**MICROPROCESSOR SYSTEMS**

**ASSIGNMENTS # 1, 2, 3**

**DATE: 15 MARCH 2023**

**CLASS: BEE 13 D**

|  |  |
| --- | --- |
| **NAME** | **REG NO** |
| MARYAM IJAZ | 367703 |
| HAFSA MISBAH | 373776 |
| ALISHBA ARIF | 372182 |
| RAMEEN SHAHZAD | 373757 |

**INTRODUCTION:**

Most of the first-ever microprocessors created during the early 1970s were 4-bit microcontrollers. Some of these microprocessors such as TMS1000, NEC μPD751, and the intel 4004 had 4-bit word lengths. The 4-bit architecture was limiting and a shift towards higher bits was made.

The 4-bit microcontrollers found their way into many battery-powered and low-power instruments and devices. Most of the early 4-bit microprocessors were used almost exclusively in calculators and toys, and later for various utilities, video games, controllers, and early computers. 4 bits was a logical choice for many calculators that used BCD numbers representation.

**EXPLANATION:**

The 4-bit architecture is a microprocessor or computer architecture that has a data path width or a highest operand width of 4 bits or a nibble. These architectures typically have a matching register file with a register width of 4 bits and 4-8-bit wide addresses.

Here we have created a 4-bit microcontroller that performs a few simple functions.

**PARTS OF THE MICROCONTROLLER:**

In the microcontroller, we have

1. The switches
2. The instruction register
3. The instruction decoder
4. 4 memory registers (A, B, C, D)
5. ALU that consists of its own registers and storage.
6. 7-segment display

**COMPONENTS USED:**

The components used in the microcontroller are:

1. D Flip flops
2. Full adders
3. Not gates
4. And gates
5. Xor gates
6. Or gates
7. Switches
8. Clock
9. 3 to 8 decoder
10. 7 segment display

**WORKING:**

In this microcontroller, we shall begin with the instruction register which is composed of a clock and the switches that are connected to the D flip flops through a combination of NOT, AND, and OR gates. The switches will decide where the result will be stored. The result could be stored in either of the 4 storage registers we have made. The selection is based on the value of the first switch when the clock is high.

One among the A, B, C, and D memory registers is chosen above. These temporary memory registers like the instruction register are composed of D flip flops and a combination of NOT, AND, and OR gates. These registers store the numbers on which the function is to be implemented.

The other 3 switches send instructions to the instruction decoder. The combination of signals from these switches will select which function will be chosen to be implemented. In this microprocessor 6 functions are being implemented. Each function has its specific 3-bit code arrangement that selects the function,

|  |  |
| --- | --- |
| **Signal** | **Function** |
| 000 | Add |
| 001 | Sub |
| 010 | Inc |
| 011 | Dec |
| 100 | OR |
| 101 | ADD |

The instruction decoder will then create an 8-bit code that will trigger the circuit of the function. We have 4 different circuits, one for the full adder and subtractor, one for increment and decrement, one for the OR function, and one for AND function.

The 4-bit code attained from these circuits is then input to the ALU storage that is connected to the memory register and that is connected to the ALU memory register.

The stepwise working is explained:

1. The instruction register selects the memory registers in which the numbers are to be saved.
2. The numbers are entered into the memory register through their switches.
3. The switches of the instruction register send code to the instruction decoder i.e. 3:8 decoder that selects the function to be performed.
4. The numbers enter the function circuits from the memory registers and the result leaves the function circuits as a 4-bit code.
5. This code is then sent to the ALU storage.
6. From the ALU storage the code is then sent to a memory register to store the value and to the 7-segment display to view the result.

**SOFTWARE IMPLEMENTATION:**

The software implementation of this microprocessor has been made on simulator.io.

**SIMULATION FILE:**

NOTE: Please ignore the gibberish at the top of the board as it is not erasing. The microprocessor starts below.

<http://simulator.io/board/H9v4V15rlK/16>, [simulator.io | Board](https://simulator.io/board/H9v4V15rlK/16)

**CONCLUSIONS:**

Thus, to conclude, we have made a simple 4-bit microcontroller that demonstrates the flow of data and control signals. The microcontroller demonstrates 6 simple functions i.e. add, subtract, increment, decrement, AND, and OR functions.

The End.