

PRACTICE EXERCISES OF THE MICROPROCESSORS & MICROCONTROLLERS

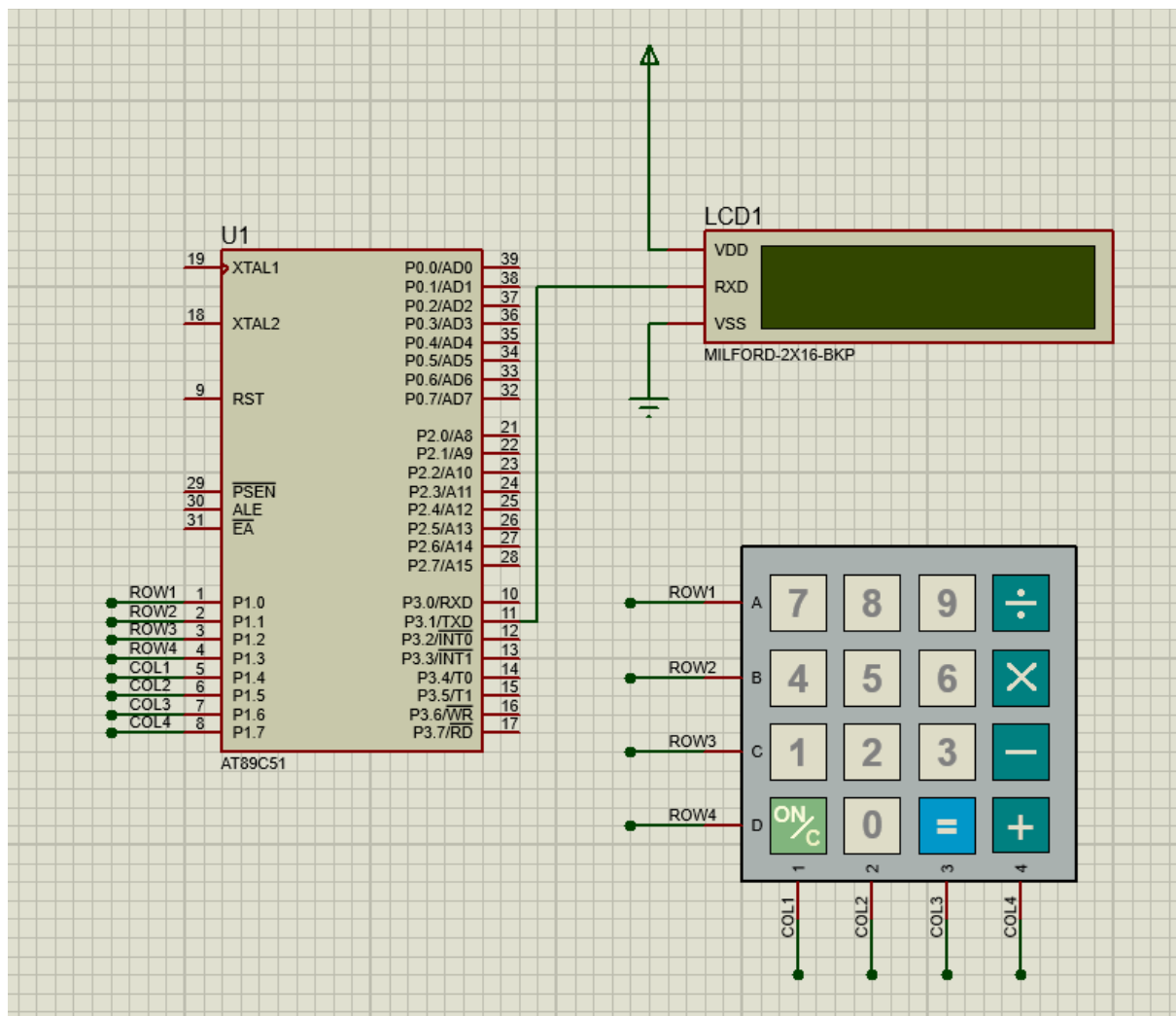
Instructor: The Tung Than

Student's name: Trần Lê Minh Đăng

Student code: 21520684

PRACTICE REPORT NO 4 USING UART

I. Design result (screenshot and pasted in the report).

