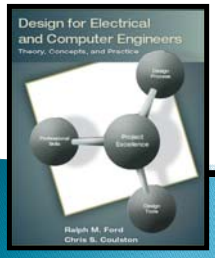
Chapter 6 – System Design II: Behavioral Models



5.5 Application: Digital Design – Stopwatch

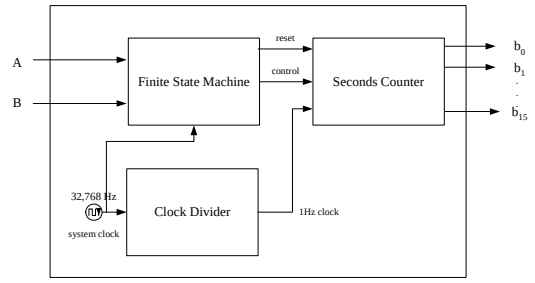
Requirements: The system must

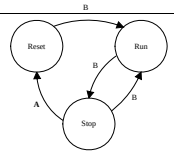
- have no more than two control buttons;
- implement three functions (run, stop, and reset);
- output a 16 bit binary number that represents seconds elapsed.

Level 0

Module	Stopwatch
Inputs	- A = Reset button signal. When the button is pushed it resets the counter to zero. - B = Run/stop toggle signal. When the button is pushed it toggles between run and stop modes.
Outputs	- b_0-b_{15}. 16 bit binary number that represents the number of seconds elapsed.
Functionality	The stopwatch counts number of seconds after the run button (B) is pushed and system is either in reset or stop mode. When in run mode and the stop button (B) is pushed, it stops the count. A reset button (A) push will reset the output value of the counter to zero only when in stop mode.

Level 1



Module	Finite State Machine
Inputs	-A. Signal to reset the counter signal. -B. Signal that provides a toggle between run and stop modes.Clock. 1Hz clock signal.
Outputs	●Reset. Signal to reset the counter to zero. ●Control. This signal enables or disables the counter.
Functionality	

Motivation

- Functional Design
  - Appropriate for function-oriented systems: inputs, outputs, and some transformation between them.
- There are other types of *system behavior* that designers need to be able to understand.
  - State behavior
  - Logic and flow
  - Data flow
  - Database relationships
  - …

Learning Objectives

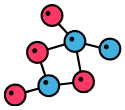
By the end of this chapter, you should:

- Have a familiarity with the following modeling tools for describing ECE system behavior: state diagrams, flowcharts, data flow diagrams, entity relationship diagrams, and the Unified Modeling Language.
- Understand the intention and expressive power of the different models.
- Understand the domains in which the models apply.
- Be able to conduct analysis and design with the models.
- Understand what model types to choose for a given design problem.

6.1 Models

Models – what do you think of?









Properties of Models

A good model should be

- Abstract
- Unambiguous
- Allow for innovation
- Standardized
- Facilitate good communication
- Modifiable
- Remove unnecessary details & show important features
- Break system into sub-problems.
- Substitute sequence of actions by a single action.
- Assist in verification
- Assist in validation

Definitions

Model

Modeling Language

Object Type

Intention

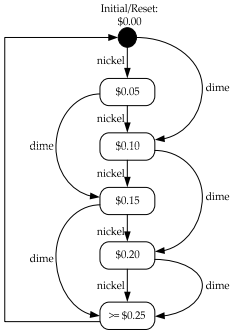
6.2 State Diagrams

Intention is ?

How do you know if you should use one?

State Diagram Symbols

Example: Vending Machine



Example: Better Vending Machine

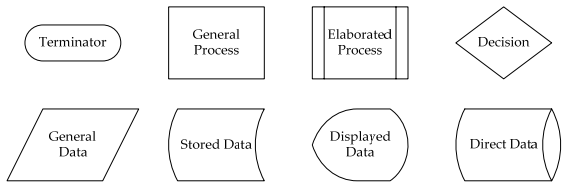
Consider the state diagram for the vending machine shown on previous page Figure 6.2. Now assume that the system accepts nickels, dimes, and quarters. Also assume that it is capable of returning change to the user after a purchase. Create a state diagram that represents this new system. Make sure to define the output signals and their value for each state.

Solve this individually in class – 5 minutes

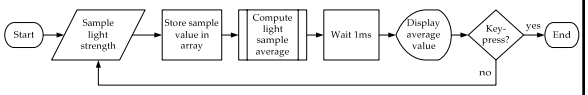
6.3 The "Lowly" Flowchart

The intention of a flowchart is to model what type of behavior ???

Flowchart Symbols

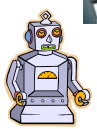


Example: Light Monitoring System



- Can you figure out the operation of the system by looking at this?
- That is why the flowchart is elegant and not so lowly.

Example: Security Robot



Requirements (loosely)

- Must roam randomly around facility
- Detect intruders by recognizing sound
- Set-off an alarm if detects noise, transmit position, and wait.
- Must regularly conduct a self-test to determine if it is working properly.

Design Details

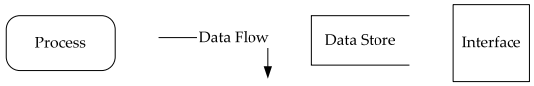
- Has the three ultrasonic sensors and can measure distance to objects to the left, forward, and right.
- Has a microphone that it uses to monitor sounds.

