Room FSM

**Next State Table**

Current State Inputs Next State

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Sn | N | E | S | W | V | R | Sn\* |
| S0 | x | 1 | x | x | x | 0 | S1 |
| S1 | x | x | x | 1 | x | 0 | S0 |
| S1 | x | x | 1 | x | x | 0 | S2 |
| S2 | 1 | x | x | x | x | 0 | S1 |
| S2 | x | x | x | 1 | x | 0 | S3 |
| S2 | x | 1 | x | x | x | 0 | S4 |
| S3 | x | 1 | x | x | x | 0 | S2 |
| S4 | x | x | x | x | 1 | 0 | S5 |
| S4 | x | x | x | x | 0 | 0 | S6 |
| x | x | x | x | x | x | 1 | S0 |

**Encoding Table**

|  |  |
| --- | --- |
| State | Encoding S2:0 |
| S0 | 000 |
| S1 | 001 |
| S2 | 010 |
| S3 | 011 |
| S4 | 100 |
| S5 | 101 |
| S6 | 110 |

**Next state table with binary encodings**

Current State Inputs Next State

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S2 | S1 | S0 | N | E | S | W | V | R | S2\* | S1\* | S0\* |
| 0 | 0 | 0 | x | 1 | x | x | x | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | x | x | x | 1 | x | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | x | x | 1 | x | x | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | x | x | x | x | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | x | x | x | 1 | x | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | x | 1 | x | x | x | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | x | 1 | x | x | x | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | x | x | x | x | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | x | x | x | x | 0 | 0 | 1 | 1 | 0 |
| x | x | x | x | x | x | x | x | 1 | 0 | 0 | 0 |

Next state equations:

S2\* = R’(S2’S1S0’E + S2S1’S0’V + S2S1’S0’V’) = R’(S2’S1S0’E + S2S1’S0’(V + V’))

S2\* = R’S2’S1S0’E + R’S2S1’S0’

S1\* = R’(S2’S1’S0S + S2’S1S0’W + S2’S1S0E + S2S1’S0’V’)

S0\* = R’(S2’S1’S0’E+ S2’S1S0’N + S2’S1S0’W+ S2S1’S0’V)

S0\* = R’S0’(S2’S1’E+ S2’S1N + S2’S1W+ S2S1’V)

**Output table with binary encodings**

Current State Outputs

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S2 | S1 | S0 | sw | d | win | S0 | S1 | S2 | S3 | S4 | S5 | S6 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

Output Equations:

sw = S2’S1S0

win = S2S1’S0

d = S2S1S0’

S0 = S2’S1’S0’

S1 = S2’S1’S0

S2 = S2’S1S0’

S3 = S2’S1S0

S4 = S2S1’S0’

S5 = S2S1’S0

S6 = S2S1S0’

Sword FSM

**Next state table:**

Current State Inputs Next State

|  |  |  |  |
| --- | --- | --- | --- |
| S0 | sw | R (reset) | S0\* |
| 0 | 1 | 0 | 1 |
| 1 | x | 0 | 1 |
| x | x | 1 | 0 |

S0\* = S0’swR’ + S0R’

**Output table:**

|  |  |
| --- | --- |
| S0 | v |
| 0 | 0 |
| 1 | 1 |

v = S0