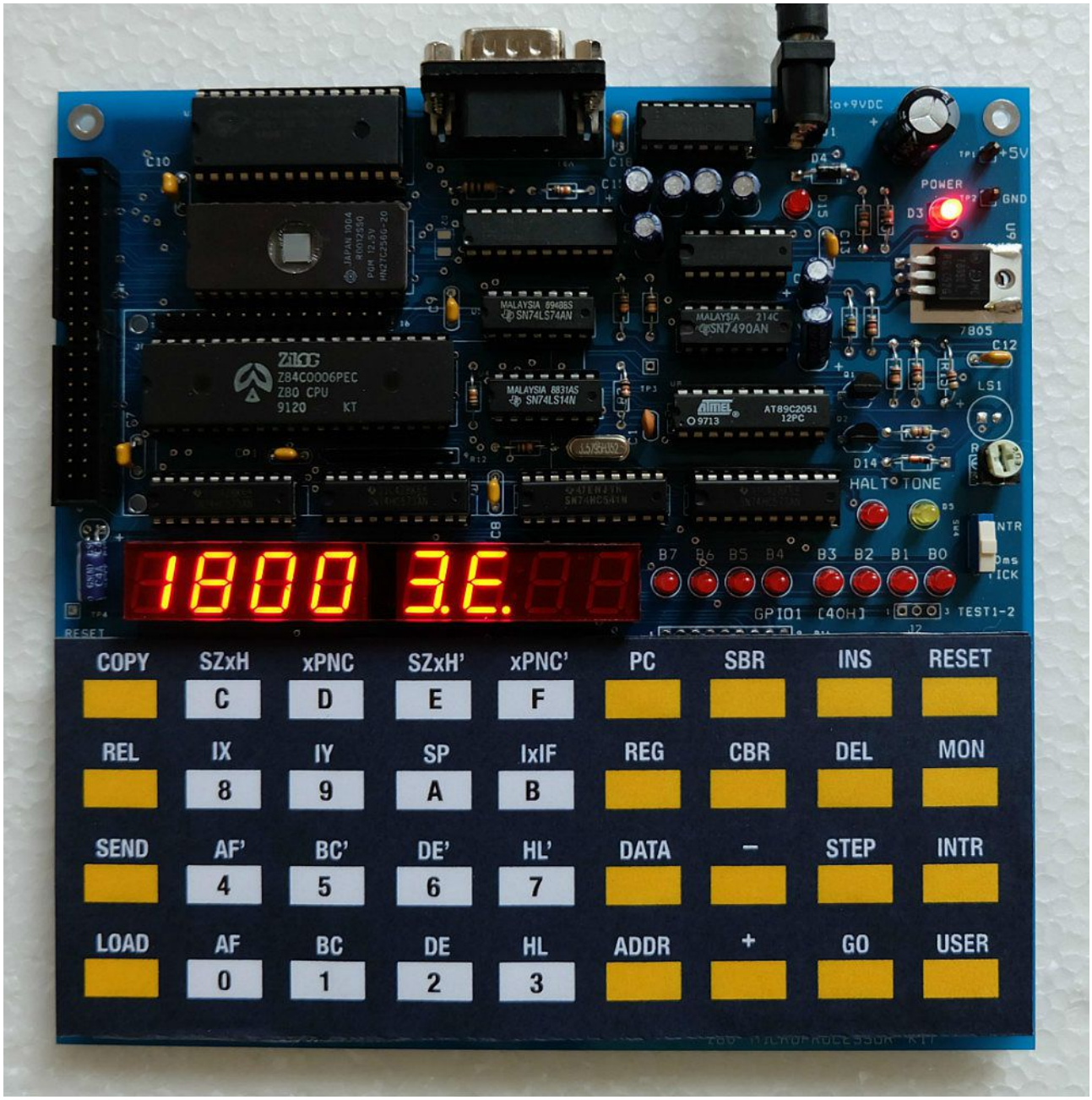
# Z80 Microprocessor Kit
# Construction Manual



# Z80 MICROPROCESSOR KIT
# CONSTRUCTION MANUAL

## CONTENTS

## INTRODUCTION

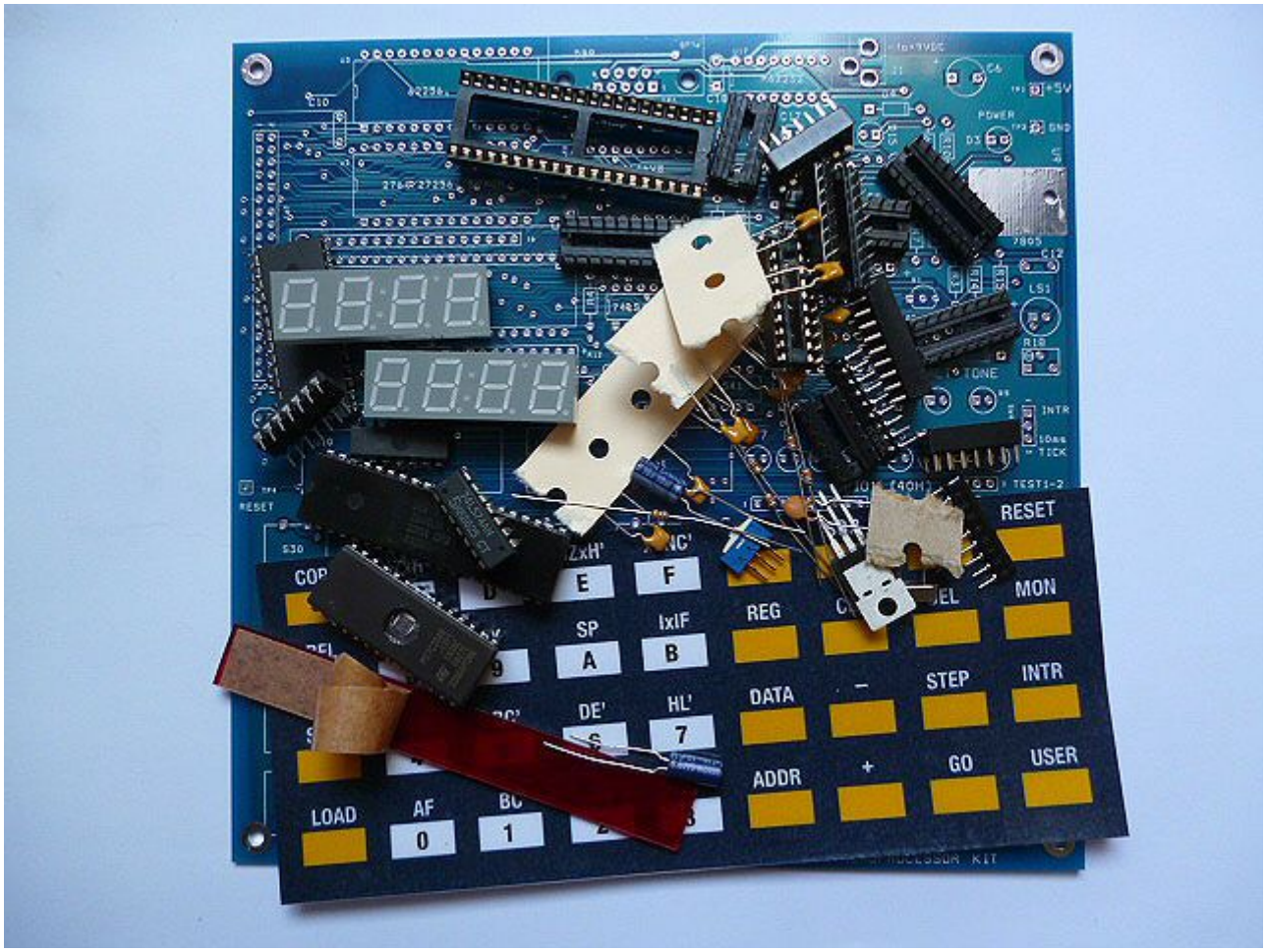
This construction manual provides steps to assemble the Z80 microprocessor kit.

Preprogrammed IC chips are 1) monitor ROM 27C256, 2) Programmable Logic Device, PLD GAL16V8 and 3) AT89C2051 microcontroller.

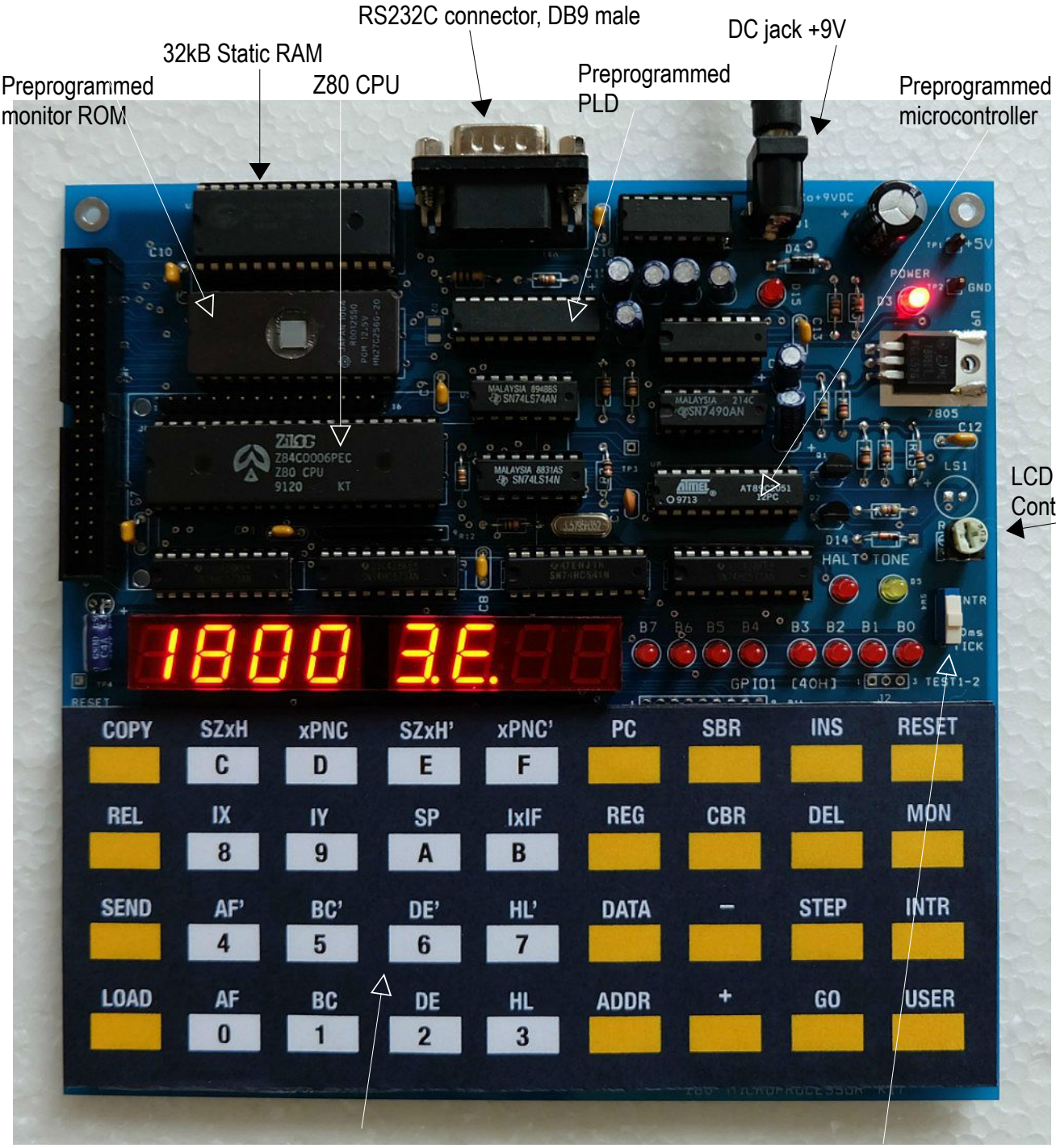
Be advised, 95% that causes not working circuit is from soldering! Good soldering point makes the current flow easily. Bad soldering points may causes high impedance path, consequences the distorted pulse signal! The Z80 will not functioning properly!

When pick the PCB, do not touch the soldering pad! Pick it with the board border only!

For those who live in dry and cold area, electrostatic discharge may zaps the CMOS chips. Keep them at the low impedance sheet, like aluminum foil.



