Baud Rate, it has to set this Jumper to the preferable side. Remember, this new Baud Rate is effective after Power-On Module.

-6. ***RS232***: This Connector RS232 has IC Line Driver on board completely; so, it can connect with Port 232 of PC or Port 232 of MCU as IC Line Driver type instantly. In this case it has to cross Cables between Rx and Tx as pin arrangement below. Pin VCC and GND are connected with Power Supply 3.3V or 5V for the Module.



Show Pin Arrangement of Connector RS232

-7. ***BCD-Key Code***: This Connector 8-Pin sends out Key Code of the pressed or released Key in the format of Binary BCD8421, including state of pressing/releasing Key to user. Please look at the pin arrangement below;



Show Pin Arrangement of Connector BCD-KEY CODE

Details of Pin

VCC/GND = Connect Power Source 3.3V (if using with 3.3V MCU) or 5V (if using with 5V MCU) for Module

BCD = This PIN sends out Key Code 4-Bit. PIN No.8 is Bit MSB, the Key Code is sent out every time pressing and releasing Key. This Key Code is the value of the Key that is recently pressed or released, and the value still remains until user presses any new Key.

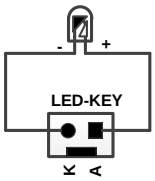
$\overline{P}$/R = This Pin receives/sends state of pressing/releasing Key. When it is normal status when no pin is pressed, this Pin becomes Logic 1 and still remains. When it presses any Key, this Pin becomes Logic 0 and still remains until user releases the Key.

NC = No Connect (Disconnection)

***NOTE**: It has to choose only one type of reading Key Code between ASCII Code or Binary Code because it has to choose only one Connector as well. If reading Key Code as ASCII Code, it has to choose Connector RS232 or TTL232; or, if reading Key Code as Binary BCD8421, it has to choose Connector BCD-Key Code.*

For Power Source of Board (VCC, GND), it chooses only one Connector between Connector RS232 or TTL232 or BCD-Key Code.

-8. ***LED-KEY***: This Connector 2 PIN connects with LED to shows state of pressing Key; it has to connect anode terminal (+) of LED with Pin A and cathode terminal (-) of LED with Pin K. LED is lit up after pressed the Key and LED is off after released the Key.



Show how to interface LED to notify the state of pressing Key

-9. ***DL-FW***: Upgrade new Firmware for the Module (normally, it has to return to ETT to upgrade).

-10. ***KEY_IN2***: This Connector Key Pad 8-Pin Single Row is especially designed for using with Key Pad that has the same pin arrangement and structure of Matrix Key (Row/Column) as the diagram below; so, it can be directly connected together without soldering new any cable.

Example of Key Pad 4x4

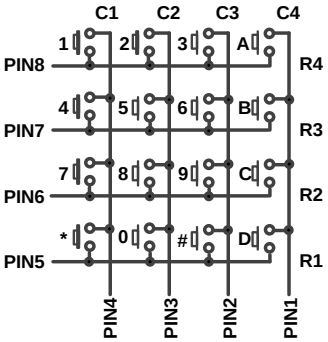
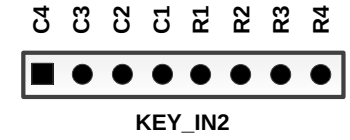


Diagram Key Pad 4x4



Pin arrange of Connector KEY_IN2

For other Key Pad that has different pin arrangement and structure from the diagram above, user can wire cable from Key Pad to Connector KEY_IN2 by self. Remember, when connecting new cable, it has to refer to the position of Pin Row/Colum according to the diagram above.

-11. ***KEY_IN1***: This Connector Key Pad 4x4 Block 10 Pin that is especially designed for using with Key Pad that has the same pin arrangement and structure of Matrix Key (Row/Column) as the diagram below; so, it can be directly connected together without soldering any new cable.

