**MÔ TẢ GIẢI THUẬT CHƯƠNG TRÌNH ĐO TẦN SỐ (Hz) VÀ CHU KÌ NHIỆM VỤ (%)**

- **Họ và tên:** Lê Đức Quý

- **MSSV:** 2114597

**Thứ tự các mục tương ứng với cấu trúc chương trình chạy (trước, sau)**

1. **Khai báo các PORT:**

* Port B: xuất LCD
* PortC: điều khiển LCD
* PortD: **PD6** ngõ vào

1. **In ký tự:**

* In các ký tự **“f(Hz):”** ở dòng 1 và **“CKNV(%):”** ở dòng 2 lên LCD

1. **Đo tần số**

* Cấu hình Timer1 mode NOR, nạp giá trị **-31250** cho TCNT1 để Timer1 đếm 1 giây
* Dùng 2 thanh ghi **R24, R25** làm biến điếm số xung (**1 xung hoàn chỉnh bao gồm 1 mức cao và 1 mức thấp**) có được trong **1 giây**.
* Timer1 chạy
* Liên tục kiểm tra **cạnh lên** của PD6,
* Nếu xuất hiện cạnh lên thì liên tục kiểm tra **cạnh xuống** của PD6
* Nếu xuất hiện cạnh xuống thì đã hoàn thành **1 xung**, tăng cặp biến đếm lên **1** bằng câu lệnh

ADIW R24,1

* Kiểm tra **Timer1 tràn**, nếu chưa tràn thì tiếp tục quay vòng đếm số xung
* Nếu Timer1 tràn thì xóa cờ, dừng Timer, đọc giá trị **R24, R25** (chứa số xung đếm được trong 1 giây)

- Chuyển nội dung thanh ghi **R24,R25** lần lượt vào **R20,R21**

- Gọi chương trình chuyển số nhị phân 16 bit thành 5 ký số BCD và in lên màn hình tần số (tổng số xung đếm được trong 1 giây).

1. **Đo chu kì tổng**

* Cấu hình Timer1 mode NOR, N=8, bắt ngõ vào cạnh lên
* Chờ bắt được xung kích khởi, cất thời điểm lấy mẫu cạnh lên
* Chờ bắt được xung kích khởi, cất thời điểm lấy mẫu cạnh lên tiếp theo
* Trừ thời điểm sau với thời điểm đầu

🡺 Chu kì tổng của xung sẽ được lưu trong thanh ghi **R20,R21**

**\*Ý tưởng giải quyết vấn đề chu kì nhiệm vụ**

* Vấn đề: Ta xét chu kì nhiệm vụ max = 90 %

Nếu xung có thời gian mức cao **>655us** (tương ứng chu kì tổng **>727us**) thì khi ta nhân thời gian mức cao với 100, giá trị tử số sẽ **>65535 (>16 bit)**. Dẫn đến vấn đề tràn bit.

* Do đó, ta có **cách tính đối với các xung có thời gian mức cao lớn hơn 655** như sau:
* Cách tính này, ta chia chu kì tổng cho 100 trước, sau đó mới lấy thời gian mức cao chia cho chu kì tổng (đã bị chia 100). Em sử dụng cách tính để khắc phục vấn đề tràn bit đối với các xung có chu kì nhiệm vụ lớn hơn 655us. Tuy nhiên sẽ tăng thêm một chút sai số.
* Đối với **xung có thời gian mức cao nhỏ hơn 655, ta dùng cách tính sau:**
* Cách tính này, ta nhân thời gian mức cao cho 100 trước (do nên ta không lo bị tràn bit), sau đó lấy thời gian mức cao (đã nhân với 100) chia cho chu kì tổng. Cách tính này ít sai số hơn cách tính 1, nhưng không giải quyết được vấn đề tràn bit.
* **Em sẽ sử dụng cả 2 cách tính cho từng trường hợp để khai thác tối đa phần cứng và giảm thiểu độ sai số ít nhất có thể. Để làm được điều đó, ta cần thêm bước kiểm tra giá trị chu kì tổng lớn hơn hay nhỏ hơn 727us. (ở đây em lấy 725 cho chẳn).**

1. **Kiểm tra**

* Kiểm tra xem chu kì tổng có **>= 725us** không
  + Nạp giá trị **725** cho cặp thanh ghi R16(LOW), R17(HIGH)
  + Trừ R20,R21 với R16,R17
  + Nếu có cờ Carry thì giá trị chu kì tổng **<725us**, nhảy tới nhãn **PERIOD2**
  + Nếu không có cờ Carry thì giá trị chu kì tổng **>=725us**, nhảy tới nhãn **PERIOD1**
* Nhãn **PERIOD2**: nạp giá trị chu kì tổng này vào R4,R5 (số chia của ctc DIV\_16), đồng thời đánh dấu đây là trường hợp 2 bằng **R22=2**
* Nhãn **PERIOD1**: chia giá trị chu kì tổng này cho **100** sau đó mới nạp vào R4,R5 (số chia của ctc DIV\_16), đồng thời đánh dấu đây là trường hợp 1 bằng **R22=1**

1. **Đo chu kì nhiệm vụ**

* Cấu hình Timer1 mode NOR, N=8, bắt ngõ vào cạnh lên
* Chờ bắt được xung kích khởi, cất thời điểm lấy mẫu cạnh lên
* Cấu hình Timer1 mode NOR, N=8, bắt ngõ vào cạnh xuống
* Chờ bắt được xung kích khởi, cất thời điểm lấy mẫu cạnh xuống
* Trừ thời điểm sau với thời điểm đầu

🡺 Thời gian mức cao sẽ được lưu trong thanh ghi R20,R21

1. **Tính toán**

* Kiểm tra trường hợp
  + : Trường hợp 1, nhảy tới nhãn **DUTY1**
  + : Trường hợp 2, nhảy tới nhãn **DUTY2**
* **DUTY1:** nạp R20,R21 (thời gian mức cao) vào số bị chia của ctc DIV\_16
  + Gọi ctc DIV\_16 thực hiện phép chia R20,R21 (thời gian mức cao) với R4,R5 (chu kì tổng đã bị chia **100**)
  + In ra LCD chu kì nhiệm vụ (%)
* **DUTY2**: nhân R20,R21 (thời gian mức cao) với **100**, sau đó kết quả nạp vào số bị chia của ctc DIV\_16

**\*Giải thuật nhân số 16 bit (R20 (byte thấp), R21 (bytecao) ) với số 8 bit (R16)**

* 1. MUL R20,R16; nhân byte thấp với R16 trước
  2. MOVW R18,R0; chuyển tích vào R18, R19
  3. MUL R21,R16; nhân byte cao với R16
  4. ADD R19,R0; cộng byte cao của tích vào R19
  5. MOVW R20,R18; chuyển tích từ R18,R19 vào R20,R21
  + Gọi ctc DIV\_16 thực hiện phép chia R20,R21 (thời gian mức cao đã được nhân **100**) với R4,R5 (chu kì tổng)
  + In ra LCD chu kì nhiệm vụ (%)