## COMPLETE ASSEMBLED KIT



## PREPARING FOR ASSEMBLY

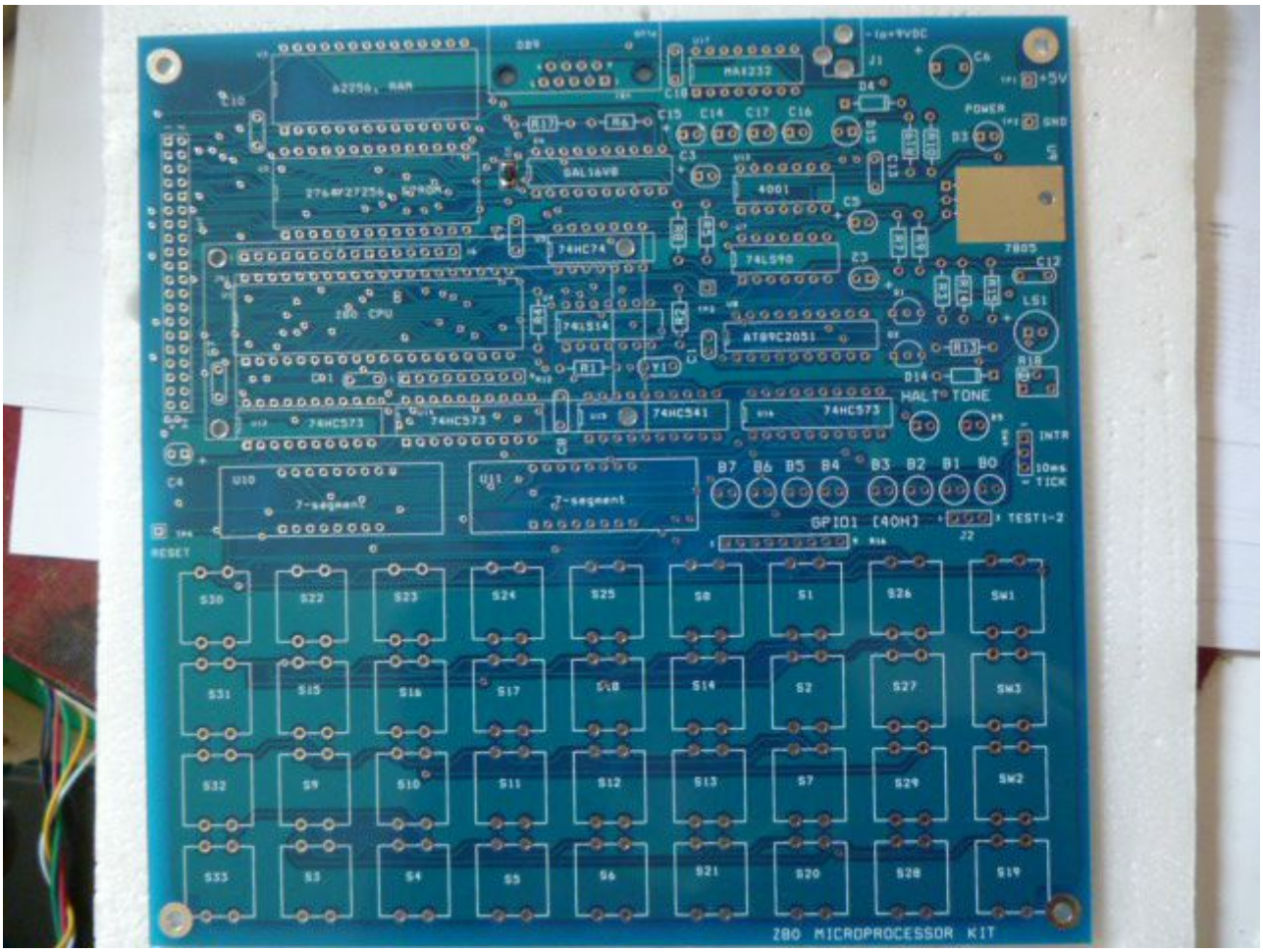
### Tools

1. The example of soldering tools is shown below. Main tools are soldering iron, solder lead, wire cutter. Voltmeter is for checking +5V power supply.

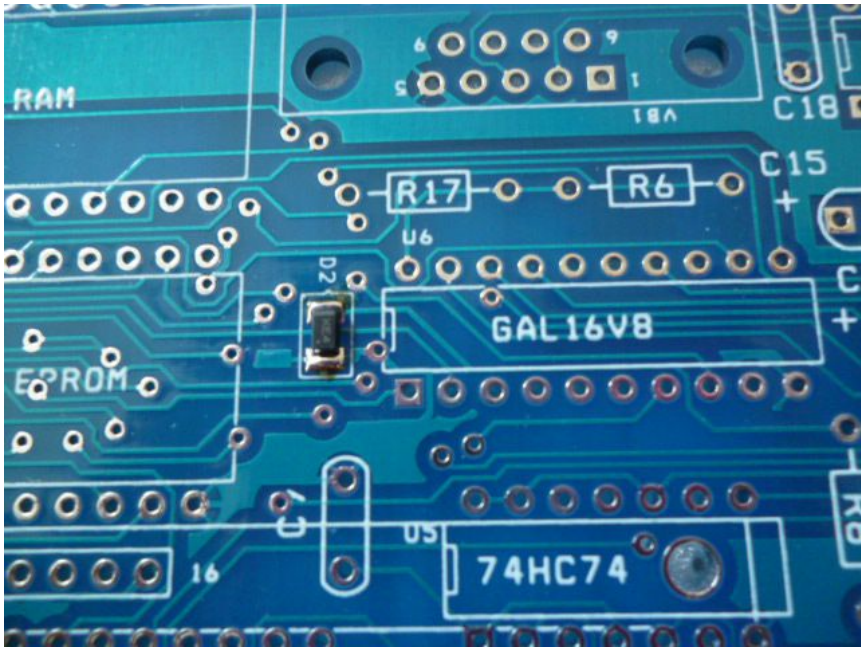

Author also has the logic probe, it can check the oscillator running easily. No need the expensive oscilloscope.

## ASSEMBLY STEPS

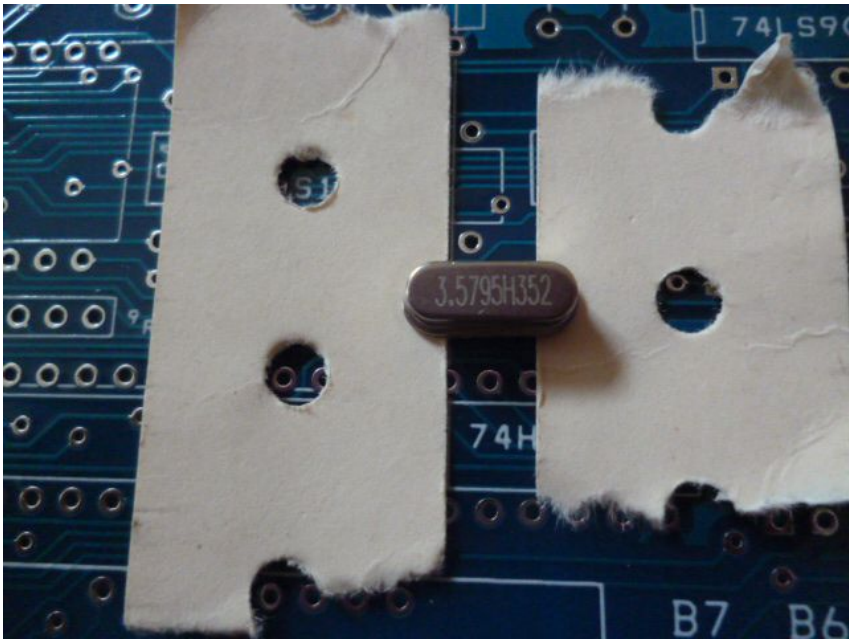
☐ 1.Check the PCB, no wire or conductive parts on both sides. Remember do not touch the PCB pad with finger. Pick it at the board border. The board comes with presoldered transient voltage suppressor, D2.



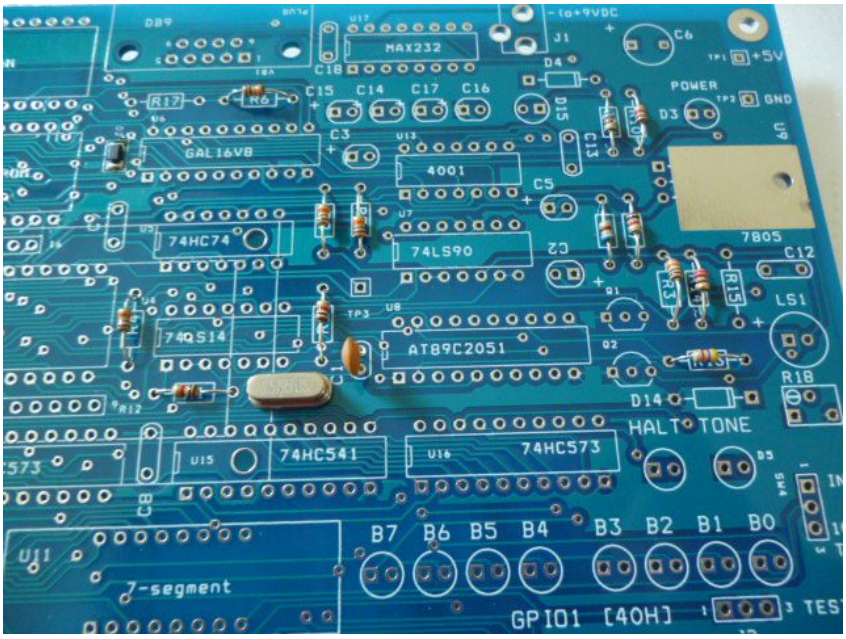
☐ 2.Check the transient voltage suppressor, D2 on the PCB. The gray line on the chip is cathode pin! It must be soldered as shown below.



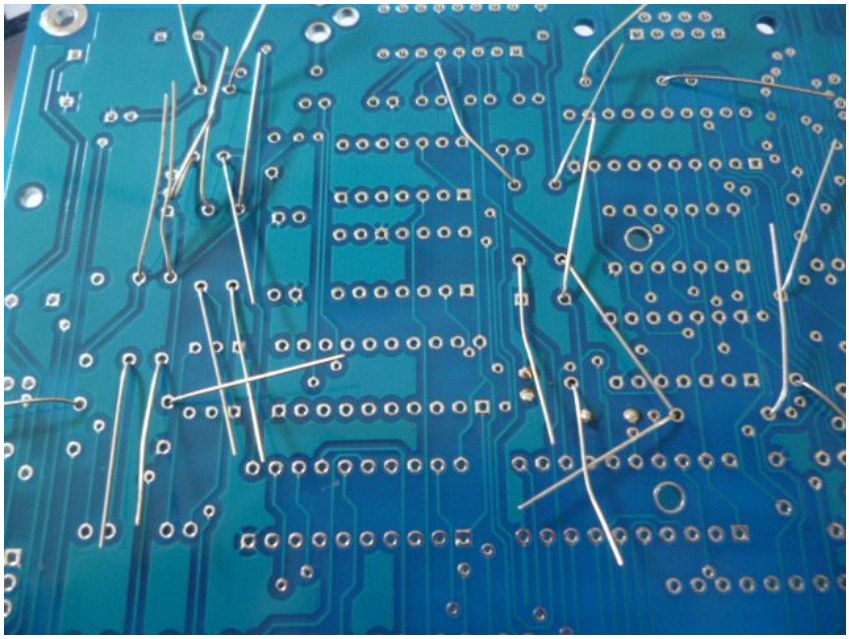

☐ 3.We begin with low profile devices. Insert Y1, XTAL 3.579MHz. To prevent the Xtal case touch the PCB, we may use two pieces of tick paper insert at the bottom. Use one finger press it while soldering at the soldered side.



☐ 4. Insert C1, 100pF ceramic disc capacitor then followed with resistors. Place them in axial direction.




For resistors and capacitors, we can bend the legs to make it in place. So we can solder it easily.

