Báo cáo vi xử lý :

Điều khiển đèn giao thông và hiện thị số trên màn hình led 7 đoạn 2 chữ số bằng vi điều khiển 8051- AT89C51

1. Tìm hiểu về vi điều khiển AT89C51 thuộc họ 8051
   1. giới thiệu
   2. các chân trong AT89C51(kèm ảnh)
   3. Tổ chức bộ nhớ
   4. Bộ nhớ ngoài
   5. Giao tiếp ngoại vi (với led 7 đoạn, )
2. ứng dụng 8051 điều khiển đèn giao thông
   1. Tổng quan

* Đề tài
* Các thành phần chính: mạch điều khiển trung tâm dùng vđk AT89C51, mạch hiển thị LED 7 đoạn, trạng thái đèn; respack – 8
* Nguyên lý hoạt động: AT89C51 đưa dữ liệu đến các LED xanh, đỏ, vàng đề điều khiển các LED đóng mở.
  1. phần cứng
* mạch hiển thị đếm ngược LED 7 đoạn
* mạch hiển thị led đơn
  1. Chương trình Assembly (đang viết comment )

<https://www.academia.edu/6824259/B%E1%BB%98_GIAO_TH%C3%94NG_V%E1%BA%ACN_T%E1%BA%A2I_TR%C6%AF%E1%BB%9CNG_%C4%90%E1%BA%A0I_H%E1%BB%8CC_GIAO_TH%C3%94NG_V%E1%BA%ACN_T%E1%BA%A2I_TP_HCM_%C4%90%E1%BB%81_t%C3%A0i_%E1%BB%A8NG_D%E1%BB%A4NG_8051_%C4%90I%E1%BB%80U_KHI%E1%BB%82N_%C4%90%C3%88N_GIAO_TH%C3%94NG>

<https://sites.google.com/site/t2vietdtk/Downhome/gi/8051/vdk/cac-chan-cong-vao-ra>

(Dựa trên vid ) <https://www.youtube.com/watch?v=FoOVjXVUcOA>

Powerpoint:

1. Giới thiệu qua về AT89C51(Ảnh, giới thiệu tổng quan)
2. Hình ảnh mạch điều khiển đèn giao thông

* Nguyên lý hoạt động
* Các thành phần và chức năng của nó
* Chương trình Assembly

Chương trình Assembly:

;====================================================================

; Main.asm file generated by New Project wizard

;

; Created: Thu Nov 26 2020

; Processor: AT89C51

; Compiler: ASEM-51 (Proteus)

;====================================================================

$NOMOD51

$INCLUDE (8051.MCU)

;====================================================================

; DEFINITIONS

;====================================================================

;====================================================================

; VARIABLES

;====================================================================

;====================================================================

; RESET and INTERRUPT VECTORS

;====================================================================

; Reset Vector

org 0000h

jmp Start

;====================================================================

; CODE SEGMENT

;====================================================================

org 0300h

TBL: DB 0C0H,0F9H,0A4H,0B0H,99H,92H,82H,0F8H,80H,90H ;7seg data for comm. anode type

; number: 0 1 2 3 4 5 6 7 8 9

;address : 300 301 302 303 304 305 306 307 308 309

ORG 30H

Start:

; Write your code here

MOV P2,#00H ;tat den led

MOV P3,#00H ; tat den led

ACALL FRONT

MOV DPTR,#TBL ; chuyen du lieu tu TBL vao DPTR

; DPTR data pointer

CLR A

MOV 40H,#10 ; gan gia tri 10 cho d/c 40H

MOV 43H,#10 ; gan gia tri 10 cho d/c 43H

MOV 46H,#20 ; gan gia tri 20 cho d/c 46H

MOV 49H,#20 ; gan gia tri 20 cho d/c 49H

MOV R0,#35 ; gan gia tri 35 cho thanh ghi R0

MOV R6,#30 ; gan gia tri 30 cho thanh ghi R6

MOV R7,#40 ; gan gia tri 40 cho thanh ghi R7

X1: MOV A,40H ; gan gia tri o o nho 40H den thanh ghi A

MOV B,#10 ; gan gia tri 10 cho thanh ghi B

DIV AB ; lay gia tri trong A chia cho gia tri trong B

; A chia cho 10, phan thuong ghi trong A,

; phan du ghi trong B

MOV 41H,A ; gan gia tri trong A vao o nho 41H

; gtri trong 41H =1

MOV 42H,B ; gan gia tri trong B vao o nho 42H

; gtri trong 42H =0

A1: SETB P3.0 ; cong P3.0 len bit 1

CLR P3.1 ; chan cong P3.1 thanh bit 0

MOV A,41H ; gan gia tri o o 41H vao A

MOVC A,@A+DPTR ; gan gia tri o o nho @A+DPTR vao thanh ghi A

; d/c = 1+300=301 -> A = 0F9H

MOV P2,A ; gan gtri cua A cho cong P2

; p2 co gtri = 0F9H

ACALL DELAY

MOV P3,#00H ; dat 8 chan cua cong P3 xuong muc thap

; tat het led

SETB P3.1 ; cho sang led tren chan P3.1

CLR P3.0 ; bit tren chan P3.0 = 0

MOV A,42H ; gan gia tri o o 42H vao A

; A= 0

MOVC A,@A+DPTR ; d/c = 0 + 300 = 300

; A = 0C0H

MOV P2,A ; P2 = 0C0H;

