

**ĐỀ THI CUỐI KỲ**  
**MÔN: VI ĐIỀU KHIỂN**  
**LỚP: CĐ ĐKĐT 19A,B**

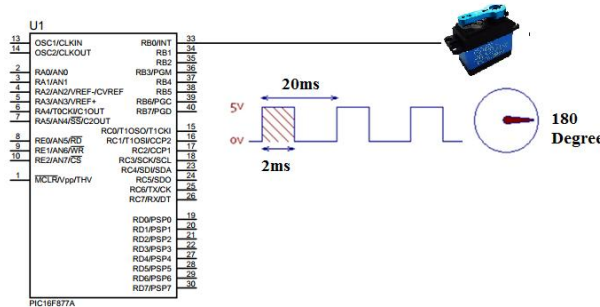
Ngày thi: --/--/2021

Thời gian: 90 phút (Không kể thời gian chép/phát đề thi)

-----*(Sinh viên sử dụng tài liệu 1 tờ giấy chép tay A4, không trao đổi tài liệu)*-----

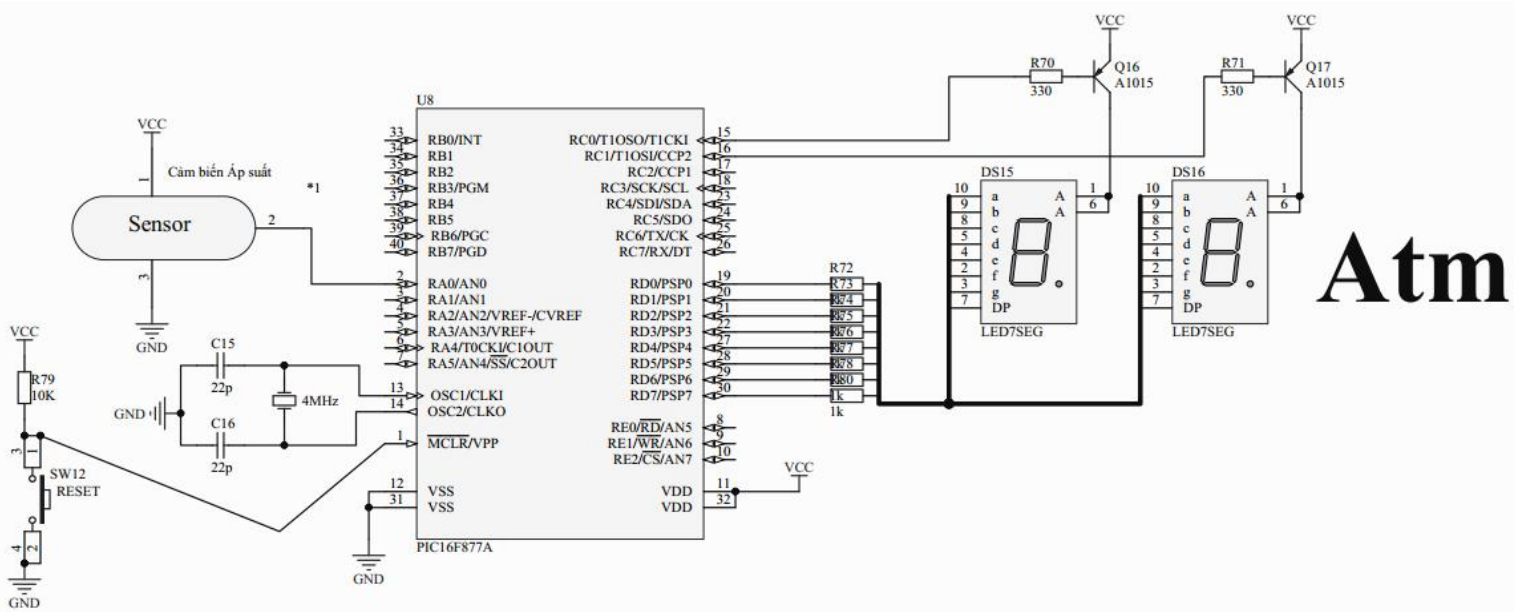
**Câu 1: (2.5 điểm)** Sử dụng vi điều khiển PIC16F877A cho ứng dụng phát xung điều khiển động cơ quay góc RC Servo như hình bên dưới. Viết chương trình điều chế xung vuông cố định ra chân RB.0 điều khiển động cơ RC Servo quay góc 180 độ sử dụng bộ Timer0.