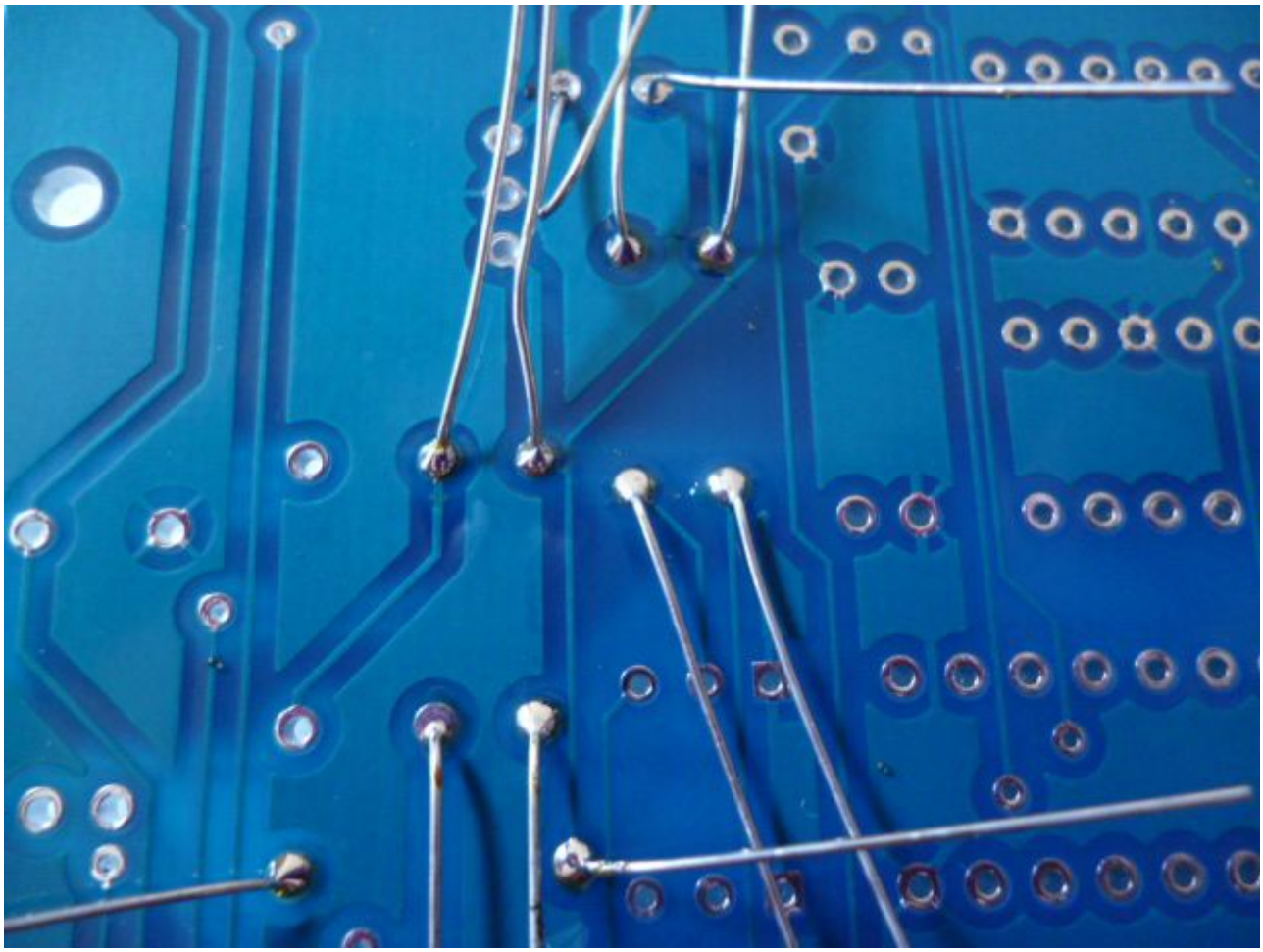


Finished soldering point should look like this.

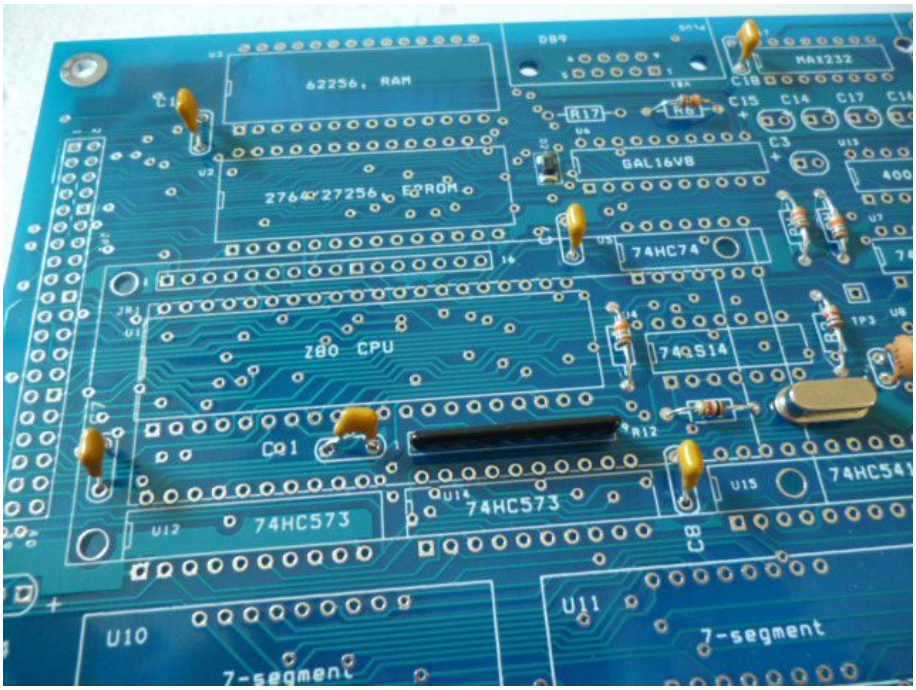
Remember, place the soldering iron's tip to touch to the pad and leg for 1-2 seconds, then put the solder wire at the tip, count one, two three, then take them out.

Do not lift the tip up and down! Enough solder will make the finished work like PAGODA and looks shine!

Cut the legs, we can insert another parts.

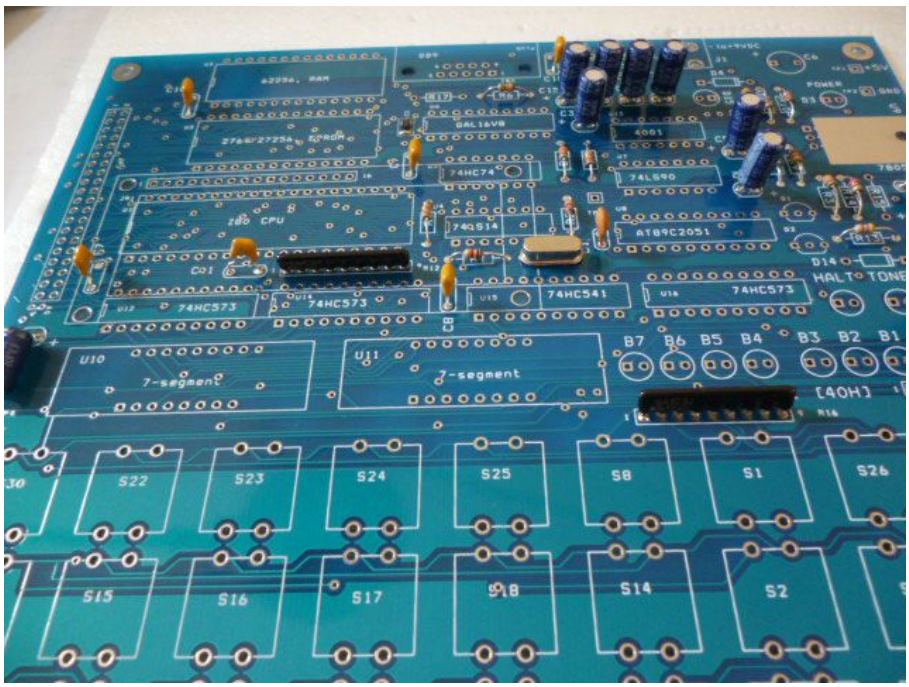


☐ 5. Now insert multilayer capacitors, 100nF and R12, R PACK. Solder it.

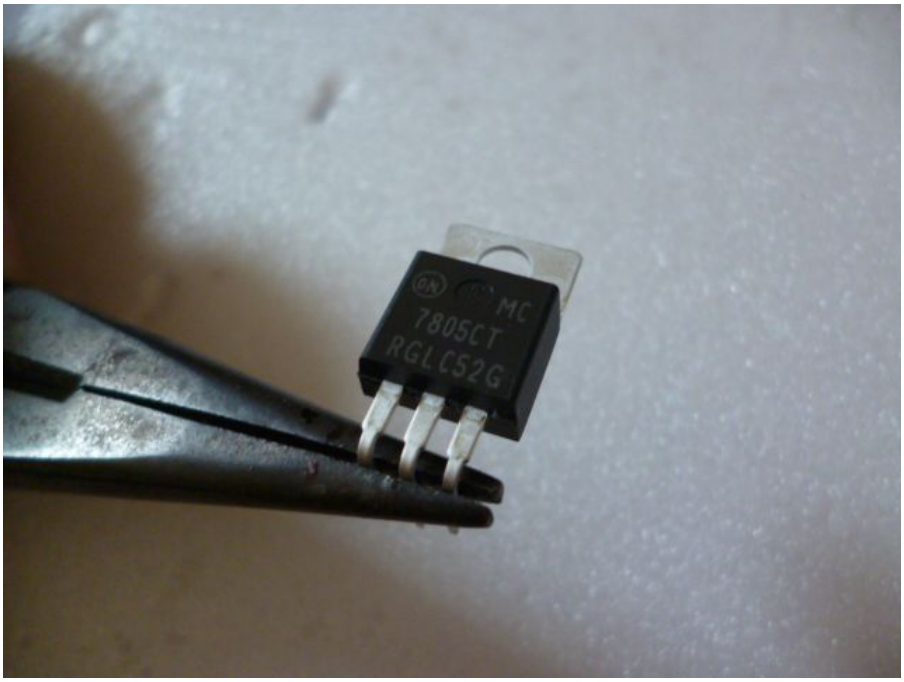


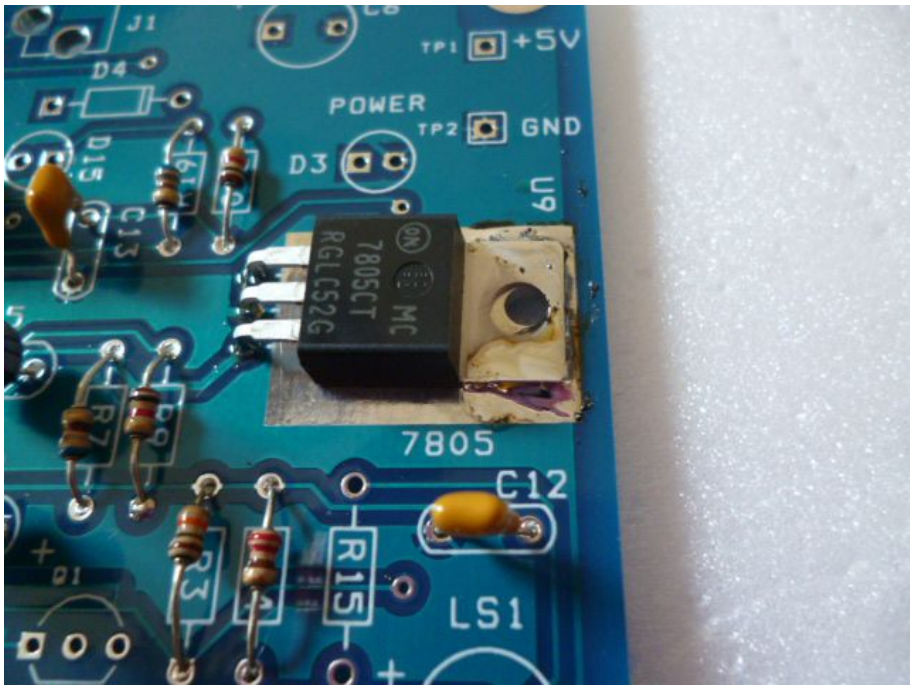

☐ 6. Insert R16, R PACK. Pin 1 of R pack is a dot point! Soldering R pack many need your finger to press it.

Then insert and solder electrolytic capacitors, 10uF. All electrolytic capacitors has polarity. Longer leg is positive, put it to the square footprint (with + indicator)!