Example of Key Pad 4x4

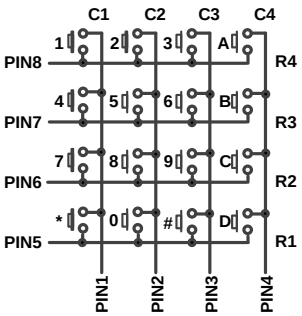
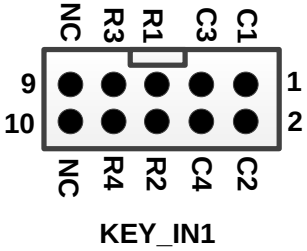


Diagram Key Pad 4x4



Pin Arrangement of Connector KEY_IN1

Moreover, the Key Pad 4x3 that has the same structure of Matrix Key, pin arrangement and position of Row/Column as the diagram below, it can be connected together with Connector KEY_IN1 instantly; in this case, it requires ET-Convert Key M1 to convert Port Key into Block 10 Pin.

Feature of Key Pad 4x3

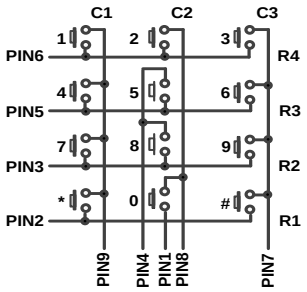


Diagram Key Pad 4x3

Other Key Pad 4x4 and 4x3 that has different structure and pin arrangement from the diagram above, user can wire cable from Key Pad to the Connector KEY_IN1 by self. Remember, when connecting new cable, it has to refer to the position of Pin Row/Column according to the example Diagram of Key_IN1. *When connecting cable from KEY PAD 4x3 to KEY_IN1, the side of Row of Key PAD 4x3 must be connected with Pin R1,R2,R3,R4 and the side of Column must be connected with Pin C1,C2,C3,C4 of KEY_IN1.*

3. OPERATION and How to read Key Code of ET-MATRIX KEY DECODER

Operation: After provided power to the Module completely, user can hear Beep sound twice, state of PIN BCD at Connector BCD-KEYCOD becomes Logic 0, and Pin $\overline{P}$/R becomes Logic 1. For Connector 232, it does not send out any data. When pressed any Key, user can hear Beep sound once at Connector LED-Key. Moreover, if connected with LED, LED is lit up and still remains until user released the Key and then the LED is off.

Every time it presses or releases any Key, the Output Key Code in the format of Binary and ASCII will be sent out to Connector 232 and BCD-KEY Code. The state of Signal $\overline{P}$/R becomes Logic 0 after the Key is pressed and it becomes Logic 1 instead after released the Key.

TABLE: KEY CODE of ET-MATRIX KEY DECODER

KEY	FOR Binary MODE						FOR ASCII Mode (RS232 TTL-Line Drive)	
	BCD 8421 KEY CODE						ASCII KEY CODE	
	8	4	2	1	HEX		ASCII	HEX
1	0	0	0	1	0x01		'1'	0x31
2	0	0	1	0	0x02		'2'	0x32
3	0	0	1	1	0x03		'3'	0x33
4	0	1	0	0	0x04		'4'	0x34
5	0	1	0	1	0x05		'5'	0x35
6	0	1	1	0	0x06		'6'	0x36
7	0	1	1	1	0x07		'7'	0x37
8	1	0	0	0	0x08		'8'	0x38
9	1	0	0	1	0x09		'9'	0x39
0	0	0	0	0	0x00		'0'	0x30
A	1	0	1	0	0x0A		'A'	0x41
B	1	0	1	1	0x0B		'B'	0x42
C	1	1	0	0	0x0C		'C'	0x43
D	1	1	0	1	0x0D		'D'	0x44
E(#)	1	1	1	0	0x0E		'E'	0x45
F(*)	1	1	1	1	0x0F		'F'	0x46

***Note**: For Matrix Key 4x3, the Code value will be varied according to Key 0-9 and E-F; in this case, Key E=Key # and Key F=Key **

3.1 How to read Key Code as Binary BCD8421

When reading Key Code as Binary BCD8421, it reads value from Connector BCD-Key Code. Signals of Connector will be varied according to the circumstances as follows;

- ***State of Initial (Default)*** = When Power-ON, the value at Connector O/P BCD Key Code will be set as follows;

PIN $\overline{P}$/R = 1

PIN8 = 0 ; PIN4 = 0 ; PIN2 = 0 ; PIN1 = 0

- ***State of pressing Key (Press)*** = When pressed Key, user can hear Beep sound and LED Status is lit up (if connected LED) and the value at Connector BCD-KEY CODE will be varied as follows;

1) Pin $\overline{P}$/R (Signal of pressing/releasing Key) is changed from Logic 1 to Logic 0 and it still remains as long as user is still pressing the Key.