**Yêu cầu:** sử dụng bộ Timer0 chức năng định thời, ngắt tràn Timer0, bộ chia Timer0: 1/2. Xung có chu kỳ tác động Ton = 2 ms, chu kỳ tuần hoàn T = 20 ms như hình.



- a) Tính toán giá trị cài Timer0, số lần ngắt tràn Timer0 để định thời gian chu kỳ tác động Ton và chu kỳ tuần hoàn xung T, vi điều khiển PIC16F877A sử dụng bộ dao động thạch anh 20MHz. (1đ)
- b) Viết chương trình C theo cấu trúc sử dụng ngắt Timer0 cho yêu cầu trên. (1.5đ)

**Câu 2: (3 điểm)** Cho sơ đồ nguyên lý như hình sau. Giả sử cảm biến đo Áp suất theo nhà sản xuất có tín hiệu ngõ ra Analog, tuyến tính với dải áp suất từ 0 (Atm) - 50 (Atm) theo công thức sau:  $U(V) = 0.05 * P(Atm)$ . Chân tín hiệu ngõ ra của cảm biến mắc vào chân RA.0, tần số thạch anh sử dụng là 4MHz.



- a) Chọn giá trị điện áp tham chiếu Vref bằng bao nhiêu để đảm bảo đo được hết tầm cảm biến và độ chính xác cao nhất? Thiết kế mạch tạo điện áp tham chiếu Vref như đã chọn và xác định tên chân cài điện áp tham chiếu?. (1đ)

- b) Viết biểu thức tính Áp suất từ giá trị đo được của bộ chuyển đổi ADC có độ phân giải 8 bit. (0.5đ)  
 c) Viết chương trình C đọc giá trị ngõ ra Analog của cảm biến và hiển thị số nguyên là **Áp suất** theo đơn vị Atm lên 2 Led 7 đoạn Anode chung theo sơ đồ nguyên lý trên. (1.5đ)

**Câu 3: (2 điểm)** Thiết kế sơ đồ nguyên lý và viết chương trình điều chế xuất xung PWM có tần số 10KHz ra chân RC.1 theo yêu cầu sau:

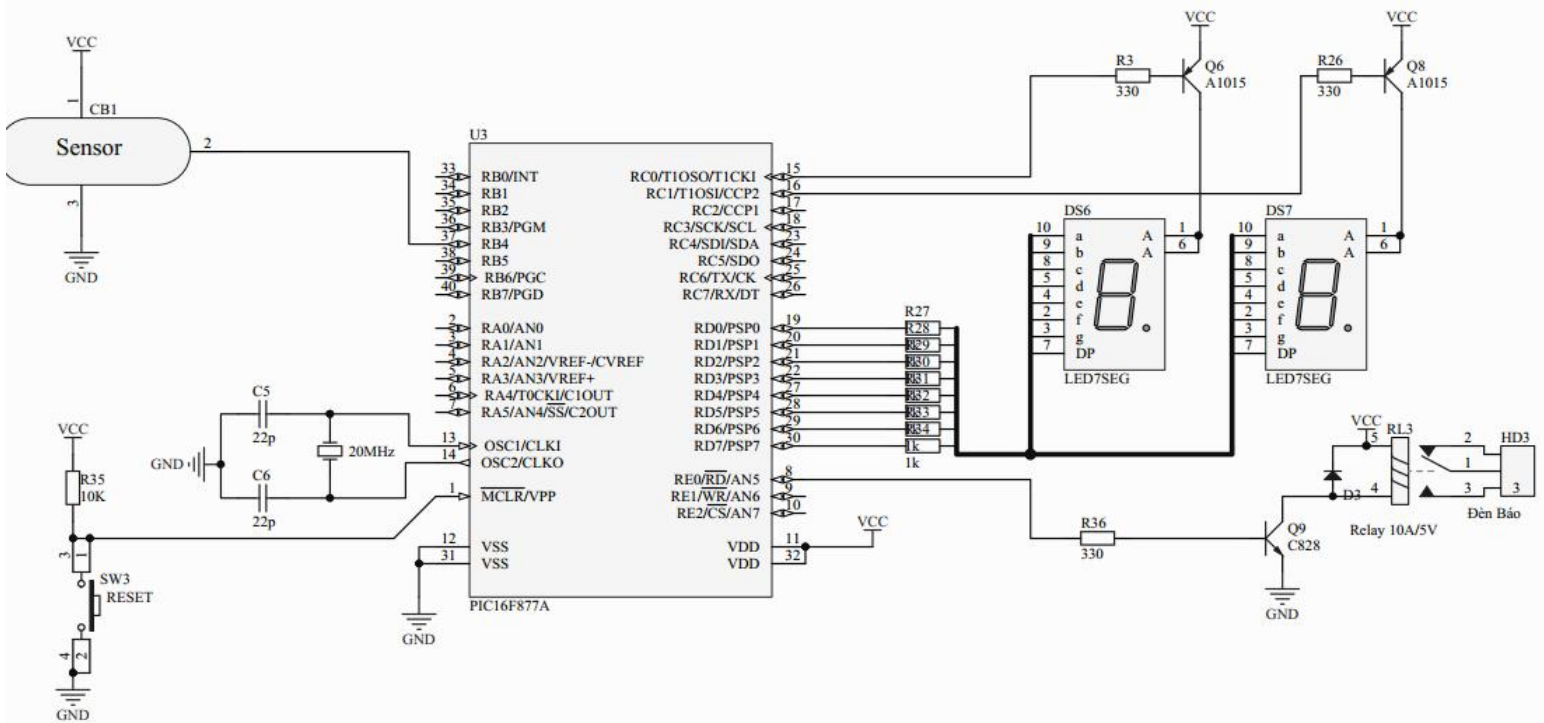
- + Bàn đầu cấp nguồn hoặc trường hợp nhấn rồi thả nút B3 thì tắt xuất xung PWM.
  - + Nhấn rồi thả nút B1 xuất xung PWM có % Duty Cycle là: 20%.
  - + Nhấn rồi thả nút B2 xuất xung PWM có % Duty Cycle là: 60%.
- Biết tần số thạch anh yêu cầu sử dụng là 20 MHz.