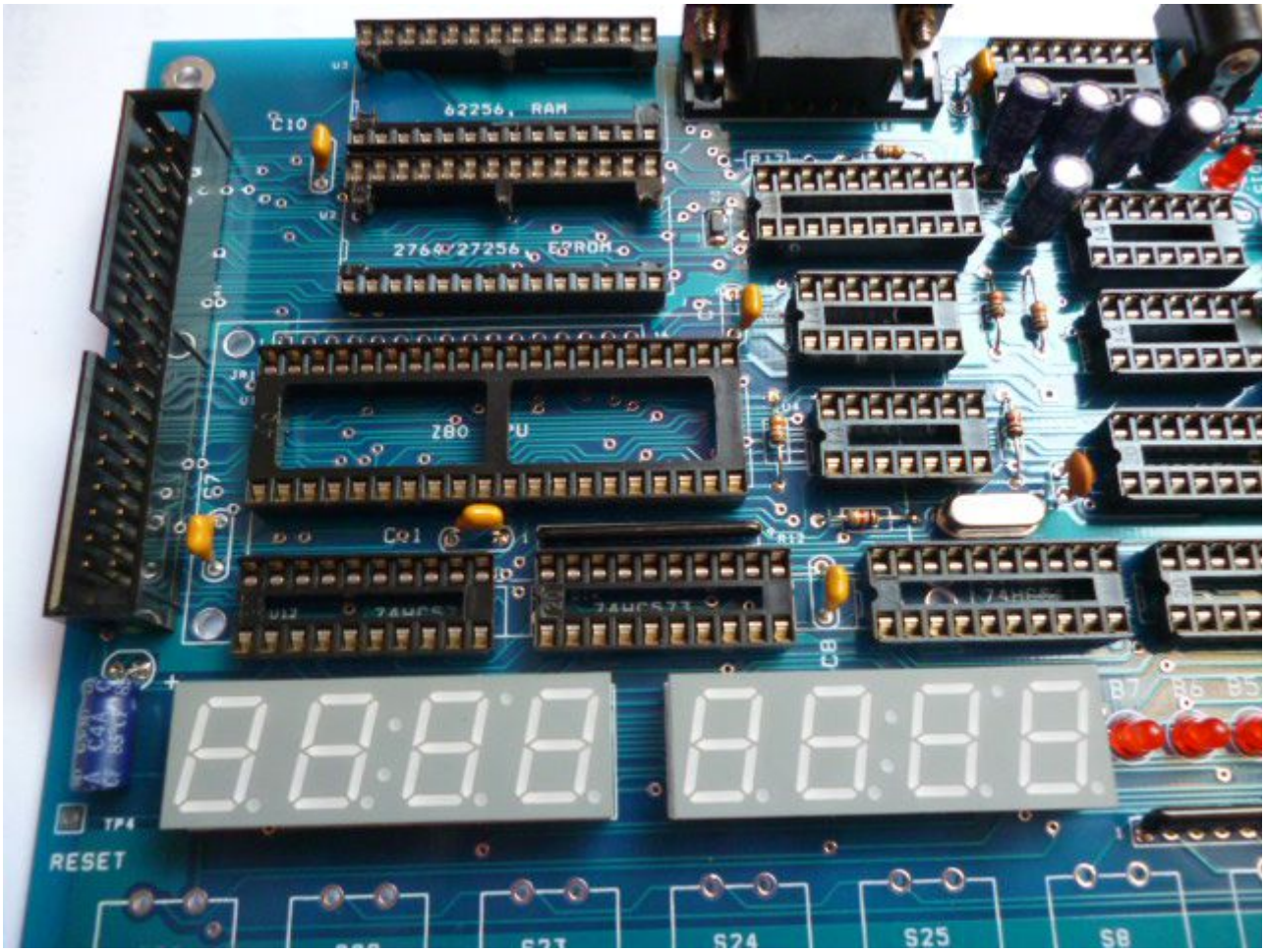
☐ 7. The voltage regulator, 7805 is TO220 package, we can bend the legs like this.




The TAP pin is GND for heatsink. We can solder it to the pad directly. The ground plan will be heatsink of the voltage regulator. You can tighten the TAP to the PCB with M3 nut and washer as well.

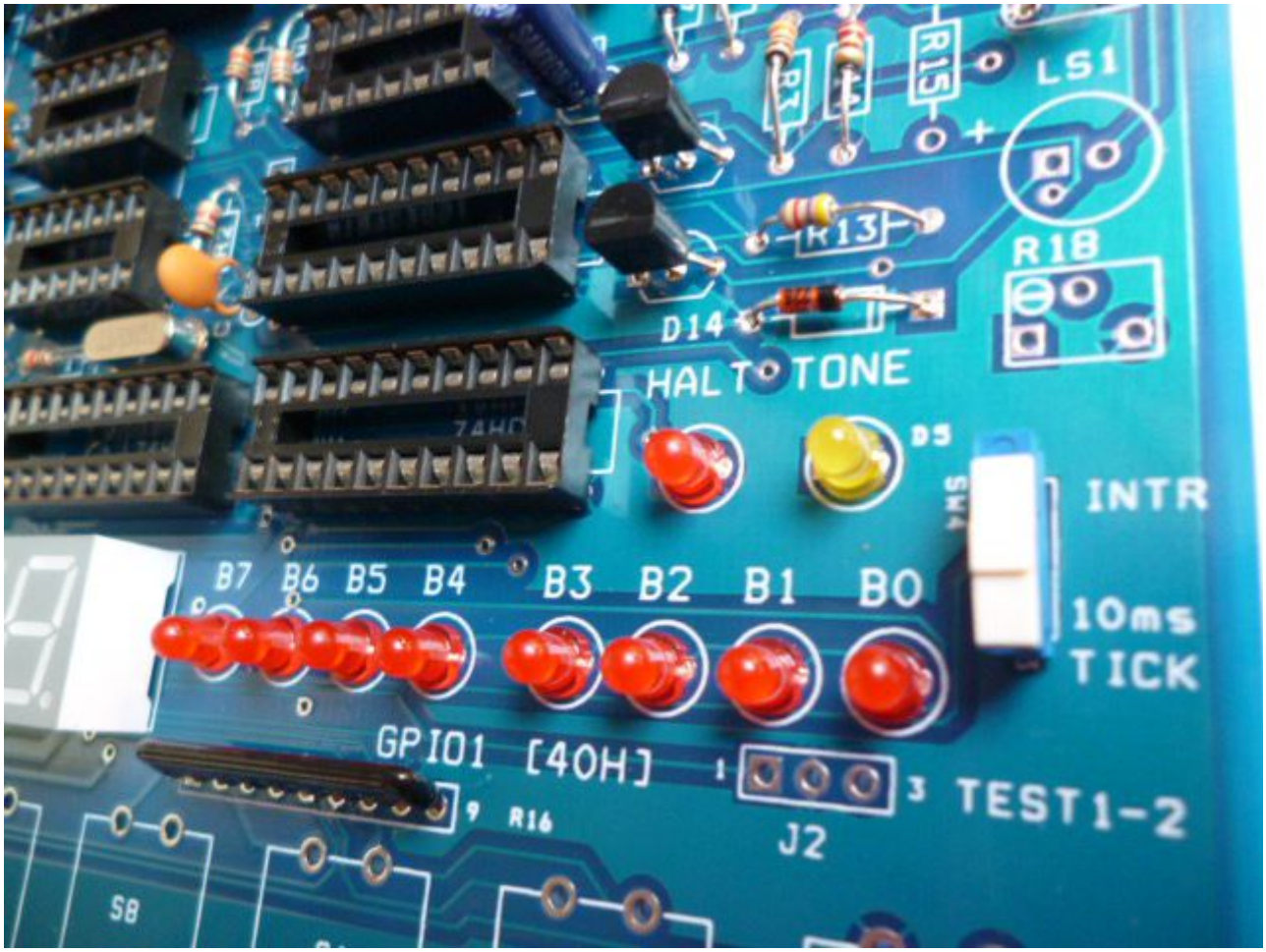
☐ 8. We can insert the IC socket and the rest parts.




For seven segment LED, we may solder only one pin beforehand. Then check the position, both should be line up on the same level.

We can adjust them by putting the solder tip, to make it slightly movable. Use finger to move it, make it in position.

☐ 9. The dot LED has polarity!, the flat side is CATHODE pin! The footprint for LED has square one for ANODE pin! Solder it only one pin beforehand, so we can adjust it in position easily. When it was in good position, finish the rest pins.



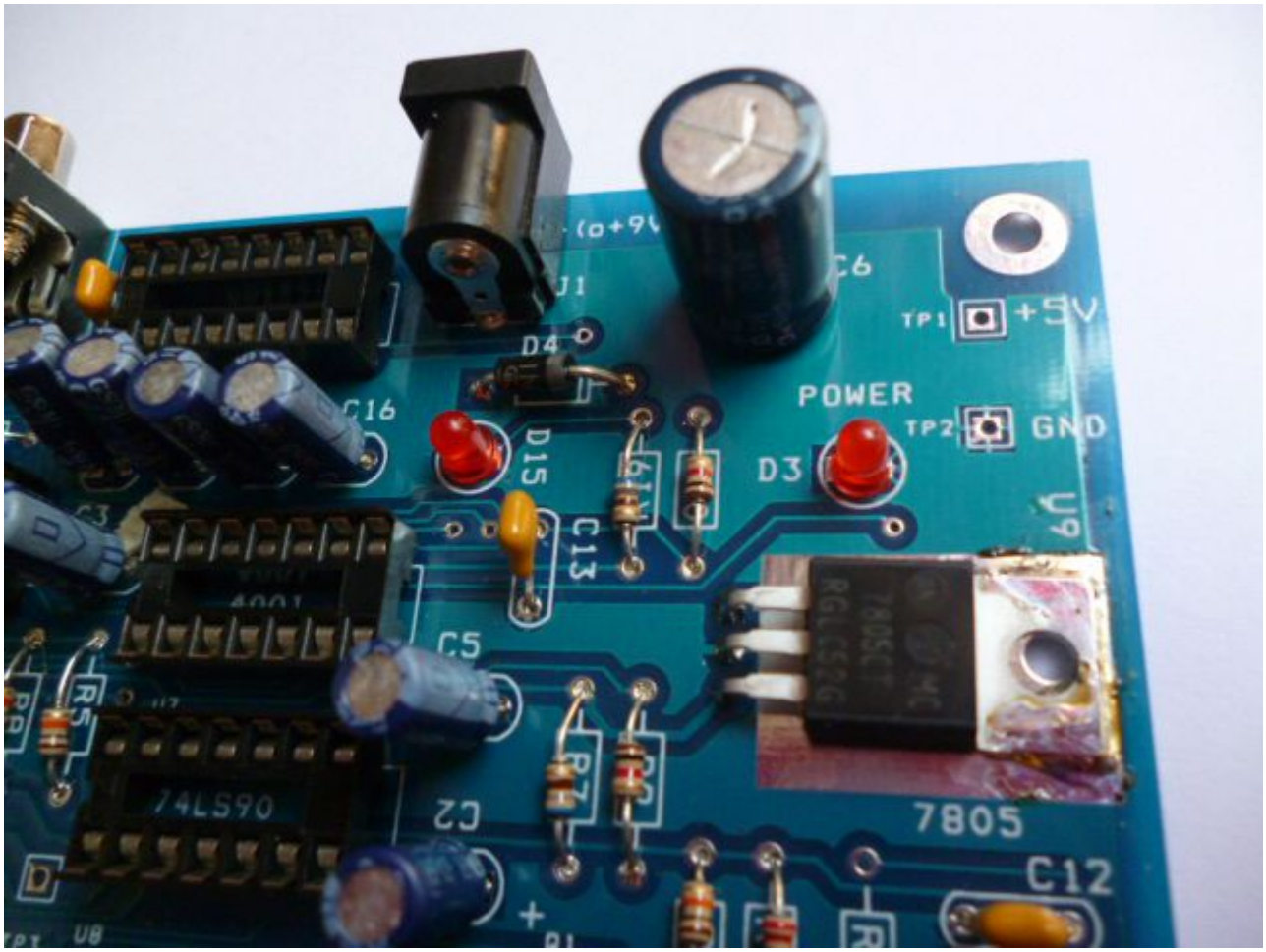

Again SW4 is not easy to stand, just solder one pin, then adjust the position with finger while soldering.

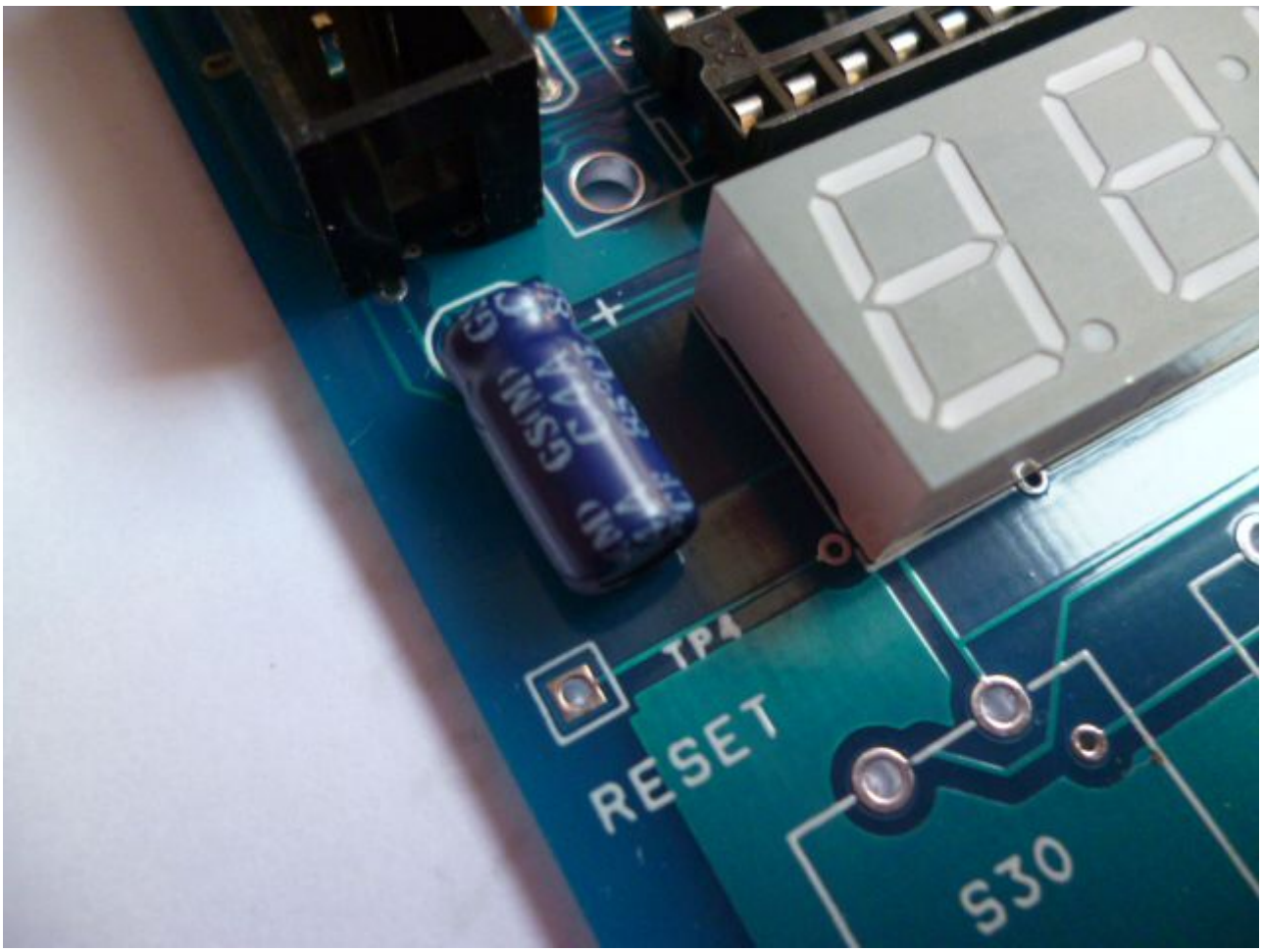
D14 also has polarity. Cathode pin has black line!