Problem: Develop a Flow Chart of its operation

6.4 Data Flow Diagrams

Intention is ???

- DFDs can have levels, just like the functional

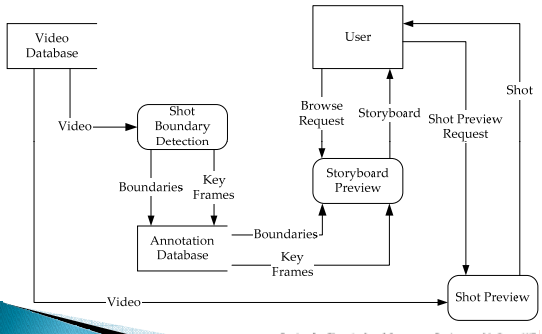


Example: The Pie Making Process

Example: The Pie Making Process

Example: Video Browsing System

Example: Video Browsing System



DFD – The Event Table

Event	Trigger	Process	Source
Annotate Video	New Video Arrival	Shot Boundary Detection	System
View Storyboard	Browse Request	Storyboard Preview	User
View Shot	Shot Preview Request	Shot Preview	User

6.5 Entity Relationship Diagrams

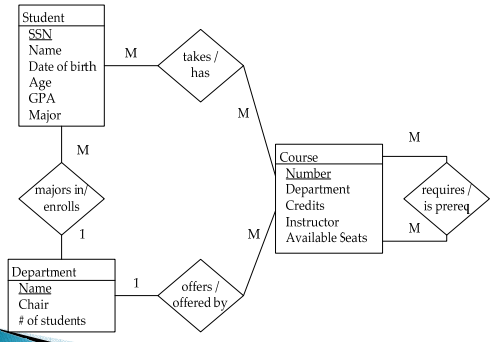
- Intention of an ERD is
- Entities =
- Relationships =
- Attributes =

ERD Symbols

Example: College Database System

	Student	Course	Department
Student			
Course			
Department			

College Database ERD



6.6 Unified Modeling Language

- For object-oriented software design.
- Value in applying it to ECE Systems.
- Has 6 different views of systems (unified!).
- Only an overview is provided here.

UML – Scenario

- Pretty popular idea – web ordering of groceries followed by home delivery.
- The "v-Grocer" system.
- User has a barcode scanner connected to home computer.
- They can scan a used item an automatically order it from the grocery store.
- Place the order and groceries delivered at pre-arranged time.

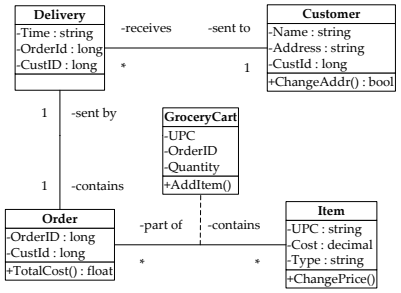
Static View

- Object view of software.
- ***Classes*** represent
  - Data
  - Methods (functions) that operate on the data
- ***Objects*** are

Customer
-Name : string -Address : string -CustId : long
+ChangeAddr() : bool

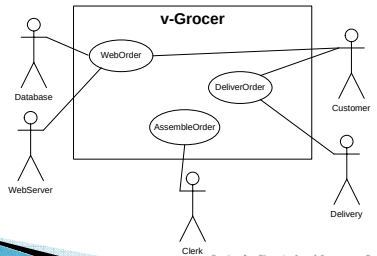
- Can allow for different security levels.

Class Diagram



Use-Case View

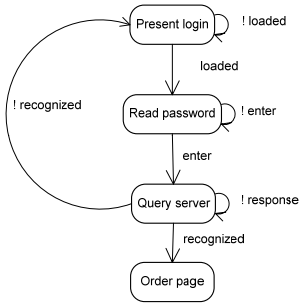
- Intention =
- Characterized by a Use-Case Diagram



Use-Case Description

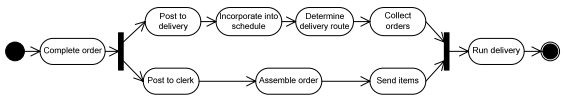
Use-Case	`WebOrder`
Actors	`Customer`, `Database`, and `WebServer`
Description	This use-case occurs when a customer submits an order via the `WebServer`. If it is a new customer, the `WebServer` prompts them to establish an account and their customer information is stored in the `Database` as a new entry. If they are an existing customer, they have the opportunity to update their personal information.
Stimulus	Customer order via the `GroceryCart`.
Response	Verify payment, availability of order items, and if successful trigger the `AssembleOrder` use-case.

State Machine View



Activity View

- Intention = describe a sequence of activities needed to complete a task.



Interaction View

- Intention = to show interaction between objects (when they must cooperate to do something useful).
- Use either a ***collaboration*** or ***sequence*** diagram.
- This example is for the **WebOrder** use-case.

Physical View

- Show the physical components that constitute the system.
- Can think of this much more generally than presentation in UML.

6.7 Project Application: Selecting Models

- See Table 6.4 of book.
- Gives guidance on how to select models based upon behavior to describe.

6.8 Summary

- Models are an abstraction of system.
- Models can be thought of as a design specification.
- Models have different intentions for describing behavior.
- Models should encourage innovation and provide for clear documentation.