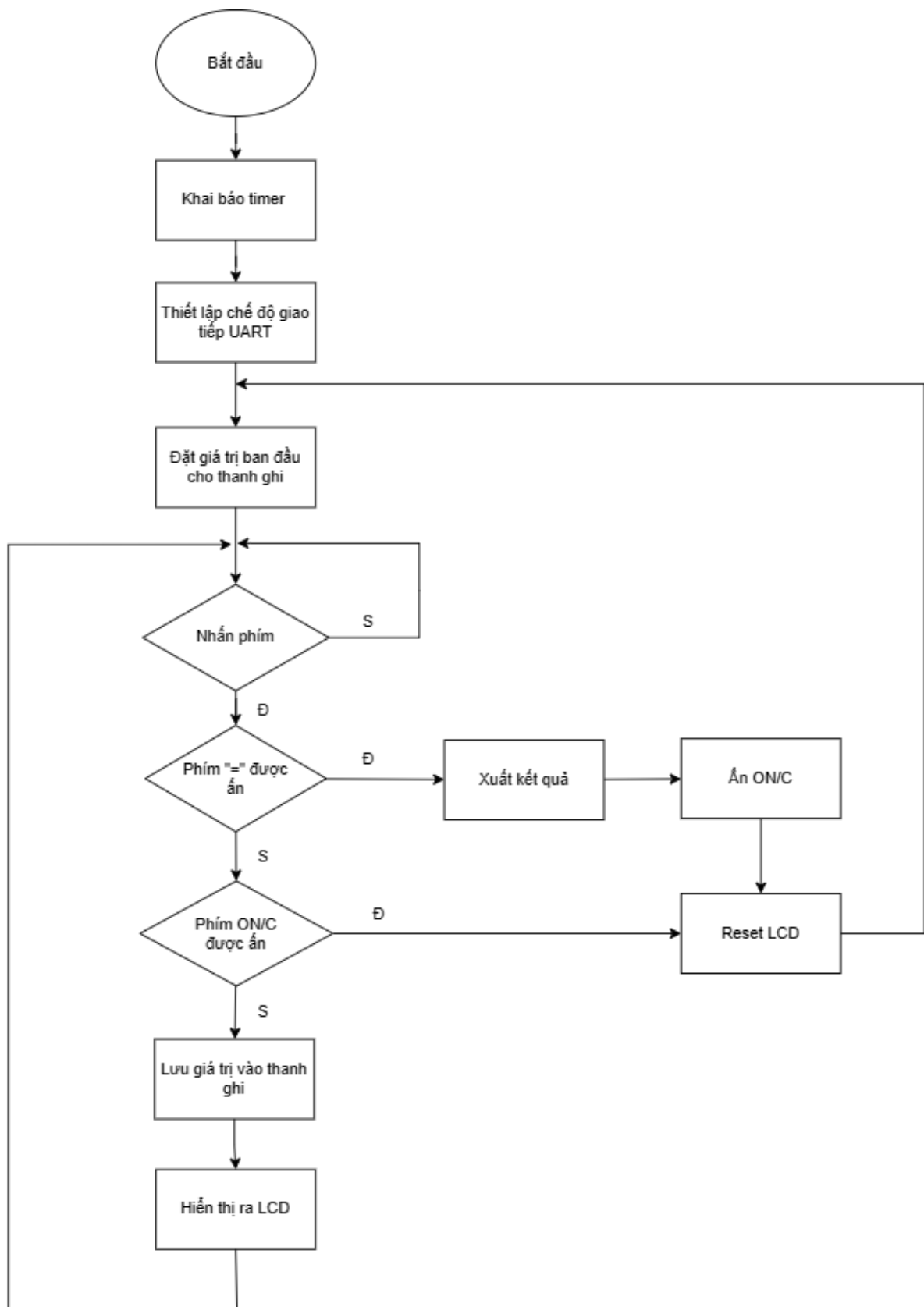


- II. Explain the operating principle of the effects, accompanied by a video (send a Google Drive link) to demonstrate the circuit operation in case the instructor cannot run the design file.