a) Vẽ sơ đồ nguyên lý gồm 3 nút nhấn điều khiển xuất xung B1, B2 và B3. Một chân ngõ ra xuất xung PWM điều khiển độ sáng của Led đơn. (0.5đ)

b) Tính toán các thông số cài đặt xuất xung PWM và viết chương trình C theo yêu cầu nêu trên. (1.5đ)

**Câu 4: (2.5 điểm)** Cho sơ đồ nguyên lý mạch hiển thị đếm số lượng xe vào trong bãi giữ xe như hình bên dưới. Một cảm biến quang tác động mức cao có ngõ ra ON/OFF 5VDC, "CB1" được nối vào chân RB.4 phát hiện có người đi vào cổng Gate1. Chân ngõ ra RE.0 dùng để kích đèn báo bãi giữ xe đã đầy. Viết chương trình C hoàn thiện, sử dụng **ngắt PORTB**, báo số lượng xe vào trong bãi theo yêu cầu sau:

- a) Bàn đầu hiển thị 00. Khi có xe vào cổng Gate1 cảm biến CB1 tác động, màn hình 2 led 7 đoạn báo số lượng xe tăng lên 1 đơn vị. Nếu số hiển thị đã tới 99, CB1 tác động sẽ không tăng thêm. Giá trị hiển thị đặt lại 00 khi nhấn Reset. Sinh viên có thể sử dụng lại chương trình tách số quét led hiển thị hai số nguyên như ở **Câu 2**. (1đ)
- b) Khi số lượng xe  $\geq 99$  thì đèn báo sáng báo bãi xe đã đầy không nhận xe nữa. Khi số lượng xe  $< 99$  thì đèn báo tắt báo bãi xe còn có thể nhận xe và hiển thị số lượng xe trong bãi. (0.5đ)
- c) Vẽ lưu đồ giải thuật của chương trình yêu cầu trên. (1đ)



TP. HCM, Ngày Tháng Năm 20  
 GV ra đề:

**ĐÁP ÁN ĐỀ THI CUỐI KỲ**  
**MÔN THI: VI XỬ LÝ**  
**LỚP: CĐ TĐ 19 A,B**  
**Thời gian: 90 phút**

	<p>a) Tính toán giá trị cài Timer0, số lần ngắt khi tràn Timer0 để định thời gian chu kỳ tác động Ton và chu kỳ tuần hoàn xung T, vi điều khiển PIC16F877A sử dụng bộ dao động thạch anh 20MHz. (1đ)</p> <p>Yêu cầu: thạch anh 20MHz, sử dụng Timer0 định thời, bộ chia 1/2</p> <p><b>+ Tính được tổng số xung để định thời được 2ms: 0,25đ</b></p> <p>Thời gian đếm lên 1 xung trong Timer0. Với hệ số chia = 2</p> $T_t = \frac{4}{f_{osc}} * pres = \frac{4}{20 * 10^6} * 2 = 0.4 \mu s$ <p>Thời gian đề bài cần định thời: <math>T_{ON} = 2ms</math></p> <p>⇒ Tổng số xung Timer1 cần đếm:</p> $\frac{T_{ON}}{T_t} = \frac{2 * 10^3 \mu s}{0.4 \mu s} = 5000 \text{ xung}$ <p><b>+ Tính được số lần ngắt tràn Timer0 để định thời được 2ms và 20ms: 0,5đ</b></p> <p>Ta có: 5000 xung &gt; 256 xung (số xung 1 chu kỳ timer 8 bit chạy từ giá trị 0 đến tràn) =&gt; tách xung  <math>5000 = 25 * 200</math> xung (<math>200 &lt; 256</math>)</p> <p>Do đó Timer0 định thời được 2ms có số lần ngắt tràn timer: 25 lần. (0,25đ)</p> <p>⇒ Chu kỳ tuần hoàn xung T = 20ms Timer0 có số lần ngắt tràn là: 250 lần (<math>20 \text{ ms} = 10 * 2\text{ms}</math>) (0,25đ)</p> <p><b>+ Tính được giá trị cài cho Timer0: 0,25đ</b></p> <p>Mỗi lần chạy 200 xung. Giá trị xung khởi tạo Timer0 bắt đầu chạy:  <math>256 - 200 = 56</math> xung</p>	<b>1đ</b>
<b>Câu 1</b>	<p>b) Viết chương trình C theo cấu trúc sử dụng ngắt Timer0 cho yêu cầu trên. (1.5đ)</p> <p><b>+ Cấu trúc chương trình ngắt timer có đầy đủ các thành phần : 0,25đ</b></p> <pre>#include &lt;16f877a.h&gt; #fuses NOWDT, HS #use delay(clock=20000000) + Khai báo kiểu dữ liệu tối ưu: 0,25đ int16 count=0; //Chương trình ngắt TMR1 #INT_TIMER0 void interrupt_timer1() {     count++; + Xử lý đúng khi xác định số lần ngắt tràn là 25 lần: 0,25đ     if(count&gt;=25) output_low(PIN_B0); + Xử lý đúng khi xác định số lần ngắt tràn là 250 lần: 0,25đ     if(count&gt;=250)     {         output_high(PIN_B0);         count =0; //reset biến đếm tràn     }     set_timer0(56); } void main() {</pre>	

	<p>+ <b>Cài đặt I/O theo đúng yêu cầu đề bài và trạng thái ban đầu RB.0: 0,25đ</b>  <code>set_tris_b(0x00);</code>  <code>output_b(0x00);</code>  <code>output_high(PIN_B0);</code></p> <p>+ <b>Cài đặt ngắt Timer0 theo đúng thông số đã tính toán: 0,25đ</b>  <code>enable_interrupts(INT_TIMER0);</code>  <code>setup_timer_0(T0_INTERNAL\T0_DIV_2);</code>  <code>enable_interrupts(GLOBAL);</code>  <code>set_timer0(56);</code>  <code>while (true)</code>  <code>{</code>  <code>}</code>  <code>}</code></p>	<b>1,5đ</b>
	<p>a) Chọn giá trị điện áp tham chiếu Vref bằng bao nhiêu để đảm bảo đo được hết tầm cảm biến và độ chính xác cao nhất ?. Thiết kế mạch tạo điện áp tham chiếu Vref như đã chọn và xác định tên chân cài điện áp tham chiếu ?. (1đ)</p> <p>+ <b>Xác định đúng điện áp tham chiếu tốt nhất: 0,5đ</b>  Để đảm bảo đo được hết tầm, điện áp tham chiếu:  <math display="block">V_{ref} = V_{out\ max} = 0.05 * 50 = 2.5\ (V)</math></p> <p>+ <b>Thiết kế được mạch tạo điện áp tham chiếu: 0,5đ</b></p> <ul style="list-style-type: none"> <li>Sử dụng mạch phân áp:  <math display="block">V_{ref} = \frac{VCC * R2}{R1 + R2} = \frac{5 * 10K}{R1 + 10K} = 2.5\ V</math> </li> </ul> <p>Chọn R2 = 10K. Ta có:  <math display="block">V_{ref} = \frac{VCC * R2}{R1 + R2} = \frac{5 * 10K}{R1 + 10K} = 2.5\ V</math></p> <p>Suy ra: R1 = R2 = 10K</p> <ul style="list-style-type: none"> <li>Sử dụng biến trở: Chỉnh điện áp ngõ ra 2.5 V  Điện áp tham chiếu cài vào chân RA.3</li> </ul>	<b>1đ</b>
	<p>b) Viết biểu thức tính Áp suất từ giá trị đo được của bộ chuyển đổi ADC có độ phân giải 8 bit. (0.5đ)</p> $P = V * 20\ (Atm)$ $P = \frac{giatriADC * 2.5 * 20}{255}$	<b>0,5đ</b>
	<p>a) Viết chương trình C đọc giá trị ngõ ra Analog của cảm biến và hiển thị số nguyên là <b>Áp suất</b> theo đơn vị Atm lên 2 Led 7 đoạn theo sơ đồ nguyên lý trên. (1.5đ)</p> <p>+ <b>Cấu trúc chương trình có đầy đủ các thành phần: 0,25đ</b>  <code>#include &lt;16f877A.h&gt;</code>  <code>#device ADC = 8</code>  <code>#fuses NOWDT, HS</code>  <code>#use delay(clock=4000000)</code></p> <p>+ <b>Khai báo chọn đúng mã led theo yêu cầu đề bài: 0,25đ</b>  <code>// ma 7 doan khong dau</code>  <code>int maled[10]={0xc0,0xf9,0xa4,0xb0,0x99,0x92,0x82,0xf8,0x80,0x90};</code>  <code>int16 giatriADC=0;</code>  <code>float Volt;</code>  <code>float P;</code>  <code>int8 gtht;</code></p>	