ACALL DELAY ; goi den lenh tre

MOV P3,#00H ; tat den led

SJMP X3 ; nhay den nhan X3

X2: SJMP X1 ; nhay den nhan X1

X3: MOV A,43H ; A = 10

MOV B,#10 ; B =10

DIV AB ; A/B, phan thuong A= 1, phan du B= 0

MOV 44H,A ; o nho 44H = 1

MOV 45H,B ; o nho 45H = 0

SETB P3.2 ; chan P3.2 co bit = 1

CLR P3.3 ; chan P3.3 co bit = 0

MOV A,44H ; A= 1

MOVC A,@A+DPTR ; d/c = 1 + 300 = 301

; A = 0F9H

MOV P2,A ; P2 = 0F9H

ACALL DELAY ; goi den lenh tre

MOV P3,#00H ; tat den led

SETB P3.3 ; chan P3.3 co bit =1

CLR P3.2 ; chan P3.2 co bit =0

MOV A,45H ; A = 0

MOVC A,@A+DPTR ; d/c = 0+ 300 = 300

; A = 0C0H

MOV P2,A; P2 = 0C0H

ACALL DELAY ; goi lenh tre

MOV P3,#00H ; tat den led

MOV A,46H ; A = 20

MOV B,#10 ; B =10

DIV AB ; A = 2, B =0

MOV 47H,A ; o 47H = 2

MOV 48H,B ; o 48H = 0

SETB P3.4 ; chan P3.4 bit =1

CLR P3.5 ; chan P3.5 bit = 0

MOV A,47H ; A = 2

MOVC A,@A+DPTR ; A = 0A4H

MOV P2,A ; P2 =0A4H

ACALL DELAY ; goi lenh tre

MOV P3,#00H ; tat led

SETB P3.5 ; chan P3.5 bit =1

CLR P3.4 ; chan P3.4 bit = 0

MOV A,48H ; A = 0

MOVC A,@A+DPTR ; A = 0C0H

MOV P2,A ; P2 = 0C0H

ACALL DELAY ; goi lenh tre

MOV P3,#00H ; tat led

MOV A,49H ; A = 20

MOV B,#10 ; B = 10

DIV AB ; A =2 , B =0

MOV 50H,A ; o 50H =2

MOV 51H,B ; o 51H =0

SETB P3.6 ; chan P3.6 bit =1

CLR P3.7 ; chan P3.7 bit = 0

MOV A,50H ; A = 2

MOVC A,@A+DPTR ; A = 0A4H

MOV P2,A ; P2 = 0A4H

ACALL DELAY ; goi lenh tre

MOV P3,#00H ; tat led

SETB P3.7 ; chan P3.7 bit =1

CLR P3.6 ; chan P3.6 bit =0

MOV A,51H ; A= 0

MOVC A,@A+DPTR; A= 0C0H

MOV P2,A ; P2 = 0C0H

ACALL DELAY ; goi lenh tre

MOV P3,#00H ; tat led

DJNZ R0,X2 ; lap vong tu nhan X2 den khi R0 = 0

MOV R0,#35 ; R0 = 35

DJNZ 40H,Q1 ; lap vong tu nhan Q1 den khi gtri cua o 40H = 0

MOV 40H,#20 ; gia tri o 40H =20

Q1: DJNZ 43H,Q2 ; lap vong tu nhan Q1 den khi gtri cua o 43H = 0

MOV 43H,#10 ; gtri o 43H = 10

ACALL RIGHT

Q2: DJNZ 46H,Q3 ; lap vong tu nhan Q3 den khi gtri cua o 46 = 0

MOV 43H,#20

MOV 46H,#10

Q3: DJNZ 49H,Q4 ; lap vong tu nhan Q4

MOV 49H,#10 ;

ACALL BACK

Q4: DJNZ R6,X4 ; lap vong tu nhan X4

ACALL LEFT

MOV 40H,#10

MOV 43H,#10

MOV 46H,#30

X4: DJNZ R7,L1

LJMP Start

L1: LJMP X1

DELAY: MOV R4,#5 ; gan gia tri 5 cho thanh ghi R4

H2: MOV R5,#0FFH ; gan gia tri 255 cho thanh ghi R5

H1: DJNZ R5,H1 ; lap vong cho den khi R5 = 0

DJNZ R4,H2 ; lap vong cho den khi R4 = 0

RET

FRONT: MOV P1,#54H ; gan gtri 54H vao cong p1

MOV P0,#02H ; gan gtr 02H vao cong p0

RET

RIGHT: MOV P1,#00CH ; gan gia tri 0A1H vao P1

MOV P0,#03H ; gan gia tri 02H vao cong P0

RET

BACK: MOV P1,#0A2H

MOV P0,#08H

RET

LEFT: MOV P1,#61H

MOV P0,#04H

RET

Loop:

jmp Loop

;====================================================================

END