-Link:

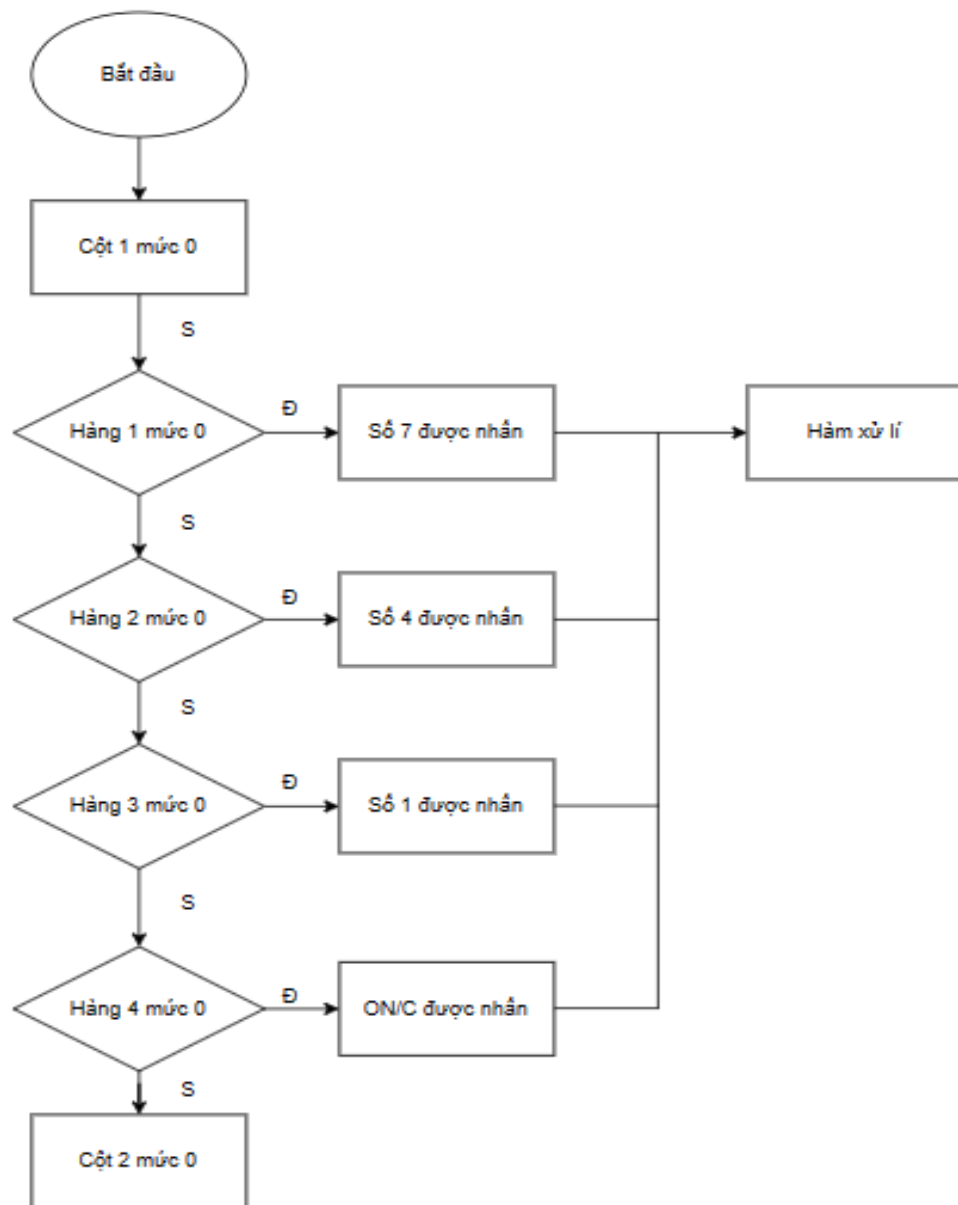
<https://drive.google.com/drive/folders/1nZo3a-TGXKSBxQy2g17jKebSZM3vmhyb?usp=sharing>

