

State Diagrams

- Like sequential digital circuits
  - Remember EE-290?
- Concepts
  - Event
  - State
  - Transition
- UML notation is similar
  - But has enhanced capabilities

---

---

---

---

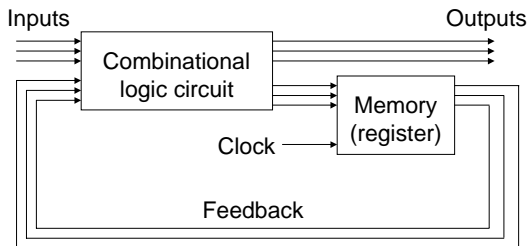
---

---

---

---

Synchronous Sequential Machine




---

---

---

---

---

---

---

---

2-bit Counter State Table

Sequence	Current State		Input	Next State		Output
	C1	C0	E	C1	C0	T
1	1	1	0	0	0	0
1	1	0	0	0	1	0
0	0	0	0	1	0	0
0	0	1	0	1	1	1
1	1	1	1	1	1	0
1	1	0	1	0	0	0
0	0	0	1	0	0	0
0	0	1	1	1	1	0
0	0	1	1	1	0	0

---

---

---

---

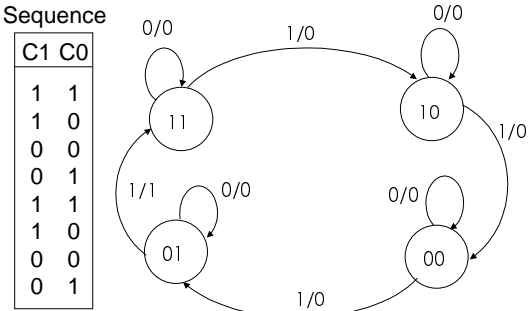
---

---

---

---

2-bit Counter State Diagram




---

---

---

---

---

---

---

---

UML State Diagrams

- **Based on Harel state charts**
  - David Harel (1987)
  - Permit nested states
- **Associated with classes**
  - Describe object state
- **Used in analysis and design**
  - Primarily a documentation tool
    - No Rose code generation (yet?)

---

---

---

---

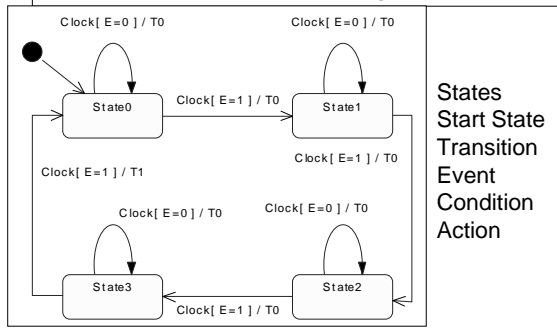
---

---

---

---

UML Counter State Diagram




---

---

---

---

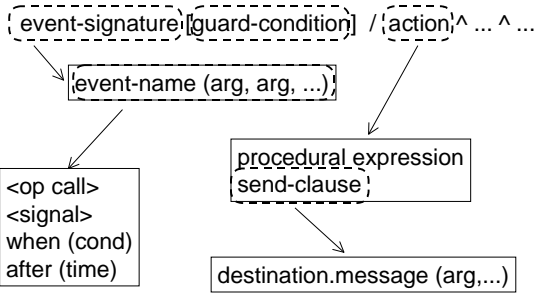
---

---

---

---

### Transition String Details




---

---

---

---

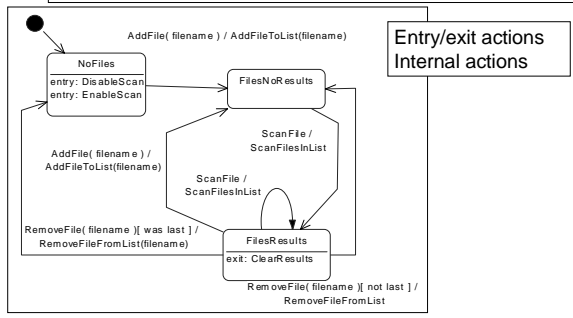
---

---

---

---

### File Scan Example




---

---

---

---

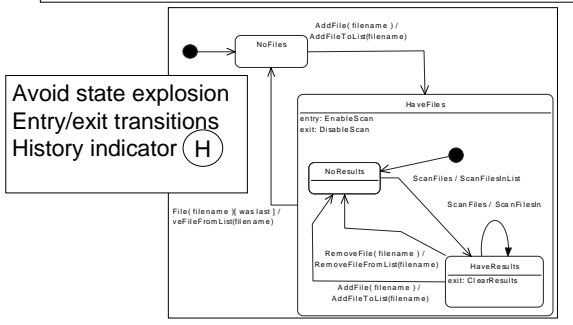
---

---

---

---

### Nested States




---

---

---

---

---

---

---

---

### Implementing State Machines

- **Distributed conditional logic**
  - “If” or “switch” in member functions
- **State machine interpreter**
  - Separate member object?
  - Call from member functions
- **Use State pattern (page 406)**
  - Base class and derived for each state
  - Behavior variants as virtual functions

---

---

---

---

---

---

---

---

---

---

### Distributed Conditional Logic

```
Project::AddFile(fn)
{
  switch (state)
  {
  case NoFiles:
    state = NoResults;
    AddFileToList(fn);
    break;
  case NoResults:
    ...; break;
  case HaveResults:
    ...; break;
  }
}
```

What's wrong with this approach?

---

---

---

---

---

---

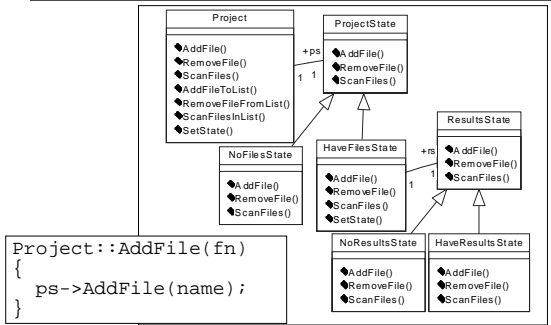
---

---

---

---

### State Pattern




---

---

---

---

---

---

---

---

---

---