2) PIN BCD 8421 changes the state according to the Key Code value of the recently pressed Key. Please refer to Table Key Code above, especially in the HEX on the upper left.

- ***State of releasing Key (Release)*** = When released the Key, the value at Connector BCD-KEY CODE will be varied as follows;

1) PIN $\overline{P}$/R (Signal of pressing/releasing Key) will be changed from Logic 0 to Logic 1 and it still remains as long as user is releasing the Key.

2) PIN BCD8421 will be changed according the Key Code value of Key that is recently released.

3.2 How to read Key Code as ASCII

When reading Key Code as ASCII, it can read the value from Connector RS232 or TTL232 that is Serial Port Interface. It can set 2 Baud Rates; 9600 and 57600 bit/s by setting Jumper JP-BR.

When choose any Connector, it depends on Board MCU type that is connected with Module, please check if it is Port 232 as TTL type or IC Drive 232 type. If it is TTL Type ((Tx,Rx is connected from Pin MCU directly), it has to choose Connector TTL232 and then set Jumper JP-232 to the side of TTL. If it is connected via IC Drive 232 or it is connected to RS232 of PC, it has to choose Connector RS232 of Module and then set Jumper JP-232 to the side of DRIVE instead. Remember, it has to cross the cables between Rx and Tx, whichever Connector user chooses; it has to connect Pin TX with Pin RX of other side and it has to connect Pin RX with Pin TX of other side.

The Key Code that is sent out to user is ASCII Code and the format of the value will be changed according to the state of pressing Key as follows;

- ***State of initial (Default)*** = When Power-ON, it does not send out any data to Port 232; it means that the Module does not send out any data to Port 232 if it does not press any Key.

-***State of pressing Key (Press)*** = When pressed Key, the Module sends out the 3-Byte Data to Port 232. The first Byte it sends is ASCII 'P' to notify the state of pressing Key, the second Byte is the ASCII Code value of the pressed Key, and the last Byte is 0x0D. The Key Code value will be changed according to the Table Key Code above, especially in the ASCII or HEX on the upper right.

	ASCII CODE		Hex
	Status Key (Byte 1)	Key Code (Byte 2)	End Byte (Byte 3)
PRESS KEY	'P' (0x50)	'0-9', 'A-F'	0x0D

Ex: After pressed the Key F (Key *), the value will be **PF** and **0x0D**

Table shows format of sending Data after pressed the Key

-***State of releasing Key (Release)*** = When released the pressed Key, the Module sends out 3-Byte Data to Connector 232. The first Byte it sends out is ASCII 'R' to notify the state of releasing Key, the second Byte is the ASCII Code value of the released key, and the last Byte is 0x0D.

	ASCII CODE		Hex
	Status Key (Byte 1)	Key Code (Byte 2)	End Byte (Byte 3)
RELEASE KEY	'R' (0x52)	'0-9', 'A-F'	0x0D

Ex: After released the Key 0, the value will be **R0** and **0x0D**

Table shows format of sending Data after released Key

Summarize how to read Key Code as ASCII: When writing program, user can check state of pressing/releasing Key by character 'P' = Press Key and 'R' = Release Key; this value is the first Byte that is sent out. Moreover, it can read Key Code value of the pressed or released Key in the second Byte

