Example solutions for the Problem 2

a) 

delay

start

Vcc

start

MCLR

RA3 → LED_on

RA1 → Pulse

RC4

RC3

RD4

RD3

RD2

RB7

RB6

RB5

RB0/INT

MCLR = 0

b) 

MCLR-L signal

init_system

read inputs

state logic

output logic

write outputs

start detection

TMRO1F detected

JSR C

Var_Delay (8-bit to program the delay)

Var_Pulse → '1' to generate the 20µs pulse

Var_LED_on → '1' when running

Flag_Start → '1' when INT0IF = 1

Flag_Timer → '1' when TMRO1F = 1

current_state (8 bits to encode the states)

A, B, C, ...
When detecting an active edge (programmable) from INT0 (R80) the hardware flag is set (INT0IF = 1) rising/falling.

When detecting TMRO overflow → TMROIF = 1
⇒ Both enabling masks must be set to allow interrupts
d) The objective is to generate the variable

\[ \text{Var\_Delay} = (\text{PortB} \& \text{Ob}11100000) \gg 5 \] 

1. Read PortD
2. Mask \text{Ob}00001100
3. Shift 1 bit and save in buffer Var\_Buffer1
4. Read PortC
5. Mask \text{Ob}0000110000
6. Shift 3 bit and save in Var\_Buffer2
7. Read PortB
8. Mask \text{Ob}11100000
9. Shift 5 bit and or and save in Var\_Delay

\[ \text{Var\_Delay} = (\text{PortB} \& \text{Ob}11100000) \gg 5 \] 

To write Var\_Pulse and Var\_LED\_on in the same Port while preserving other Port bits

1. Read the PortA
2. Mask bits of no interest (to preserve them)
3. Or the Pulse and LED\_on bits after shifting them
4. Write the PortA

\[ \text{Var\_Buffer1} = \text{PortA} \& \text{Ob}11100011; \]

\[ \text{PortA} = \text{Var\_Buffer1} | (\text{Var\_Pulse} << 4) | (\text{Var\_LED\_on} << 3); \]
f) Let's use the timer0 to generate the delay. For example:

\[
\text{Delay time} = \frac{4}{16\text{MHz}} \cdot N_1 \cdot N_2 \cdot N_3 \quad (\text{ms})
\]

\[
1 \rightarrow 1 \text{ms}
\]

\[
255 \rightarrow 255 \text{ms}
\]

\[
(256 - \text{Var. Delay})
\]

\[
\frac{1\text{ms} \cdot N_2 \cdot 1000}{\mu\text{s}}
\]

\[
\mu\text{s}
\]

\[
\text{ms}
\]

- Let's use the same timer0 to generate 20 μs. For example:

\[
N_3 \text{ is not required}
\]

\[
N_1 < N_2 \quad (256 - 20)
\]

\[
20 \mu\text{s} = 250\text{ns} \cdot 4 \cdot 20 \quad \text{TMRO}
\]

For both timings:

\[
\begin{align*}
\text{TOSC} &= \varnothing \\
\text{TOPS} &= \text{SELECT } \div 4 \quad (N_1 \text{ prescaler})
\end{align*}
\]

9) 6 states are required to generate the delay and the pulse
h) The ISR must be in charge of setting the Start_Flag and the Timer_Flag variables.

- **INT0IF = 1?**
  - **Yes** → Start_Flag = 1 → End
  - **No** → TMRO1F = 1
    - **Yes** → current_state = Delay?
      - **Yes** → TMRO1F = 0; Reload the TMRO with N2; post.scalar_value += 1
      - **No** → current_state = Delay?
    - **No** → **end**

So that it can continue counting time.

If there is a third interrupt and current state is not Delay it must be when generating the pulse.

i) Output logic generates the variable Var_Pulse, Var_LED_on and the values to configure the Timer0

<table>
<thead>
<tr>
<th>Delay</th>
<th>Current_state</th>
<th>Var_LED_on</th>
<th>Var_Pulse</th>
<th>Timer0 configuration bits and variable</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Load Timer delay</strong></td>
<td>Idle</td>
<td>0</td>
<td>0</td>
<td>Timer off</td>
</tr>
<tr>
<td><strong>Delay</strong></td>
<td>Load Timer delay</td>
<td>1</td>
<td>0</td>
<td>N1 = 9; N2 = Delay</td>
</tr>
<tr>
<td><strong>Load pulse</strong></td>
<td>Delay</td>
<td>1</td>
<td>1</td>
<td>N1 = 9; N2 = 20</td>
</tr>
<tr>
<td><strong>Pulse</strong></td>
<td>Load pulse</td>
<td>1</td>
<td>1</td>
<td>Timer off</td>
</tr>
<tr>
<td><strong>Stop timer</strong></td>
<td>Stop timer</td>
<td>0</td>
<td>0</td>
<td>Timer off</td>
</tr>
</tbody>
</table>

**CASE**

- **Idle** → load timer delay
- **Delay** → load pulse
- **Stop timer** → stop timer
This function generates all the state transitions (arrows) and it is also interpreted in a behavioural way to generate the C code. The important statement is also the switch-case.

9 operations to set the new value of the current-state variable.

As usual, with all this a, b, ... j planning, now is time to start developing the project in the lab.

→ Take an example from P10 - P11 - P12 and copy & adapt it step by step.

NEW PROJECT \( \rightarrow \text{Compile} \rightarrow \text{run} \rightarrow \text{test} \rightarrow \text{out} \rightarrow \text{Report} \)