Design a First View

Overview

In this lesson we will

- ✓ Discuss an introductory approach to the design process.
- ✓ Examine a graphical view of design.
- ✓ Examine signals and components in a design.
- ✓ Introduce how signals and components can be expressed.
- ✓ Examine how to bring the pieces together in a block diagram.
- ✓ Introduce the concepts of datapath and control path.

Introduction

Will begin by considering problem of problem solving
Not difficult but requires creative thought guided by discipline
In the abstract
    All problems are similar

Begin by exploring a summer holiday across Europe
Several questions immediately come to mind
    Where should you go
    What should you see
    How can you get around
Seems that you have a problem
    How do you begin to attack it
        Where are we now
        Where are we going
        What is missing
        Do we have a high level structure

One quick and easy way to solve the problem is to subcontract Rick Steves
    He certainly has a lot of experience
    Can easily solve the problem
Not as much fun as doing it yourself though

So once again how do we attack the problem
    Let’s walk through early stages of doing so

Early Planning

Planning such a trip

Follows same general approach as we use for design

As we begin process

We think about

What we want to do

These first ideas are a glint in the eye

These thoughts and ideas become basis for rest of planning

As we think and do more research – read books or watch travel films

We begin to think in greater detail

We propose a collection of things that we'd like to see or do

We talk with friends

We change our plans based upon their ideas and suggestions

We talk with more friends

We change our plans again

Repeat the process until pretty comfortable

Ultimately we have pretty firm list of things to do

More research

More details

Finally after days of weeks of planning

We have a trip set

Bags get packed

Off we go

As we travel – we test our planning

If we did a good job with our planning

We have a great trip

Travel to Design

When we approach a new design

- ✓ Our first step is to identify the requirements
  - We talk with our customer
    - To determine what he or she wants
      - Talk about features
      - How system expected to behave
- ✓ Move from qualitative description
  - To formal – quantified specification

- ✓ From specification
  - Try to identify major pieces of functionality
    - Divide those into smaller and more detailed pieces
- ✓ From detailed functional description
  - Decide what hardware and software
    - Will use to implement each piece of functionality
- ✓ We test our system to ensure that it behaves
  - Way that customer wanted
- ✓ Sell finished design to customer

One major difference between design and trip is need for planning

A reasonable and fun, challenging, and exciting way to plan trip

Total serendipity

That is no planning

Go and let things happen

Cool way to travel - not a good way to approach design

In design planning in detail

Right road to successful design – to successful project

Let's now walk through early stages of process

See how things unfold

Let's look at the major pieces of functionality

Back to Travel - Where to go

One easy first step to take

Get one of the full scale maps of Europe that can open on kitchen floor

…and some guide books

Then pick the major cities and other places you want to go

Because you have limited time

Don't want to retrace your steps

Where's that traveling salesman when we need him

Decide to start in London

Tour bits of England before heading across channel

For grand tour of continent

Thereafter choose to do loop through

Northern Europe

Head south through Italy

Back through Paris and return to London

You draw your tentative route on your map

So you can show and recruit friends

What to see…What to do

- Food new kinds of food
  - This is always good
  - You've heard that French is excellent
  - Who can go to Italy without trying fresh seafoods
  - Germany – ah the wursts are the best
- Art
  - You like art
  - Who isn't intrigued with the renaissance period
  - Modern art is fascinating
- Plays
  - London theatre is excellent
  - How about Shakespeare
  - Outdoor theatre
- Music
  - Music is always fun
  - The jazz scene in London and Paris is excellent
  - Street music everywhere
  - Opera in Italy
    - Well ….hmmm…then maybe not
- Museums
  - Old stuff is interesting
  - London is filled with them
  - How about Munich
  - Certainly Italy
- Castles
  - Everyone likes castles
  - Scotland
    - Hey wasn't Macbeth up there somewhere
  - Germany
    - Along the Rhine
  - France
    - Wow the Loire valley

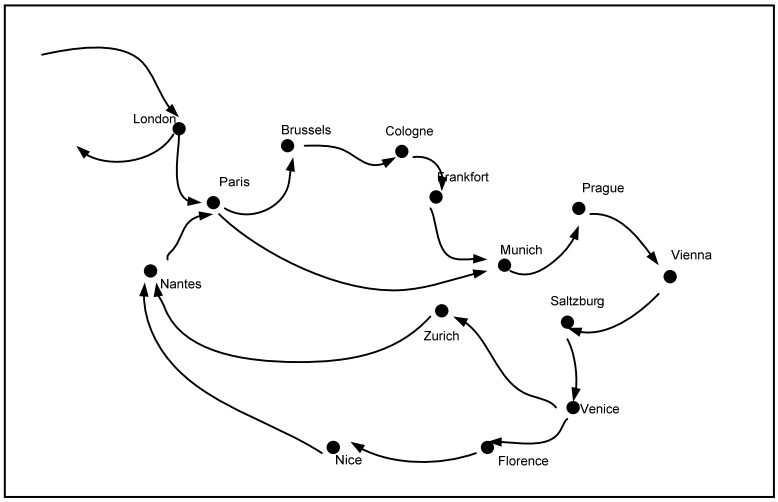
How to get Around

- You're on a roll
- Plans are moving right along
- Transportation however can be tricky
  - You don't want to hire a car