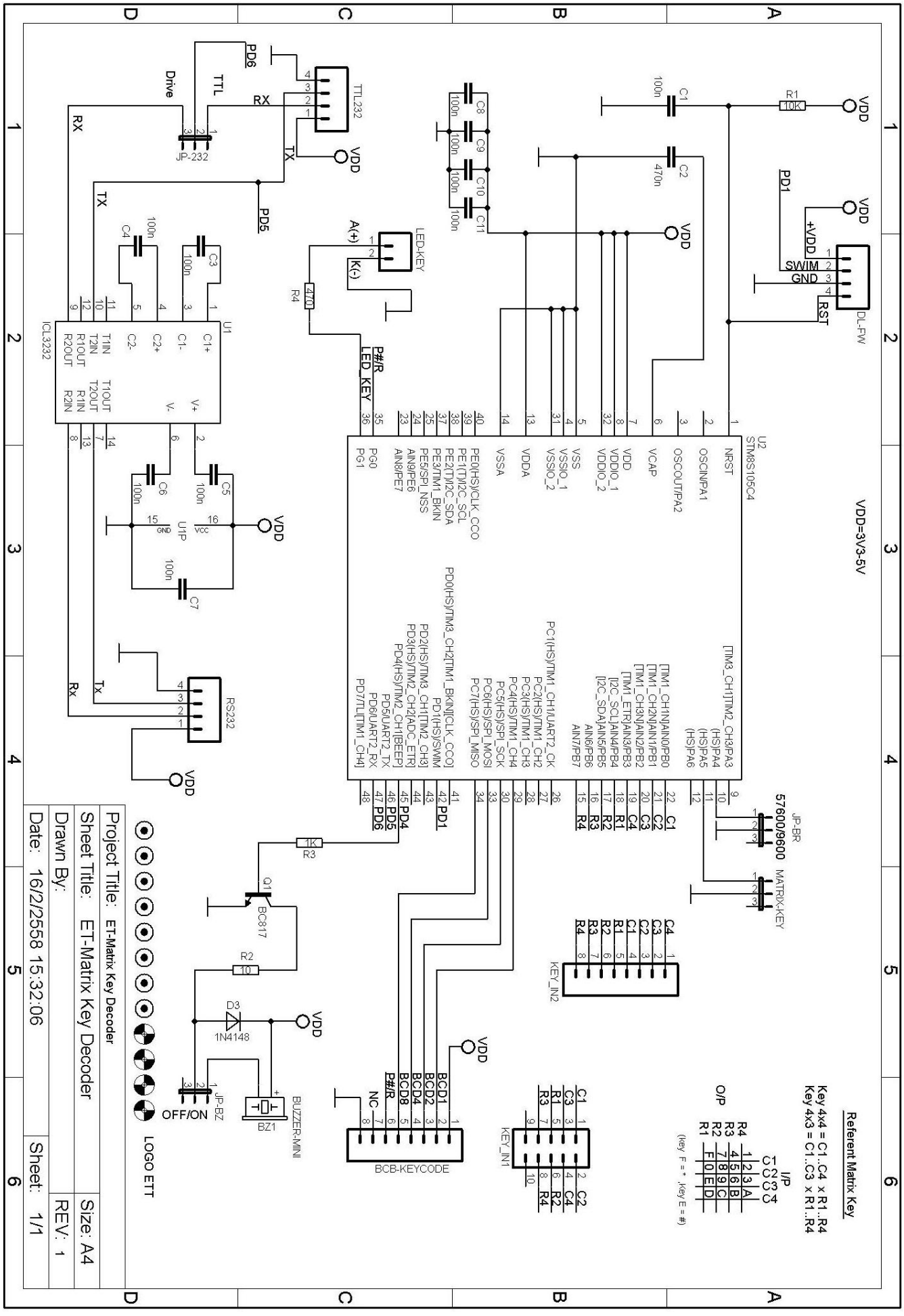
4. EXAMPLE PROGRAMS

The Example Programs are provided in the CD-ROM for AVR EASY 168(Arduino) and AVR EASY 168(Arduino), it is written by C Language. Example program for each MCU is the same. When testing the example program, user can read details of port connection, model of test board and Compiler version for writing program of each MCU at Comment above the Program header. The operation of program is described below;

Ex1_Read_BCD_Code: This Example Program illustrates how to read Key Code as BCD 8421 from Module. When connected Key Pad at Input of the Module and provided Power into the Module completely, the program starts checking state of pressing Key by Pin $\overline{P}$/R. If state of Pin $\overline{P}$/R is '0', it means that the Key is pressed; so, the program reads and stores the Key Code value of the pressed Key. Next, it sends the Key Code value to display on LED0-LED3 and it sends to Port RS232 (BR=9600) to show the Key Code value on Hyper Terminal of PC. For example, if pressed Key 9, the value that is shown on LED and Hyper Terminal is 0x09.

