The state diagram of the system is shown in the figure above. Here 5 operations are done. RESET, AND, ADD, NOR and SUB. For each operation we use 000,001,010,011 and 100 opcodes respectively.

Truth Table:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| In decimal | A | | | | | | B | | | | | | | RESET  (op = 000) | ADD  (op = 001) | AND  (op = 010) | | | | NOR  (op = 011) | | | | SUB  (A-B)  (op = 100) |
| - | A3 | A2 | | A1 | A0 | | B3 | B2 | | B1 | | B0 | | - | - | Y3 | Y2 | Y1 | Y0 | Y3 | Y2 | Y1 | Y0 | - |
| 0 (+/-) 0 | 0 | | 0 | 0 | | 0 | 0 | | 0 | | 0 | | 0 | - | 0000  (ZF=1, SF=0, CF=0) | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0000 |
| 0 (+/-) 15 | 0 | | 0 | 0 | | 0 | 1 | | 1 | | 1 | | 1 | - | 1111  (ZF=0, SF=1, CF=0) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 10001  (ZF=0, SF=0, CF=1) |
| 15 (+/-) 0 | 1 | | 1 | 1 | | 1 | 0 | | 0 | | 0 | | 0 | - | 1111  (ZF=0, SF=1, CF=0) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0000 |
| .  .  . | .  .  . | | .  .  . | .  .  . | | .  .  . | .  .  . | | .  .  . | | .  .  . | | .  .  . | -  -  - | .  .  . | .  .  . | .  .  . | .  .  . | .  .  . | .  .  . | .  .  . | .  .  . | .  .  . | .  .  . |
| 15 (+/-) 13 | 1 | | 1 | 1 | | 1 | 1 | | 1 | | 0 | | 1 | - | 11100  (ZF=0, SF=1, CF=1) | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0010  (ZF=0, SF=0, CF=0) |
| 15 (+/-) 14 | 1 | | 1 | 1 | | 1 | 1 | | 1 | | 1 | | 0 | - | 11101  (ZF=0, SF=1, CF=1) | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0001 |
| 15 (+/-) 15 | 1 | | 1 | 1 | | 1 | 1 | | 1 | | 1 | | 1 | - | 11110  (ZF=0, SF=1, CF=1) | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0000 |

The table above shows some of the results and a few of their notable ZFs (Zero Flags), SFs (Sign Flags) and CFs (Carry Flags). When the output is zero or Y = 0000, ZF = 1 otherwise ZF = 0. SF = 1 when output, Y =1xxx or when the MSB bit is 1, else SF =0. CF = 1 when carry bit is 1 or when Y = 1xxxx (overflow), otherwise CF = 0.

For AND gate and NOR gate, it is said to take bitwise inputs of A and B, means if A = 1010 (A3 = 1, A2 = 0, A1 = 1, A0 = 0) and B = 0101 (B3 = 0, B2 = 1, B1 = 0, B0 = 1) AND operation would be Y = 0000 (Y3 = 0 [because A3 =1 & B3 = 0 means 0], Y2 = 0 [because A2 =0 & B2 = 1 means 0], Y1 = 0 [because A1 =1 & B1 = 0 means 0], Y0 = 0 [because A0 =1 & B0 = 0 means 0]). And NOR operation would be Y = 0000 (Y3 = 0 [because A3 =1 ꚛ B3 = 0 means 0], Y2 = 0 [because A2 =0 ꚛ B2 = 1 means 0], Y1 = 0 [because A1 =1 ꚛ B1 = 0 means 0], Y0 = 0 [because A0 =1 ꚛ B0 = 0 means 0]).

**Fig. no**: Simplified state diagram of an ALU with five operations: RESET, AND, ADD, NOR, SUB.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | A3 | A2 | A1 | A0 |
| Input A |  |  |  |  |
|  | B3 | B2 | B1 | B0 |
| Input B |  |  |  |  |
|  | Y3 | Y2 | Y1 | Y0 |
| Output Y |  |  |  |  |

Y4 = Carry