Q1 and Q2 are PNP transistors.  Place them in position the same as symbol on the PCB!

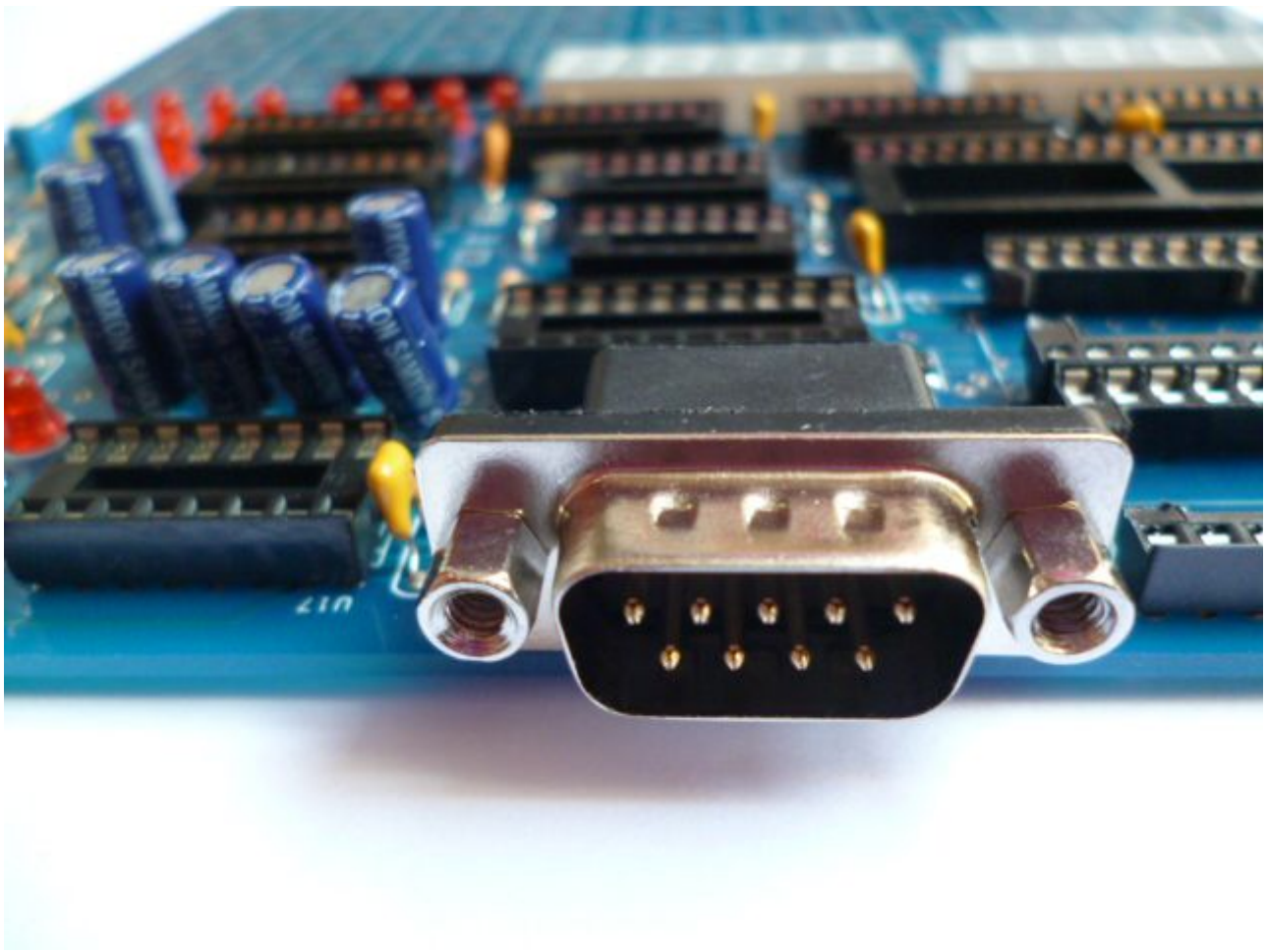
☐ 10. Insert DC jack, the positive pin will need a saw files for trimming.

Insert D4, protection diode. Insert C6, it has polarity! Recheck before solder them.





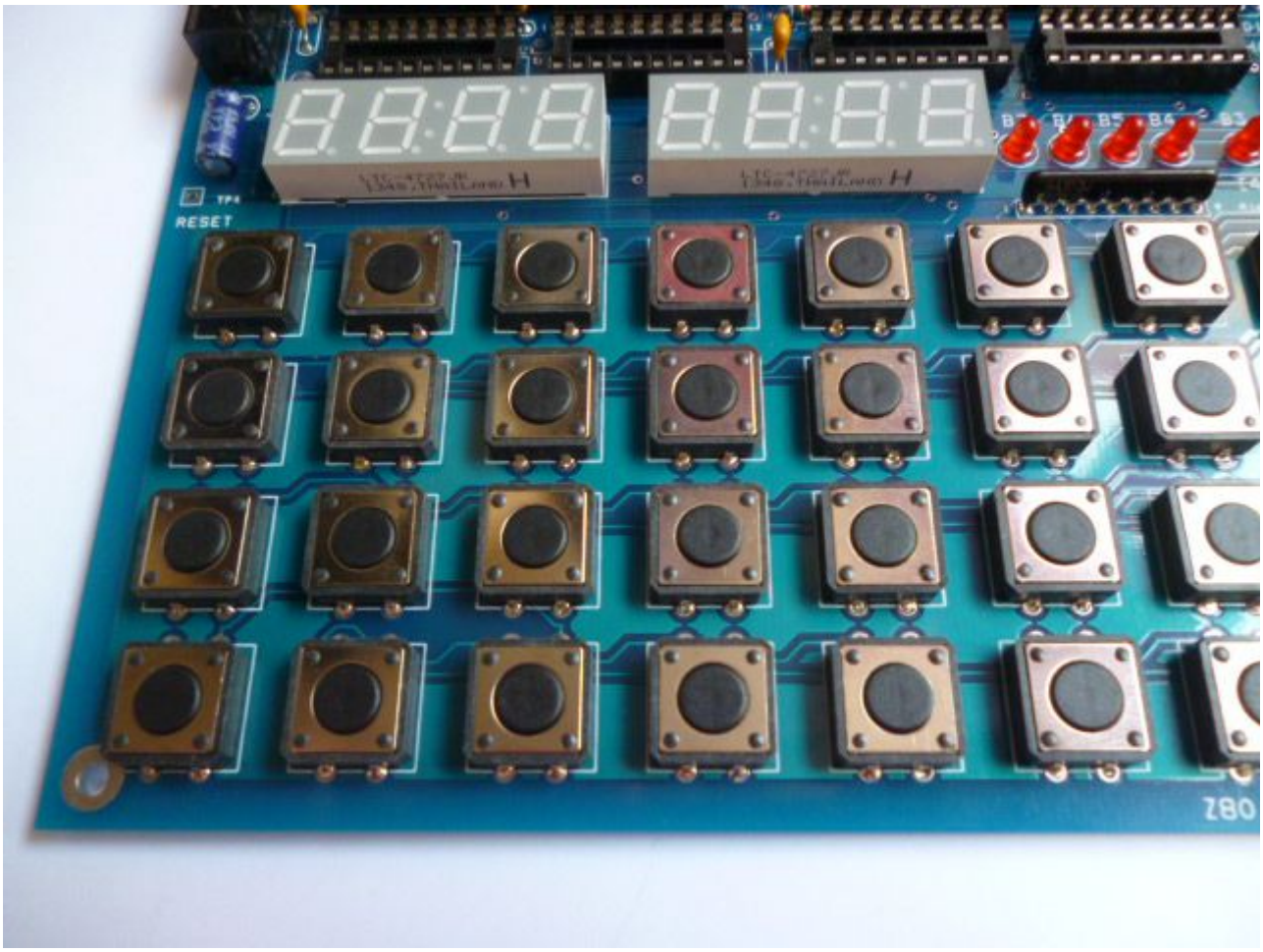

C4 can be placed axially to make it looks good.



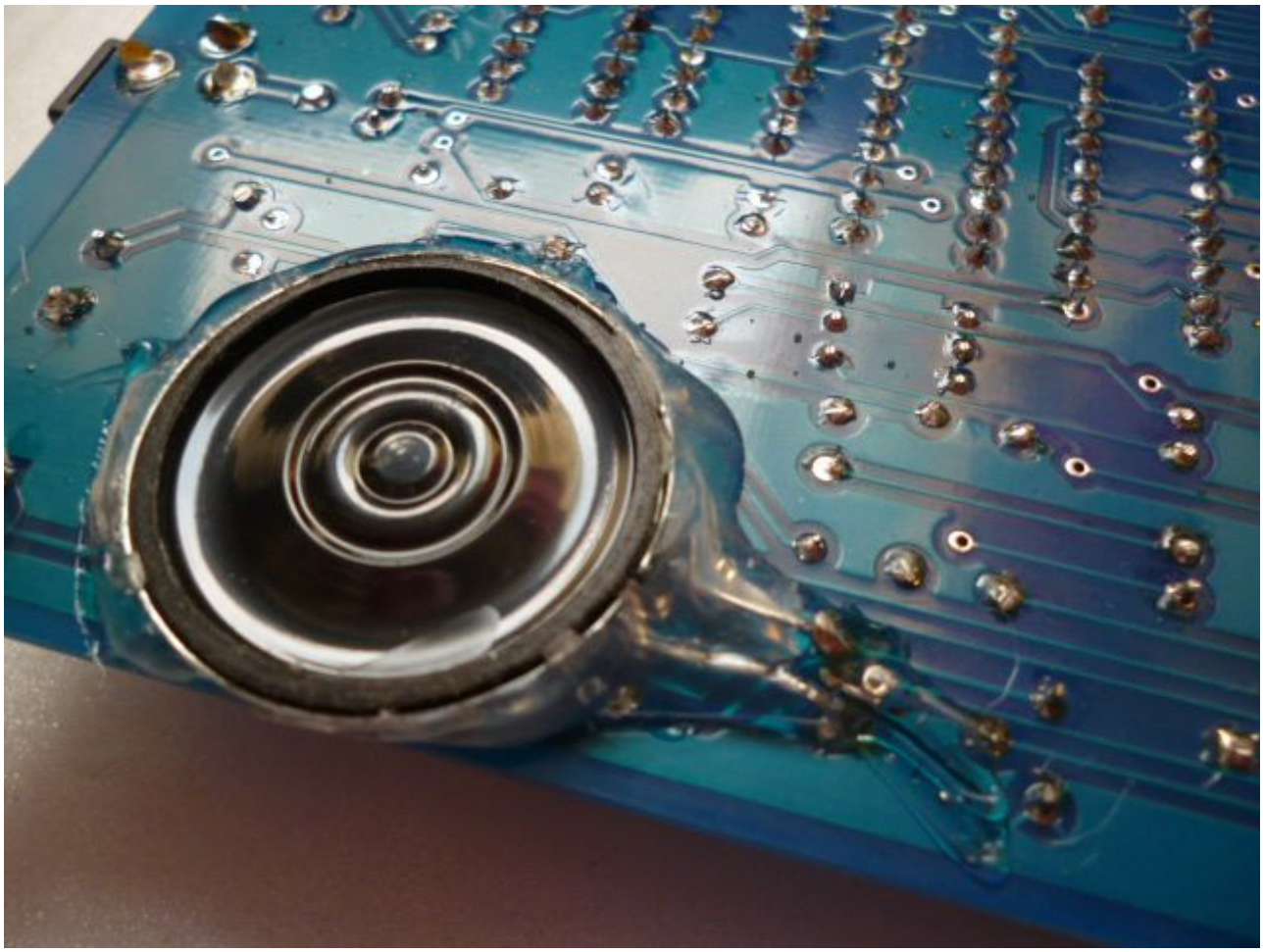
VB1 is DB9 male type connector.

