# Vending Machine FSM Requirements

* Accepts farthings, F, Ha’pennies, H, and Pennies, P.
* Dispenses an item, , after 1.25 has been input
* Gives change
* Only one coin can be inserted at a time (F, H, or P)

# Determine Inputs and Outputs:

**Inputs:** Farthings, Ha’penies Pennies

**Outputs:** Item, Change

# State Transition Diagram: $ in the machine

S0

S1

S2

S3

**Number of States: 9**

**Bits of State Memory (# of D-FFs): 2ff 2 bits**

# State Transition Table

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Current State | Encoded State | Inputs | Next State | Encoded Next State | Encoded inputs |  |  |  |  |
| S0 | 0 0 0 0 | 0 0 0 | S0 | 0000 | 00 |  |  |  |  |
| S0 | 0 0 0 0 | 0 0 1 | S1 | 0001 | 01 |  |  |  |  |
| S0 | 0 0 0 0 | 0 1 0 | S2 | 0010 | 10 |  |  |  |  |
| S0 | 0 0 0 0 | 0 1 1 | x |  |  |  |  |  |  |
| S0 | 0 0 0 0 | 1 0 0 | S4 | 0100 | 11 |  |  |  |  |
| S0 | 0 0 0 0 | 1 0 1 | x |  |  |  |  |  |  |
| S0 | 0 0 0 0 | 1 1 0 | x |  |  |  |  |  |  |
| S0 | 0 0 0 0 | 1 1 1 | x |  |  |  |  |  |  |
| S1 | 0 0 0 1 | 0 0 0 | S1 | 0001 | 00 |  |  |  |  |
| S1 | 0 0 0 1 | 0 0 1 | S2 | 0010 | 01 |  |  |  |  |
| S1 | 0 0 0 1 | 0 1 0 | S3 | 0011 | 10 |  |  |  |  |
| S1 | 0 0 0 1 | 0 1 1 | x |  |  |  |  |  |  |
| S1 | 0 0 0 1 | 1 0 0 | S5 | 0101 | 11 |  |  |  |  |
| S1 | 0 0 0 1 | 1 0 1 | x |  |  |  |  |  |  |
| S1 | 0 0 0 1 | 1 1 0 | x |  |  |  |  |  |  |
| S1 | 0 0 0 1 | 1 1 1 | x |  |  |  |  |  |  |
| S2 | 0 0 1 0 | 0 0 0 | S2 | 0010 | 00 |  |  |  |  |
| S2 | 0 0 1 0 | 0 0 1 | S3 | 0011 | 01 |  |  |  |  |
| S2 | 0 0 1 0 | 0 1 0 | S4 | 0100 | 10 |  |  |  |  |
| S2 | 0 0 1 0 | 0 1 1 | x |  |  |  |  |  |  |
| S2 | 0 0 1 0 | 1 0 0 | S6 | 0110 | 11 |  |  |  |  |
| S2 | 0 0 1 0 | 1 0 1 | x |  |  |  |  |  |  |
| S2 | 0 0 1 0 | 1 1 0 | x |  |  |  |  |  |  |
| S2 | 0 0 1 0 | 1 1 1 | x |  |  |  |  |  |  |
| S3 | 0 0 1 1 | 0 0 0 | S3 | 0011 | 00 |  |  |  |  |
| S3 | 0 0 1 1 | 0 0 1 | S4 | 0100 | 01 |  |  |  |  |
| S3 | 0 0 1 1 | 0 1 0 | S5 | 0101 | 10 |  |  |  |  |
| S3 | 0 0 1 1 | 0 1 1 | x | x |  |  |  |  |  |
| S3 | 0 0 1 1 | 1 0 0 | S7 | 0111 | 11 |  |  |  |  |
| S3 | 0 0 1 1 | 1 0 1 | x | x |  |  |  |  |  |
| S3 | 0 0 1 1 | 1 1 0 | x | x |  |  |  |  |  |
| S3 | 0 0 1 1 | 1 1 1 | x | x |  |  |  |  |  |
| S4 | 0 1 0 0 | 0 0 0 | S4 | 0100 | 00 |  |  |  |  |
| S4 | 0 1 0 0 | 0 0 1 | S5 | 0101 | 01 |  |  |  |  |