Hey Europe has good trains
What's this thing called a Eurail pass
Busses what about busses
Do they have ferries
Those can work
How about walking

You finally have your trip planned
…and a friend to join you
You both have some things that
You want to see together
Others that you want to explore on your own

Your high level itinerary now looks something like this



From Travel Back to Design Again

All of the thought and planning put into trip
Parallels that necessary for
Moving design from idea to implementation

Taking rather high level view for moment
Planned route analogous to what is called *datapath* in system
What route does data entered into system
Travel to reach output
Things to see and do metaphorically map to
Components along datapath
- Registers

- Multiplexors
- Selectors
- Computational circuitry

Transportation becomes *control elements*
Orchestrating flow along datapath

Thus like trip elements of design comprise
- ✓ Datapath
- ✓ Datapath components
- ✓ Control of movement along datapath

We will revisit each of these
As we examine and develop design process
First examine some of the tools we'll use
Relate each back to design elements

Design a Graphical View

You've just come up with fantastic idea
For new product no one has ever thought of
How do you tell someone else about it

Are many many ways

- Certainly could build one
  Take it around and show people
  This can be a lot of work
  May have good idea
  But miss a few important things
  That someone else might have seen

- Try to describe it using spoken words
  Another possibility
  Can certainly get and idea around quickly
  How well does this work for big complex ideas
  What if
  You tell two people or three
  Each telling gives a slightly different description
  Some tellings leave out important details

- Try to describe it using written words
  Yet another possibility
  Goes long way to addressing problems with spoken words
  Can be rather affective
    Providing people will read

- Describe it mathematically
  Sets of equations describing idea
    Good for design
    Poor for conveying idea

- Try to describe it using pictures
  Approach worked well for our ancestors
    They drew pictures or pictographs
      On sides of caves and later clay tablets or walls
  As humans we tend to grasp graphical presentation
    More quickly than text
  Let me draw you a picture
    Expression commonly used during an explanation

At the end of the day
  Best approach is probably combination of all
  During different phases of design process
    Will different methods to capture, utilize, relate knowledge
  As engineers
    We often use graphical view during early phases

  Let's examine that view
    Discover some of the graphical
      Symbols and techniques we might use

Components and Signals

At high level our designs comprised of two things
- ✓ Components, pieces, entities, objects
- ✓ Signals that flow into, out of, between them

Initially components and signals
  Expressed at very high level
As design evolves
  Expression becomes more detailed

May ultimately morph into
Specific interconnecting wires
Mathematical equations
Words in an HDL modeling language or software program
Silicon paths in an integrated circuit

Expressing Signals

Let's begin at high level with signals
In early descriptions of our design
Goal is to convey central idea
While hiding non-essential details
We abstract our concepts

Key early notions we wish to convey

- Movement of collections of signals
- Collections may be of different sizes
  Movement of subsets of collections
  Need to express this clearly
- Movement of single signals
- Movement
  Into, out of, or into or out of component

We express signals using a line

- ✓ *Width* of line to give approximation of number of signals
  Wide line for bus or aggregation of signals
  Narrower line to express a subset
  Thin line to express single signal
- ✓ *Direction* indicated by arrow head
- ✓ *Annotate* lines
  Description or nature of signals conveyed

During later stages of design
Use similar line notation
Extend annotation to reflect
Number of comprising signals

Referring back to our trip
Signal paths through system being designed
Analogous to route traced out on map
Early in planning

Talk in general terms

Need to get from one city to another

No specific method names

As planning evolves

Level of detail increases

Select train to move between cities

Within city bus or taxi to destination

At destination

Choose to walk or take bicycle

As we design

Initially we focus on abstract

As design develops

Focus on details

A High Level View

As with the trip

Will begin with a high level set of tools

Let's look first at movement between places

Will ignore places for moment

Within a design we move signals

- As an aggregate we call a *bus*
- As an individual signal

Busses may reflect

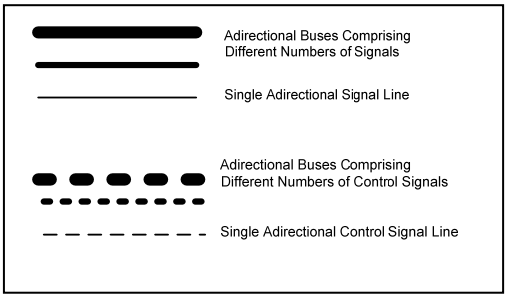
Signals moving

- One direction
- Two directions

Signals expressing

- Data
- Control

We begin with these basic ideas as illustrated in following graphics



As graphic shows
- Distinguish
  - Data and control flow
    - Solid vs. dotted or dashed line
  - Relative number of signals comprising bus
    - Width of bus