☐ 11. Insert TACT switches, 36 keys. Key must be pressed until locked with clicking sound!

Then solder it. To make sure that all keys are in placed, turn the board back, recheck all keys are the same level! Some key may springs back, press it until click!



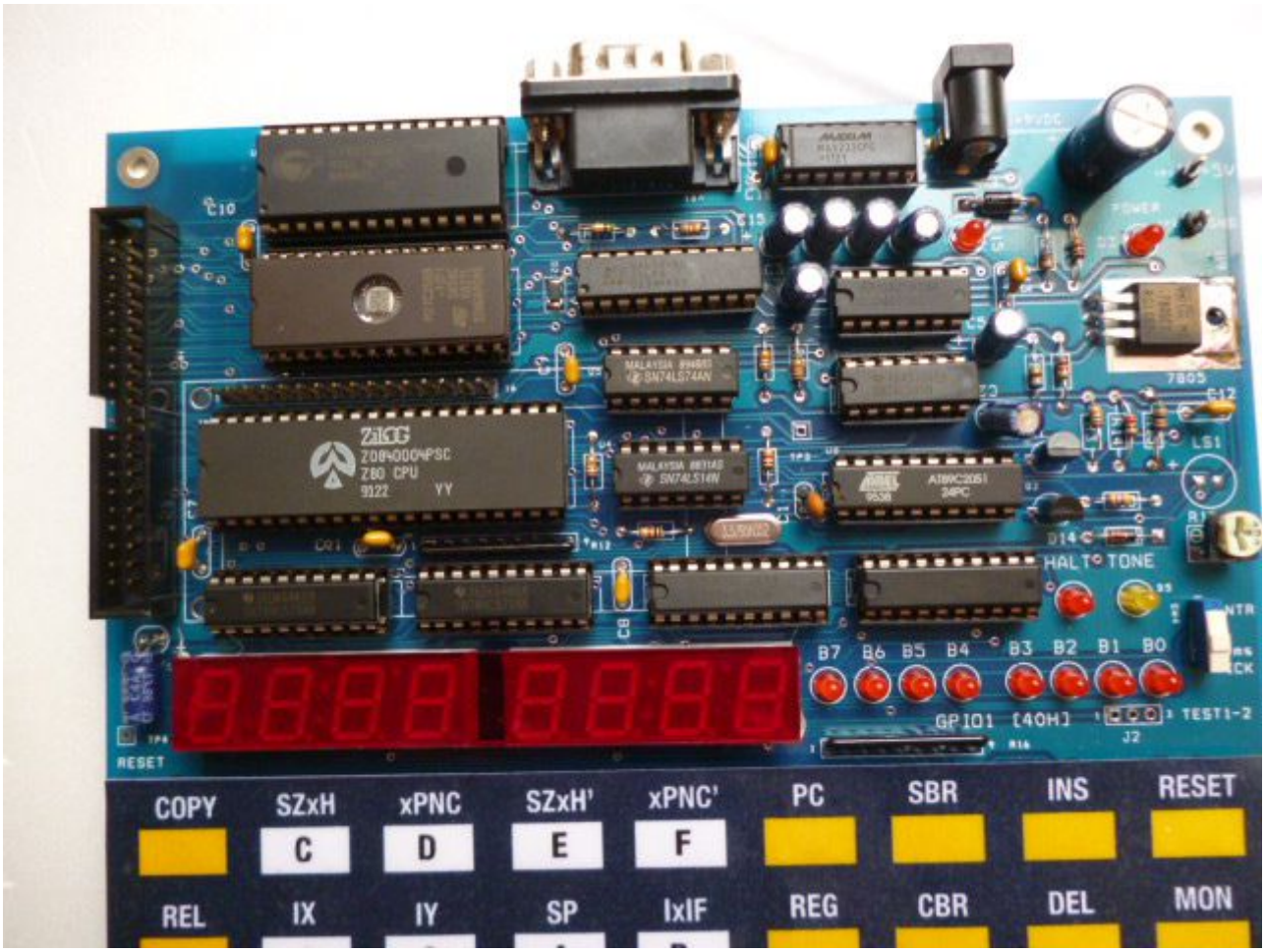
☐ 12. Small speaker can be placed at the bottom layer and have hot glue to fix it.



☐ 13. Before insert the IC chips, check +5V by powering the board with AC adapter.

The DC input accepts from +9VDC to +12VDC. The 7805 needs at least +2V dropped voltage. Use a volt meter check at test points TP1+5V and TP2 for GND pin. If it was nearly +5V, it works fine. Then remove the power.

Insert the IC chips carefully. Recheck the pin position. See the example below picture.



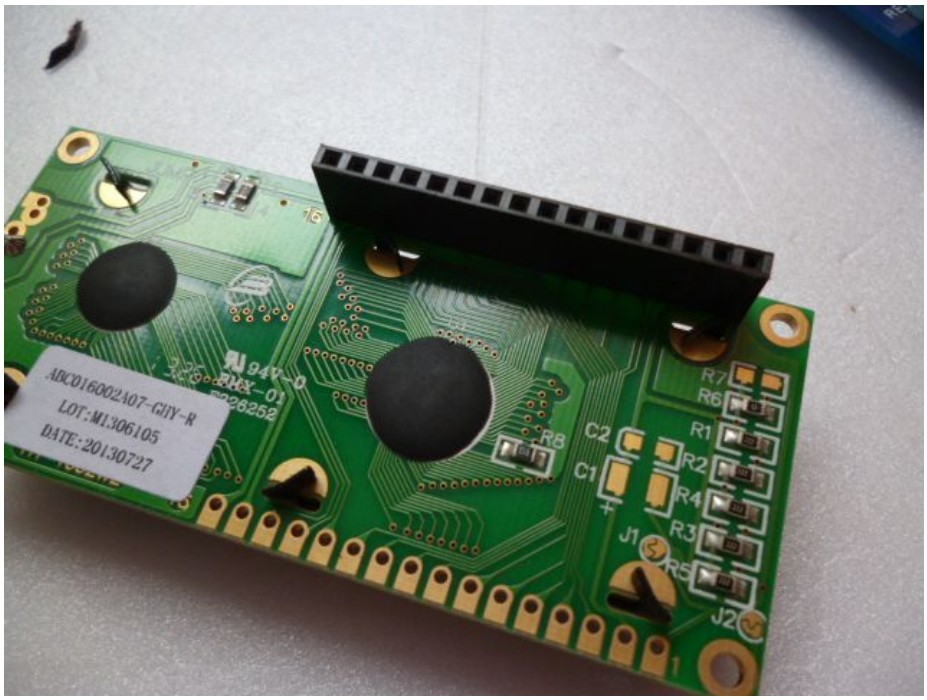
Pin 1 has indicator on the IC chips. Shown Zilog Z80 CPU and monitor EPROM, 27C256.


The quartz window of the EPROM can be opened for showing student to see the silicon chip.

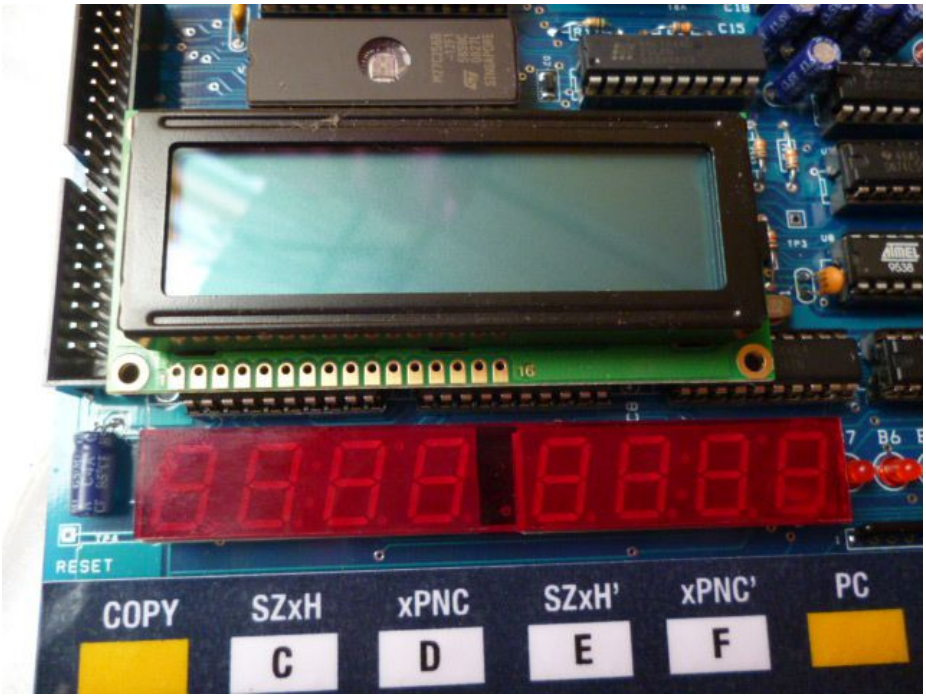
However if you like to close it with small sticker, then you can make it. The normal light or even sunshine can not kick the charge that stored in the floating gate of the FET in the CMOS structure. To kick it, you will need strong UV with wavelength about 253.7 nm.

Read more at http://en.wikipedia.org/wiki/EPROM

Text LCD with 16-pin socket for plugging to the 16-header on the board.

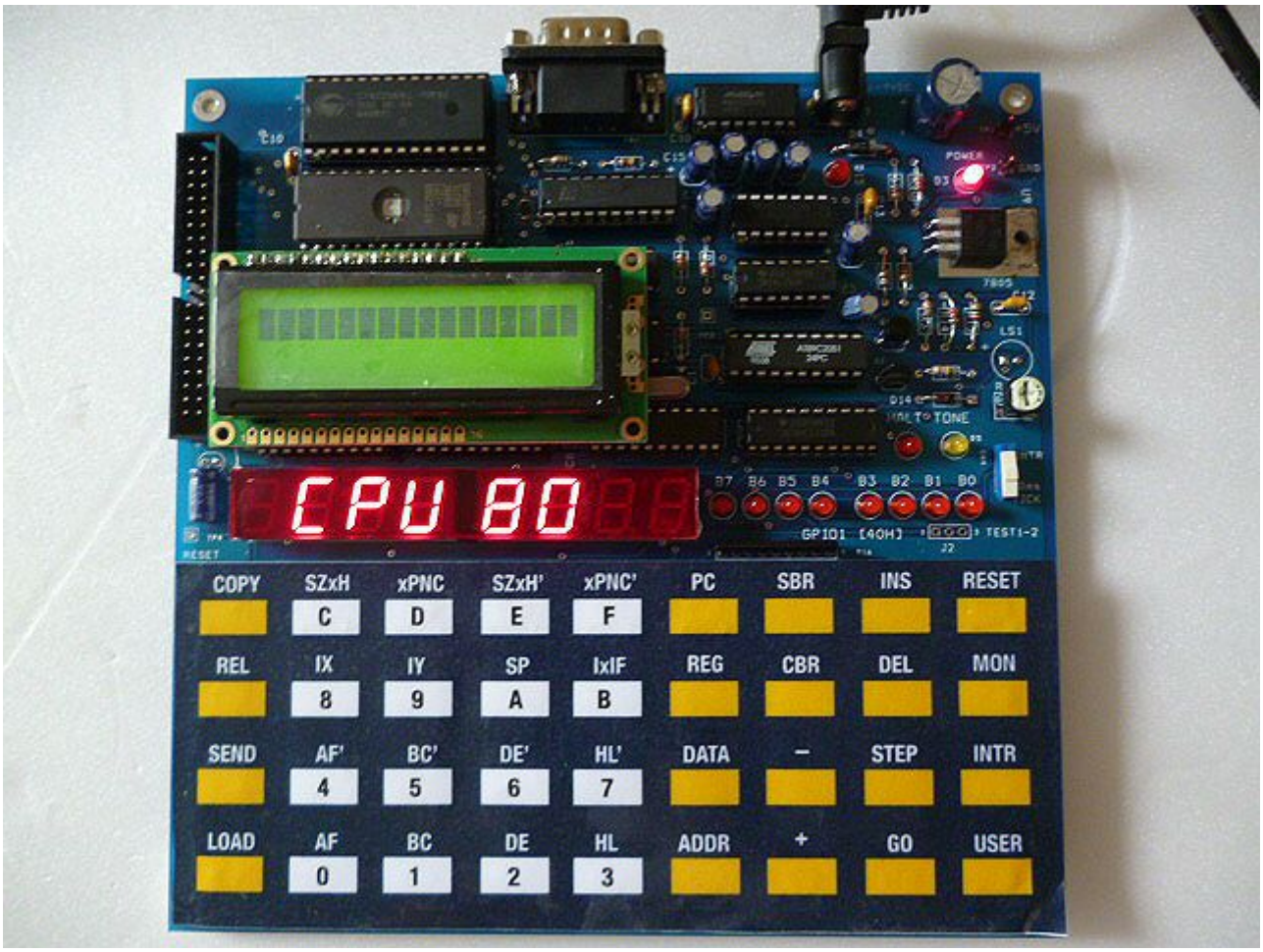



**Remember, plugging/removing the LCD must be done when the board is powered off!**



Your first computer shall look like this when powers it on. For LCD, adjust R18 to make the first line appears.