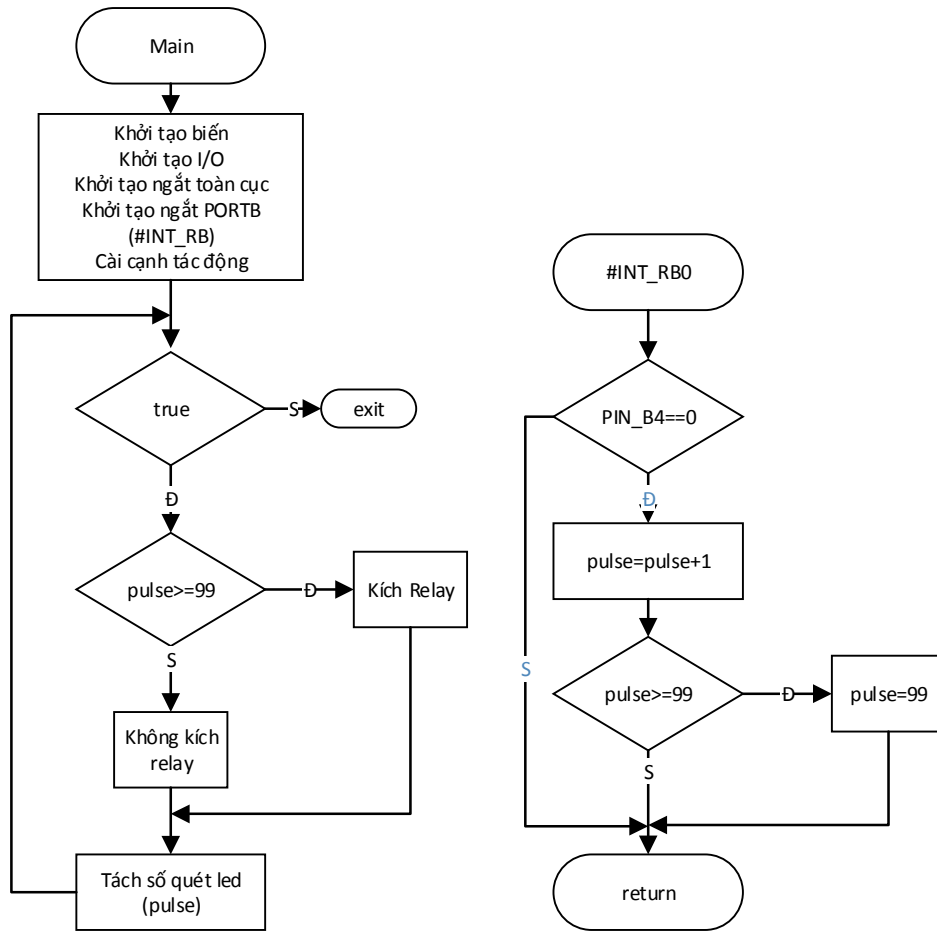
<p><b>Câu 2</b></p>	<p><b>+ Tách số quét led theo đúng yêu cầu đề bài: 0,5đ</b>  void tachso_quetled(int8 value)  {  int8 chuc=0,donvi=0;  chuc = value/10;  donvi = value%10;  output_low(PIN_C0);  output_d(maled[chuc]);  delay_ms(10);  output_high(PIN_C0);  output_low(PIN_C1);  output_d(maled[donvi]);  delay_ms(10);  output_high(PIN_C1);  }  void main()  {  set_tris_c(0x00);  set_tris_d(0x00);  output_c(0xff);  output_d(0x00);  <b>+ Cài đúng bộ chuyển đổi ADC theo yêu cầu đề bài: 0,25đ</b>  setup_adc_ports(AN0_AN1_AN2_AN4_VSS_VREF); // chan A3 cai dien ap tham chieu  set_adc_channel(0);  setup_adc(ADC_CLOCK_DIV_2);  delay_us(10);  while (true)  {  <b>+ Xử lý đúng công thức tính nhiệt độ: 0,25đ</b>  giatriADC = read_adc();  Volt = (float)(giatriADC*2.5)/255.0;  P = Volt*20.0;  gtht = (int8)(P);  tachso_quetled(gtht);  }  }</p>	<p><b>1,5đ</b></p>
	<p>a) Vẽ sơ đồ nguyên lý gồm 3 nút nhấn điều khiển xuất xung B1, B2 và B3. Một chân ngõ ra xuất xung PWM điều khiển độ sáng của Led đơn. (0.5đ)  + Vẽ đầy đủ 3 thành phần cơ bản để vi điều khiển hoạt động: Nguồn, nút reset, bộ dao động thạch anh: 0,25đ  + Vẽ đầy đủ và đúng nguyên lý các ngoại vi như đề bài yêu cầu: 3 nút nhấn điều khiển xuất xung B1, B2 và B3, có đầu dây vào các chân I/O đầy đủ, ngõ ra xuất xung PWM điều khiển độ sáng của Led đơn: 0,25đ  Gợi ý đáp án:</p>	<p><b>0,5đ</b></p>