- Giải thuật chương trình chính



- Giải thuật quét phím

+ Lưu đồ thuật toán quét phím theo cột 1. Các cột khác tương tự



Code	Giải thích
ORG 0H	;Bắt đầu chương trình
MOV TMOD,#21H	;Chọn timer 1 chế độ 2, timer 0 chế độ 1
MOV TH1,#0FDH	;Nạp TH1 = 0xFD
MOV SCON,#50H	;Nạp thanh ghi SCON, thiết lập chế độ 1
SETB TR1	;Khởi động timer 1
//////////////////INPUT////////////////////	
INPUT:	;Hàm input khởi tạo giá trị ban đầu cho các
MOV R3,#-1	;thanh ghi chứa các số và kí tự nhập từ bàn
MOV R4,#"-1"	;phím.Trong đó R3, R5 chứa 2 số, R4 chứa
MOV R5,#-1	;toán tử
//////////////////MAIN////////////////////	
MAIN:	;Hàm main
MOV R1,#"-1"	;R1 = "-1". R1 chứa ký tự khi nhập phím
MOV P1,#0FFH	;P1 = 0xFF
CALL KEY_PRESS	;Gọi hàm KEY_PRESS
CJNE R1,#"-1",LOOP	;R1 khác -1 thì nhảy tới LOOP
JMP MAIN	;Nhảy về hàm MAIN
LOOP:	;Hàm LOOP
CJNE R3,#-1,OP	;R3 khác -1 thì nhảy tới OP
NUM_1:	;Hàm num_1 để chuyển số thứ nhất sau khi
MOV A,R2	nhập từ R2 sang R3
MOV R3,A	
JMP DISPLAY	
OP:	;Hàm OP để chuyển kí tự toán tử từ R1 sang
CJNE R4,#"-1",NUM_2	R4
MOV A,R1	
MOV R4,A	
JMP DISPLAY	
NUM_2:	;Hàm NUM_2 để chuyển số thứ hai từ R2

CJNE R5,#-1,DISPLAY	
MOV A,R2	sang R5
MOV R5,A	
JMP DISPLAY	
DISPLAY:	;Hàm DISPLAY để hiển thị ra màn hình LCD
MOV SBUF,R1	;Truyền R1 vào SBUF
CALL HIENTHI	;Gọi hàm HIENTHI
CALL DELAY	;Gọi hàm DELAY
JMP MAIN	;Quay lại hàm MAIN
////////////////QUET PHIM////////////////	
KEY_PRESS:	;Hàm nhấn phím
MOV P1,#0EFH	;P1 = 0xEF
MOV A,P1	;P1 = A
ANL A,#0EFH	;A = A and 0xEF
CJNE A,#0EFH,COL1	;A khác 0xEF nhảy tới COL1
MOV P1,#0DFH	
MOV A,P1	
ANL A,#0DFH	
CJNE A,#0DFH,COL2	
MOV P1,#0BFH	
MOV A,P1	
ANL A,#0BFH	
CJNE A,#0BFH,COL3	
MOV P1,#7FH	
MOV A,P1	
ANL A,#7FH	
CJNE A,#7FH,COL4	
RET	
////////////////COL1////////////////	
COL1:	;Hàm xử lí khi có phím ở cột 1 được nhấn
JNB P1.0,SO_7	;P1.0 khác 1 nhảy tới SO_7

JNB P1.1,SO_4	;P1.1 khác 1 nhảy tới SO_4
JNB P1.2,SO_1	;P1.2 khác 1 nhảy tới SO_1
JNB P1.3,CLEAR	;P1.3 khác 1 nhảy tới CLEAR
SO_7:	;Hàm xử lý khi số 7 được nhấn
MOV R1,#"7"	;R1 = "7"
MOV R2,#7	;R2 = 7
JMP LOOP	;Nhảy về LOOP
SO_4:	
MOV R1,#"4"	
MOV R2,#4	
JMP LOOP	
SO_1:	
MOV R1,#"1"	
MOV R2,#1	
JMP LOOP	
CLEAR:	;Hàm xử lý khi nhấn phím ON/C
MOV SBUF,#254	; 4 dòng code có tác dụng reset LCD
CALL HIENHI	;
MOV SBUF,#1	;
CALL HIENHI	;
JMP INPUT	;Nhảy tới INPUT
RET	;Thoát hàm
//////////COL2//////////	
COL2:	;Hàm xử lý khi có phím ở cột 2 được nhấn
JNB P1.0,SO_8	
JNB P1.1,SO_5	
JNB P1.2,SO_2	
JNB P1.3,SO_0	
SO_8:	
MOV R1,#"8"	
MOV R2,#8	