Building the computer is easy and fun!




Let us test the Z80 to execute our program by entering the Z80 instructions in hex code.

Press key PC, 3E, +, 01,+, D3,+, 40,+ , 76, PC, GO.

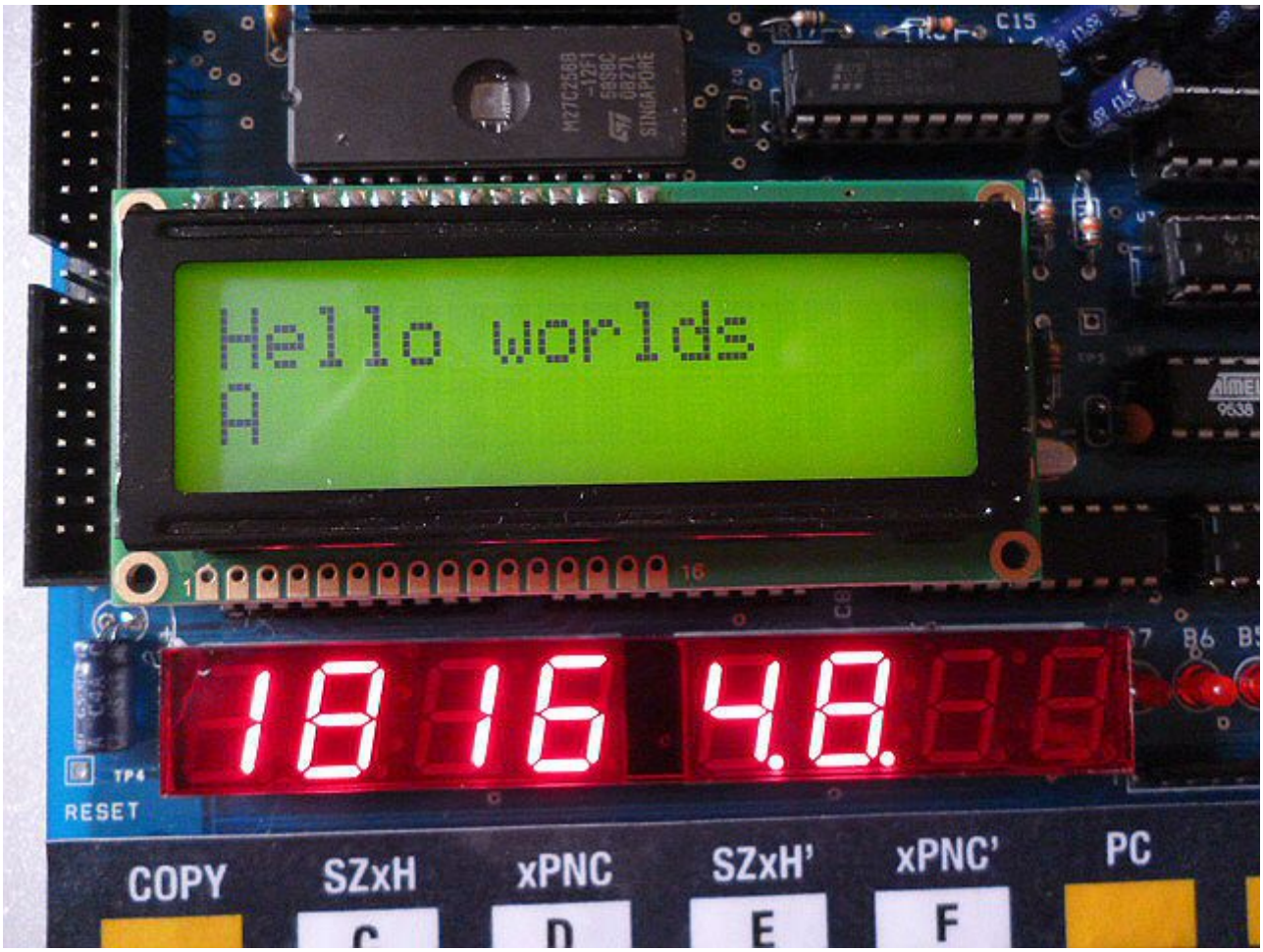
Did you see the binary number 0000 0001 on the GPIO1 LED? We program the Z80 to write the accumulator contents to the 8-bit GPIO1. This brings the internal data to outside realworld!

The program is simple. It has only three instructions. The hex code is only five bytes.

```
1800  3E 01          LD A, 1             ; load accumulator with 1
1802  D3 40          OUT (40H),A         ; write to port 40H
1804  76             HALT                ; Halt the CPU
```

You can change the value to be loaded easily at the 2nd byte located at address 1801. Then see the change when press key GO again.  How easy it is!

Another example of displaying text on LCD. After the LCD initialized, we can use built-in LCD drivers to print text or ASCII letter on screen. We will learn how to use LCD with the Z80 programming lab book.



Before take a rest after completed assembly, let us enter the hex code for running LED.

```
line      addr hex code    label          instruction       comment
----      ----  --------     -----          ------------     ---------------
0001   1800                                .ORG 1800H
0002   1800
0003   1800                 GPIO1          .EQU 40H
0004   1800
0005   1800 3E 01           MAIN           LD A,1
0006   1802
0007   1802 D3 40           LOOP           OUT (GPIO1),A
0008   1804 CD 0C 18                       CALL DELAY
0009   1807 CB 07                          RLC A
0010   1809 C3 02 18                       JP LOOP
0011   180C
0012   180C
0013   180C 11 FF FF        DELAY          LD DE,-1
0014   180F 21 00 10                       LD HL,1000H
0015   1812 19              LOOP2          ADD HL,DE
0016   1813 38 FD                          JR C,LOOP2
0017   1815 C9                             RET
0018   1816
0019   1816                                .END
0020   1816
tasm: Number of errors = 0.
```

The program is simple forever loop writing the accumulator contents to the GPIO1 LED.

Let us entering the hex code with hex key and + key now.

Started at 1800, 3E, 01, D3, 40, CD, 0C, 18,..... until 38, FD and last byte C9. Then press PC, GO. What is happening?

Then take a rest and have your time to learn more how to program the Z80 with the lab book.

Have fun!

## TROUBLESHOOTING

| PROBLEM | CAUSE | ACTION |
|---|---|---|
| No display, no power LED | 1. no power from AC adapter<br>2. wrong polarity jack<br>3.wrong direction of protection diode, D4 | 1. Check AC outlet<br>2. correct the jack polarity, use new adapter<br>3. correct diode direction |
| No display, power LED lit | 1. wrong IC chip insertion<br>2. wrong monitor program<br>3. wrong PLD code<br>4. oscillator is not running<br>5. CPU is not reset | 1. correct chip position<br>2. reprogram the monitor EPROM<br>3. reprogram PLD<br>4. check XTAL soldering, R1 and R2 must be 1k, C1 must be 100pF<br>5.wrong C2 polarity, check TP4 with logic probe, press RESET key will make it logic “low”. |
| No display on GPIO1 LED | 1. wrong direction of D14<br>2. wrong direction of LED | 1. correct the direction<br>2. correct Cathode/Anode pins |
| No beep when press key | 1. no speaker<br>2. wrong position of Q2 | 1. solder the speaker<br>2. correct the position of Q2 |
| Heat up at voltage regulator, U9 7805 | 1. DC input is much higher than +7V | 1. Lower input DC voltage |
| No display on LCD | 1. wrong insertion position | 1. correct the position |
| No back-light on LCD | 1. no R17 installed<br>2. R17 is higher resistant<br>3. LCD module has no back-light<br>4. back-light pins is not the same as schematic | 1. install R17<br>2. use 5-10 Ohms for R17<br>3. use LCD module with back-light<br>4. use LCD module with correct back-light pins |
| Load/Send hex file problem | 1. terminal is not set correct format<br>2. Z80 kit stops loading<br><br>3.RS232 cable is not cross cable. | 1. set terminal for 2400 bit/s, 8-data bit, no parity, one stop bit, no flow control.<br>2. File is not Intel HEX file. Ensure the file is Intel HEX file.<br>3. check the cable, it must be |

| | | |
|---|---|---|
| | 4. Send hex file displays unreadable letters | cross cable<br><br>4. correct terminal settings. |
| 10ms tick LED, D15 is not blinking | 1. wrong polarity of D15<br>2. AT89C2051 is not programmed | 1. correct the polarity<br>2. reprogram the microcontroller |
| Some key has no response | 1.not yet solder it | 1. check and solder it |

**PARTS LIST**

**Semiconductors**

U1 Z80 40-pin DIP microprocessor
U2 27C64 or 27C256 EPROM
U3 HM62256B 32kB SRAM
U4 74LS14 inverter
U5 74HC74 Dual D-type flip-flop
U6 GAL16V8B programmable logic device
U7 74LS90 decade counter
U8 AT89C2051 20-pin microcontroller
U9 LM7805/TO voltage regulator
U11,U10 LTC-4727JR 4-digit 7-segment LED
U12,U14,U16 74HC573 8-bit latch
U13 CD4001 quad NOR gate
U15 74HC541 tri-state buffer
U17 MAX232A rs232 converter
D1,D6,D7,D8,D9,D10,D11, 3mm LED
D12,D13,D15
D2 TVS5V_SOD123 transient voltage suppressor
D4 1N4007 rectifying diode
D14 1N5226A +3.3V zener 500mW
Q2,Q1 BC557 PNP transistor
D3 POWER 3mm LED
D5 TONE 3mm LED

**Resistors (all resistors are 1/8W +/-5%)**

R1,R2,R9 1k
R3,R4,R5,R6,R8,R18 10K
R7,R19 680 Ohms
R10 2k
R16,R12 10k RESISTOR SIP 9
R13 4.7k
R14 330 Ohms
R17,R15 10 Ohms

**Capacitors**

C1 100pF ceramic
C2,C3,C15,C16,C17 10uF electrolytic
C4 100uF electrolytic
C5 10uF 16V electrolytic
C6 1000uF 16V electrolytic
C7,C8,C9,C10,C11 0.1uF multi layer
C12,C13 0.1uF multilayer
C14 10uF 10V electrolytic
C18 0.1uF multilayer

