**Microprocessor Basics**

**Technical Terms and Definitions**

1. The truth table is used to illustrate the output for all possible combinations of inputs.
2. The Boolean equation can be used to mathematically illustrate the functional operation of a logic gate.
3. The major subsections of a CPU are: The ALU and control unit.
4. The main thing the microprocessor does is execute instructions.
5. In a computer, an instruction cycle is followed by an execution cycle.
6. The System Clock signal is used to synchronize all microprocessor operations.
7. A Write operation is performed when a data word is placed into memory.
8. Its address determines where a data word will be stored in a memory unit?
9. An AND Gate performs the carry out function of an adder.
10. Instructions are processed sequentially.
11. Three types of buses include Address, Data, and Control (power).
12. Three of the basic parts to a computer are: Input Unit; Output Unit Memory; and CPU.
13. A computer “bus” system is a group of wires that all perform a common function, go everywhere, and connect everything together.
14. Examples of control signals include: 1) clock; 2) enable/disable; 3) read/write; 4) interrupt; and 5) ready to send/clear to send.
15. There are eight (8) bits in a byte?
16. Following is the truth table for two input (A and B) addition:

|  |  |  |  |
| --- | --- | --- | --- |
| A | B | Σ | C |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

1. The phrase “totem pole” in a circuit means: A two transistor circuit where each transistor has an opposite state (when one transistor is on the other is off and vice versa). It is used to produce output voltages that are closer to ideal.
2. The two things a clock signal does are: 1) triggers the start of an instruction cycle; and, 2) defines the duration of the instruction cycle.
3. A decoder circuit uses logic gates to turn on a single output from a particular binary number (code) input.
4. Describe the relationship between the program counter and the stack: The program counter holds the address of the next instruction to be executed. The stack holds the addresses of other programs and data that are not currently being used but have been used at other times and may be called for use again. Addresses from the stack and the program counter go back and forth as needed.