JMP LOOP

SO_5:

MOV R1,#"5"

MOV R2,#5

JMP LOOP

SO_2:

MOV R1,#"2"

MOV R2,#2

JMP LOOP

SO_0:

MOV R1,#"0"

MOV R2,#0

JMP LOOP

RET

//////////COL3//////////

COL3:

JNB P1.0,SO_9

JNB P1.1,SO_6

JNB P1.2,SO_3

JNB P1.3,EQUAL

SO_9:

MOV R1,#"9"

MOV R2,#9

JMP LOOP

SO_6:

MOV R1,#"6"

MOV R2,#6

JMP LOOP

SO_3:

MOV R1,#"3"

MOV R2,#3

;Hàm xử lí khi có phím ở cột 3 được nhấn

<pre> JMP LOOP EQUAL: MOV R1,#"=" MOV SBUF,R1 CALL HIENTHI CALL DELAY CALL OUT RET ////////COL4////////// COL4: JNB P1.0,CHIA JNB P1.1,NHAN JNB P1.2,TRU JNB P1.3,CONG CHIA: MOV R1,#"/" JMP LOOP NHAN: MOV R1,#"x" JMP LOOP TRU: MOV R1,#"-" JMP LOOP CONG: MOV R1,#"+" JMP LOOP RET //////////TINH KET QUA////////// OUT: MOV A,R3 </pre>	<pre> ;Hàm xử lí khi ấn phím "=" ; 4 dòng code có tác dụng hiển thị "=" ra LCD ; ; ; ;Gọi hàm OUT ;Hàm xử lí khi có phím ở cột 4 được nhấn ;Hàm tính kết quả và xuất ra LCD ;A = R3 </pre>
---	--

CJNE R4,#"+",TRU_1	;R4 khác “+” thì nhảy tới TRU_1
CONG_1:	;Hàm tính tổng 2 số
ADD A,R5	;A = A + R5 = R3 + R5
JMP DIS	;Nhảy tới DIS
TRU_1:	;Hàm tính hiệu 2 số
CJNE R4,#"-",NHAN_1	;R4 khác “ - ” thì nhảy tới NHAN_1
CLR C	;C = 0
SUBB A,R5	;A = A – R5 – C = A – R5 = R3 – R5
JMP DIS	;Nhảy tới DIS
NHAN_1:	;Hàm tính tích 2 số
CJNE R4,#"x",CHIA_1	;R4 khác “x” thì nhảy tới CHIA_1
MOV B,R5	;B = R5
MUL AB	;A = A x B
JMP DIS	;Nhảy tới DIS
CHIA_1:	;Hàm tính thương 2 số
MOV B,R5	;B = R5
DIV AB	;A = thương A/B, B = phần dư A/B
JMP DIS	;Nhảy tới DIS
DIS:	;Hàm hiển thị kết quả ra LCD
MOV B,#10	;Tách giá trị ở thanh ghi A thành 2 chữ số,
DIV AB	; A = hàng chục, B = đơn vị
ADD A,#30H	;Chuyển kiểu số sang kí tự
MOV R6,A	;R6 = A
MOV A,B	;A = B
ADD A,#30H	;Chuyển kiểu số sang kí tự
MOV R7,A	;R7 = A
MOV SBUF,R6	;Hiển thị ra LCD
CALL HIENTHI	;
MOV SBUF,R7	;
CALL HIENTHI	;
JMP MAIN	;Quay lại hàm MAIN

RET	;
////////HIENTHI////	
HIENTHI:	
JNB TI,\$;Lặp cho đến khi TI = 1
CLR TI	;Xóa bit TI
RET	
////////DELAY////////	
DELAY:	;Hàm DELAY
MOV R1,#5	;Lặp hàm DELAY 5 lần
DELAY1:	
MOV TH0,#HIGH(-50000)	;Nạp TH0
MOV TL0,#LOW(-50000)	;Nạp TL0
SETB TR0	;Bật TR0
JNB TF0,\$;Lặp đến khi TF0 = 1
CLR TF0	;Xóa TF0
CLR TR0	;Xóa TR0
DJNZ R1,DELAY1	
RET	
////////END////////	
END	;Kết thúc chương trình