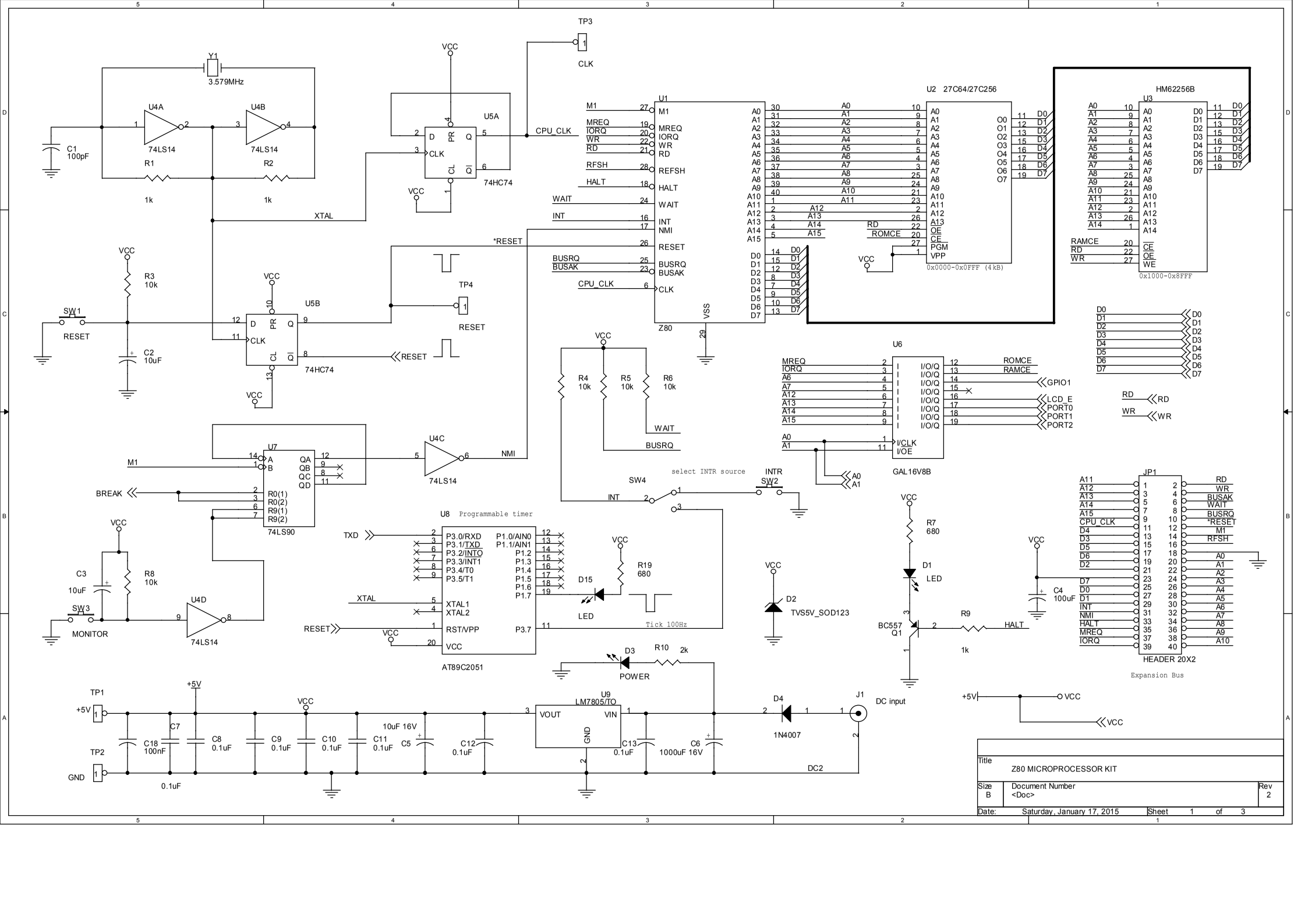
**Additional parts**

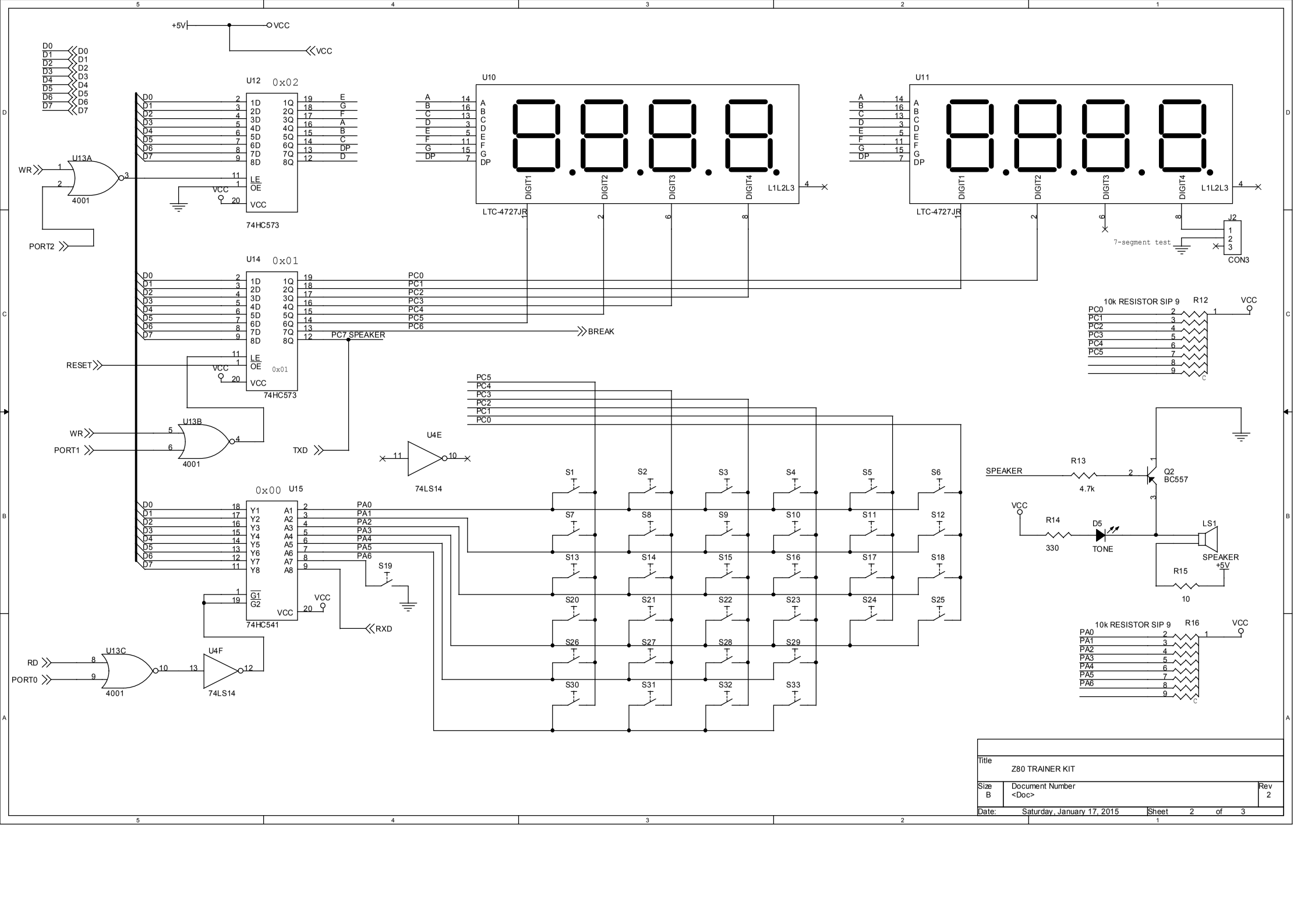
JP1 HEADER 20X2
JR1 CONN RECT 16 pins
J1 DC input JACK
J2 HEADER 3 pins
LS1 8 Ohms SPEAKER

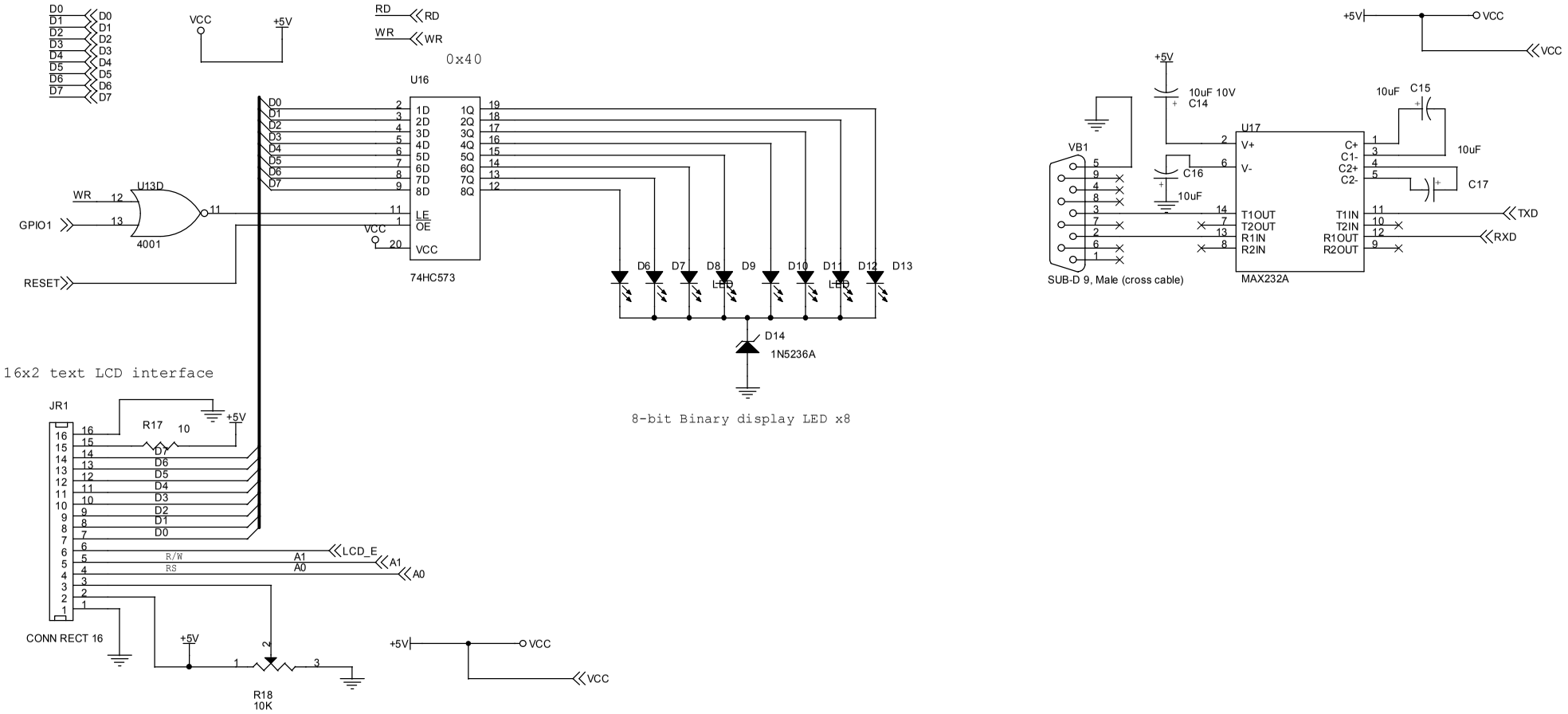
SW1 RESET 12mm tact switch
SW2 INTR 12mm tact switch
SW3 MONITOR 12mm tact switch
SW4 slide switch-SPDT

S1,S2,S3,S4,S5,S6,S7,S8, 12mm TACT switch
S9,S10,S11,S12,S13,S14,
S15,S16,S17,S18,S19,S20,
S21,S22,S23,S24,S25,S26,
S27,S28,S29,S30,S31,S32,
S33

VB1 DB 9, Male connector
Y1 XTAL 3.579MHz
PCB double side plate through hole
LED cover Clear RED color acrylic plastic
Keyboard sticker printable SVG file


TP3
CLK
VCC
Y1
3.579MHz
U4A
U4B
74LS14
74LS14
C1
100pF
R1
R2
1k
1k
XTAL
U5A
74HC74
CPU_CLK
U1
Z80
M1
MREQ
IORQ
WR
RD
REFSH
HALT
WAIT
INT
NMI
RESET
BUSRQ
BUSAK
CPU_CLK
*RESET
U2 27C64/27C256
0x0000-0x0FFF (4kB)
HM62256B
U3
0x1000-0x8FFF
RAMCE
ROMCE
R3
10k
SW1
RESET
C2
10uF
U5B
74HC74
TP4
RESET
R4
10k
R5
10k
R6
10k
WAIT
BUSRQ
U6
GAL16V8B
ROMCE
RAMCE
GPIO1
LCD_E
PORT0
PORT1
PORT2
U7
74LS90
BREAK
U4C
74LS14
NMI
SW4
INT
select INTR source
INTR
SW2
U8 Programmable timer
AT89C2051
TXD
XTAL
RESET
R19
680
D15
LED
Tick 100Hz
C3
10uF
R8
10k
SW3
MONITOR
U4D
74LS14
D2
TVS5V_SOD123
R7
680
D1
LED
BC557
Q1
R9
1k
HALT
C4
100uF
JP1
HEADER 20X2
Expansion Bus
D3
POWER
R10
2k
TP1
TP2
+5V
GND
C18
100nF
C7
0.1uF
C8
0.1uF
C9
0.1uF
C10
0.1uF
C11
0.1uF
C5
10uF 16V
C12
0.1uF
U9
LM7805/TO
VOUT
VIN
GND
C13
0.1uF
C6
1000uF 16V
D4
1N4007
J1
DC input
DC2
Title
Z80 MICROPROCESSOR KIT
Size
B
Document Number
<Doc>
Rev
2
Date:
Saturday, January 17, 2015
Sheet
1
of
3


+5V
VCC
D0
D1
D2
D3
D4
D5
D6
D7
U12
0x02
74HC573
U10
U11
LTC-4727JR
DIGIT1
DIGIT2
DIGIT3
DIGIT4
L1L2L3
J2
7-segment test
CON3
U13A
4001
WR
PORT2
U14
0x01
PC0
PC1
PC2
PC3
PC4
PC5
PC6
PC7 SPEAKER
BREAK
RESET
10k RESISTOR SIP 9
R12
U13B
PORT1
TXD
U4E
74LS14
U15
0x00
74HC541
PA0
PA1
PA2
PA3
PA4
PA5
PA6
S1
S2
S3
S4
S5
S6
S7
S8
S9
S10
S11
S12
S13
S14
S15
S16
S17
S18
S19
S20
S21
S22
S23
S24
S25
S26
S27
S28
S29
S30
S31
S32
S33
RXD
U13C
U4F
RD
PORT0
SPEAKER
R13
4.7k
Q2
BC557
R14
330
D5
TONE
LS1
SPEAKER
+5V
R15
10
R16
Title
Z80 TRAINER KIT
Size
B
Document Number
<Doc>
Rev
2
Date:
Saturday, January 17, 2015
Sheet
2
of
3




| | | |
|---|---|---|
| Title | Z80 TRAINER KIT | |
| Size B | Document Number <Doc> | Rev 1 |
| Date: Saturday, January 17, 2015 | Sheet 3 of 3 | |

# NOTE