**16F877A PICMicro programming under MPLAB – Project Checklist**

**Instructions:**

**Table 1 – Team Members and Contributions:**

You are required to clearly define how you managed the project as a team. Please specify the individual contributions of each member in the "Contributions" field.

**Table 2 – Checklist:**

You are also required to fill out Table 2 honestly. All fields marked "Yes" will be verified during the project discussion.

Table 1: Team Members and Contributions

|  |  |  |
| --- | --- | --- |
| **Team Member Name** | **Team Member ID** | **Contributions** |
| **SSara Ewaida** | **1203048** | Designed the complete schematic on Proteus for the dual PIC16F877A system and implemented all requirements from 1 to 6 in the project. |
|  |  |  |
|  |  |  |
|  |  |  |

Table 2: Checklist

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **#** | **Action Required** | **Did you implement… ?** | | **Does it work?** | |
|  |  | **Yes** | **No** | **Yes** | **No** |
| 1 | Display “Welcome to” and “Division” blinking 3 times with 0.5s delay on power-up |  |  |  |  |
| 2 | Wait 2 seconds after the welcome message before input starts |  |  |  |  |
| 3 | Show “Number 1” on LCD and allow digit-by-digit input using button P |  |  |  |  |
| 4 | Increment digit on each click, wrap from 9 → 0 |  |  |  |  |
| 5 | Lock digit after 1 second of inactivity |  |  |  |  |
| 6 | Auto-fill remaining digits with same value as first digit |  |  |  |  |
| 7 | Allow user to manually adjust each digit |  |  |  |  |
| 8 | Support double-click to skip integer input and go to decimal part |  |  |  |  |
| 9 | Allow similar entry for decimal part (auto-fill and manual adjustment supported) |  |  |  |  |
| 10 | Support double-click to confirm decimal part and move to Number 2 |  |  |  |  |
| 11 | Show “Number 2” on LCD for 1 second |  |  |  |  |
| 12 | Allow same digit-by-digit entry process for Number 2 |  |  |  |  |
| 13 | After second number is entered, show “=” on screen |  |  |  |  |
| 14 | Master sends Number 1 (int & decimal) to co-processor via PortC with interrupts and acknowledgments |  |  |  |  |
| 15 | Master sends Number 2 similarly |  |  |  |  |
| 16 | Co-processor performs division correctly |  |  |  |  |
| 17 | Co-processor sends result back byte-by-byte using interrupts + acknowledgments |  |  |  |  |
| 18 | Master displays result on LCD second line, “Result” on first line |  |  |  |  |
| 19 | While result is displayed, button P cycles through Number 1 → Number 2 → Result |  |  |  |  |
| 20 | Double-click on P restarts the process from Number 1 input |  |  |  |  |
| 21 | Proteus schematic includes: 2 × 16F877A MCUs, push button, LCD |  |  |  |  |
| 22 | LCD connected via PortD in 4-bit mode with 4.7kΩ RS pull-up resistor |  |  |  |  |
| 23 | Push button connected to PortB.0 with 10kΩ pull-up (external or internal) |  |  |  |  |
| 24 | Data transferred over PortC between master and co-processor |  |  |  |  |
| 25 | 4 MHz oscillator and 2×15pF capacitors connected correctly |  |  |  |  |
| 26 | 10kΩ pull-up resistor connected to MCLR pin |  |  |  |  |