<p><b>Câu 4</b></p>	<p>Viết chương trình C hoàn thiện sử dụng ngắt PORTB, báo số lượng xe đã vào bãi. (1,5đ)</p> <pre> #include &lt;16f877a.h&gt; #fuses HS,NOWDT #use delay (clock = 20000000) int8 maled[10]={0xc0,0xf9,0xa4,0xb0,0x99,0x92,0x82,0xf8,0x80,0x90}; int8 pulse=0; void tachso_quetled(int8 value); #INT_RB void ngat_port() { + <b>Tăng giá trị hiển thị và giới hạn tăng giá trị: 0,25đ</b>   If(input(PIN_B4)==0)   {     pulse=pulse+1;     if(pulse &gt;=99) pulse=99;   } } void main() { + <b>Cài đặt IO theo sơ đồ cho trước và trạng thái ban đầu chân E.0: 0,25đ</b>   set_tris_b(0xff);   set_tris_c(0x00);   set_tris_d(0x00);   set_tris_e(0x00);   output_c(0xff);   output_d(0x00);   output_low(PIN_E0); + <b>Khai báo ngắt PORTB đúng theo đề bài: 0,25đ</b>   enable_interrupts(GLOBAL);   enable_interrupts(INT_RB);   ext_int_edge(H_TO_L);   while(TRUE)   { + <b>Kiểm tra số lượng xe &gt;=98 để báo động: 0,5đ</b>     If(pulse&gt;=99) output_high(PIN_E0);     Else output_low(PIN_E0); + <b>Đưa giá trị đếm hiển thị quét led theo sơ đồ: 0,25đ</b>     tachso_quetled(pulse);   } } </pre>	<p><b>1,5đ</b></p>
---------------------	--	--------------------

Vẽ lưu đồ giải thuật của chương trình yêu cầu trên. (1đ)

Mỗi phần 0,5đ:



1đ

TBM TỰ ĐỘNG HÓA

GV ra đề:

TS. Đặng Đức Chi

ThS. Cù Minh Phước