| S4 | 0 1 0 0 | 0 1 0 | S6 | 0110 | 10 |  |  |  |  |
| S4 | 0 1 0 0 | 0 1 1 | x | x |  |  |  |  |  |
| S4 | 0 1 0 0 | 1 0 0 | S8 | 1000 | 11 |  |  |  |  |
| S4 | 0 1 0 0 | 1 0 1 | x | x |  |  |  |  |  |
| S4 | 0 1 0 0 | 1 1 0 | x | x |  |  |  |  |  |
| S4 | 0 1 0 0 | 1 1 1 | x | x |  |  |  |  |  |
| S5 | 0 1 0 1 | 0 0 0 | S0 | 0000 | 00 |  |  |  |  |
| S5 | 0 1 0 1 | 0 0 1 | S1 | 0001 | 01 |  |  |  |  |
| S5 | 0 1 0 1 | 0 1 0 | S2 | 0010 | 10 |  |  |  |  |
| S5 | 0 1 0 1 | 0 1 1 | x | x |  |  |  |  |  |
| S5 | 0 1 0 1 | 1 0 0 | S4 | 0100 | 11 |  |  |  |  |
| S5 | 0 1 0 1 | 1 0 1 | x | x |  |  |  |  |  |
| S5 | 0 1 0 1 | 1 1 0 | x | x |  |  |  |  |  |
| S5 | 0 1 0 1 | 1 1 1 | x | x |  |  |  |  |  |
| S6 | 0 1 1 0 | 0 0 0 | S0 | 0000 | 00 |  |  |  |  |
| S6 | 0 1 1 0 | 0 0 1 | S2 | 0010 | 01 |  |  |  |  |
| S6 | 0 1 1 0 | 0 1 0 | S3 | 0011 | 10 |  |  |  |  |
| S6 | 0 1 1 0 | 0 1 1 | x | x |  |  |  |  |  |
| S6 | 0 1 1 0 | 1 0 0 | S5 | 0101 | 11 |  |  |  |  |
| S6 | 0 1 1 0 | 1 0 1 | x | x |  |  |  |  |  |
| S6 | 0 1 1 0 | 1 1 0 | x | x |  |  |  |  |  |
| S6 | 0 1 1 0 | 1 1 1 | x | x |  |  |  |  |  |
| S7 | 0 1 1 1 | 0 0 0 | S0 | 0000 | 00 |  |  |  |  |
| S7 | 0 1 1 1 | 0 0 1 | S3 | 0011 | 01 |  |  |  |  |
| S7 | 0 1 1 1 | 0 1 0 | S4 | 0100 | 10 |  |  |  |  |
| S7 | 0 1 1 1 | 0 1 1 | x | x |  |  |  |  |  |
| S7 | 0 1 1 1 | 1 0 0 | S6 | 0110 | 11 |  |  |  |  |
| S7 | 0 1 1 1 | 1 0 1 | x | x |  |  |  |  |  |
| S7 | 0 1 1 1 | 1 1 0 | x | x |  |  |  |  |  |
| S7 | 0 1 1 1 | 1 1 1 | x | x |  |  |  |  |  |
| S8 | 1 0 0 0 | 0 0 0 | S0 | 0000 | 00 |  |  |  |  |
| S8 | 1 0 0 0 | 0 0 1 | S4 | 0100 | 01 |  |  |  |  |
| S8 | 1 0 0 0 | 0 1 0 | S5 | 0101 | 10 |  |  |  |  |
| S8 | 1 0 0 0 | 0 1 1 | x | x |  |  |  |  |  |
| S8 | 1 0 0 0 | 1 0 0 | S7 | 0111 | 11 |  |  |  |  |
| S8 | 1 0 0 0 | 1 0 1 | x | x |  |  |  |  |  |
| S8 | 1 0 0 0 | 1 1 0 | x | x |  |  |  |  |  |
| S8 | 1 0 0 0 | 1 1 1 | x | x |  |  |  |  |  |

# State Encoding Table Output Table

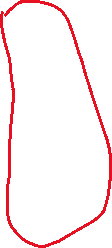
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 00 | Encoding |  | State Name | Encoded State | Output |
| S3 | 00 |  | S3 | 00 | 0 |
| S2 | 01 |  | S2 | 01 | 0 |
| S1 | 10 |  | S1 | 10 | 0 |
| S0 | 11 |  | S0 | 11 | 1 |

# Next State Logic





A picture containing text, crossword puzzle

Description automatically generated

# Output Logic