Ex2_Read_ASCII_Code: This Example Program illustrates how to read Key Code as ASCII via RS232 of Module and it sets Baud Rate at 57600 bit/s. When connected Key Pad at Input of the Module and provided Power into the Module completely, the program starts checking if the received Data via RS232 is pressed Key. If the received Data is prefixed by the letter 'P', it means that the Key is pressed and the next Data is ASCII Code of the pressed Key; next, the program reads this Key Code value to display on the LED in the format of ASCII Code of the pressed key. For example, if pressed the Key 1, the reading value on LED is 0x31 that is the ASCII Value of Key 1.

********** END **********


VDD=3V3-5V
U2
STM8S105C4
NRST
OSCIN/PA1
OSCOUT/PA2
VCAP
VDD
VDDIO_1
VDDIO_2
VSS
VSSIO_1
VSSIO_2
VDDA
VSSA
[TIM3_CH1]TIM2_CH3/PA3
(HS)PA4
(HS)PA5
(HS)PA6
[TIM1_CH1N]AIN0/PB0
[TIM1_CH2N]AIN1/PB1
[TIM1_CH3N]AIN2/PB2
[TIM1_ETR]AIN3/PB3
[I2C_SCL]AIN4/PB4
[I2C_SDA]AIN5/PB5
AIN6/PB6
AIN7/PB7
PC1(HS)/TIM1_CH1/UART2_CK
PC2(HS)/TIM1_CH2
PC3(HS)/TIM1_CH3
PC4(HS)/TIM1_CH4
PC5(HS)/SPI_SCK
PC6(HS)/SPI_MOSI
PC7(HS)/SPI_MISO
PD0(HS)/TIM3_CH2/TIM1_BKIN/[CLK_CCO]
PD1(HS)/SWIM
PD2(HS)/TIM3_CH1/[TIM2_CH3]
PD3(HS)/TIM2_CH2/ADC_ETR]
PD4(HS)/TIM2_CH1/[BEEP]
PD5/UART2_TX
PD6/UART2_RX
PD7/TLI/[TIM1_CH4]
PE0(HS)/CLK_CCO
PE1(T)/I2C_SCL
PE2(T)/I2C_SDA
PE3/TIM1_BKIN
PE5/SPI_NSS
AIN9/PE6
AIN8/PE7
PG0
PG1
P#/R
LED_KEY
C1
C2
C3
C4
R1
R2
R3
R4
PD1
PD4
PD5
PD6
DL-FW
+VDD
SWIM
GND
RST
JP-BR
57600/9600
MATRIX-KEY
KEY_IN2
KEY_IN1
BCB-KEYCODE
BCD1
BCD2
BCD4
BCD8
P#/R
NC
LED-KEY
A(+)
K(-)
TTL232
JP-232
Drive
TTL
RX
TX
ICL3232
U1
C1+
C1-
C2+
C2-
V+
V-
T1IN
T2IN
R1OUT
R2OUT
T1OUT
T2OUT
R1IN
R2IN
U1P
GND
VCC
RS232
Tx
Rx
Q1
BC817
D3
1N4148
JP-BZ
OFF/ON
BUZZER-MINI
BZ1
LOGO ETT
Referent Matrix Key
Key 4x4 = C1..C4 x R1..R4
Key 4x3 = C1..C3 x R1..R4
I/P
O/P
(key F = *, key E = #)
Project Title: ET-Matrix Key Decoder
Sheet Title: ET-Matrix Key Decoder
Size: A4
Drawn By:
REV: 1
Date: 16/2/2558 15:32:06
Sheet: 1/1