

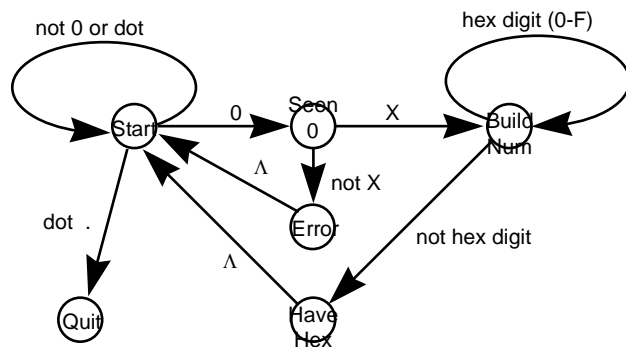
A Finite State Machine Example

PROBLEM

Construct a Finite State Machine (FSM) that recognizes a series of C++/Java hexadecimal numbers (CHN) separated by whitespace. The series ends with a dot (a period). A CHN is: $0X<\text{hex digit}>^+$ where $<\text{hex digit}>$ is one of 0-9,a-f,A-F and the $^+$ means “one or more.” Don’t worry about the case where there are no $<\text{hex digit}>$ s – accept the number anyway. (This means that we’ll actually accept $0X<\text{hex digit}>^*$ – where * means “zero or more.”)

A HEX NUMBER FINITE STATE MACHINE RECOGNIZER

From the definition above we may develop the FSM shown just below.



This FSM has 6 states:

- Start** – the initial state
- Quit** – the final state
- Seen 0** – just saw a leading zero
- Build Num** – just saw an ‘X’; now building a hex number
- Have Hex** – completed a hex number; output it
- Error** – ill-formed hex number

We can cast the FSM into its equivalent State-See-Do tabular form by inspection. From there it’s an easy step to derive a Structured Flowchart and Pseudocode. We show a version below...

| State\See | Λ | ~ 0 or . | 0 | dot | X | ~ X | HexDigit | ~ HexDigit |
|-----------------|-------|----------|-------|------|----------|-------|----------|------------|
| Start | | Start | Seen0 | Quit | | | | |
| Quit | | | | | | | | |
| Seen0 | | | | | BuildNum | Error | | |
| BuildNum | | | | | | | BuildNum | HaveHex |
| HaveHex | Start | | | | | | | |
| Error | Start | | | | | | | |

void Recognizer (void)

```

aState ← Start
Loop
  Switch on aState:
    HaveHex: Output "Number";
              Start
    Error:   Complain;
              Start
  cChar ← NextChar
  if ~ whitespace then echo cChar
  aState ← Transition (aState, cChar)
Until State = Quit
    
```

State Transition (aState, cChar)

```

Switch on aState: ("See" means is aChar =?)
Start:
  See 0, aState ← Seen0
  See dot, aState ← Quit
  else, aState ← Start
Seen0:
  See X, aState ← BuildNum
  else, aState ← Error
BuildNum:
  See a HexDigit, aState ← BuildNum
  else, aState ← HaveHex
default:
  Should never get here
Return aState
    
```