When such signals / busses expressed in diagram
- We annotate each with brief description of
  - Nature of comprising signals
    - Annotation can be general such as
      - Address Bus
      - Data Bus
    - More specific such as
      - Image Data
      - High Alarm Warning

Let's now look at collection of signal paths
- As might appear in diagram reflecting more detail

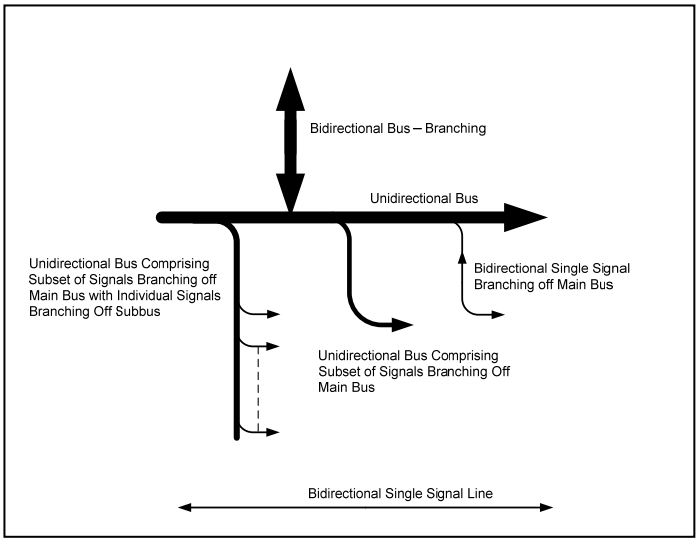


Diagram illustrates different
- Sizes and directions of data flow
  - Control flow would be similarly illustrated
    - Utilizing dotted or dashed line segments

Directional arrowheads

Could easily be added or removed

As appropriate in system being designed

Once again busses would be appropriately annotated

Reflecting nature of comprising signals

Next level of refinement

Provides specific information as to number of comprising signals

Signal cardinality in bus

Indicated by labeled forward slash

Crossing bus

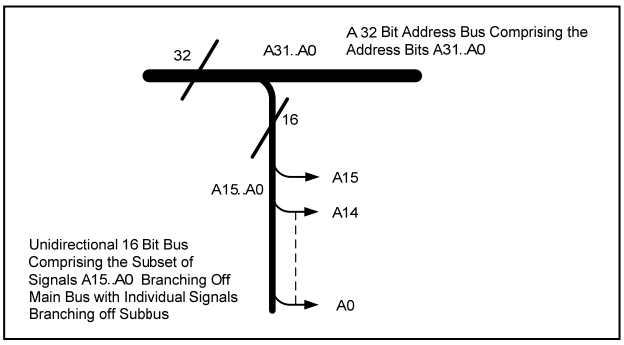
Further annotation in graphics below written in

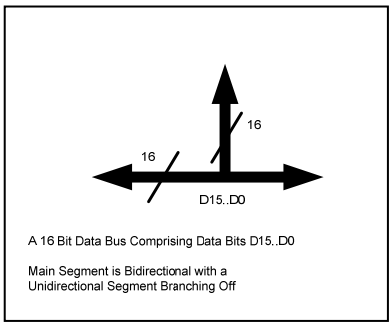
Big endian notation

Suggests uppermost or leftmost individual wire

Is *most* significant

A31 or A15 in this example





If annotation written as

A0..A31 or A0..A15

Little endian notation assumed

Suggests uppermost or leftmost individual wire

Is *least* significant

Back to Travel - Expressing Components – Part 1

- Returning to the journey once again
- Initial planning identified
  - Places and things to see and do
  - Set of things to do initially - our requirements
    - These are high level categories
      - Food
      - Art
      - Plays
      - Music
      - Museums
      - Castles
- As planning evolved
  - Each general category
    - Refined to specific types of each

Back to Design - Expressing Components – Part 1

- Design follows same initial pattern
  - Requirements Specification
    - High level *what*
  - Formal Specification
    - Detailed *what*

- Getting to work
  - Move to *how*

  - Rather than places to go or see
  - Design expresses such concepts as
  - Pieces of functionality - not F(x) type
    - *Functions*
  - Called *functional decomposition*

- As we begin design
  - We try to understand major kinds of functionality
    - That comprise system's behavior

- Our first cut at the design is at functional level
  - We identify
    - ✓ Main top level functions
    - ✓ Communication between such functions

Identical process to object oriented approach
Some tools work whether in HW or SW

Once those identified
✓ Look at lower level functions that
Implement that behavior or functionality

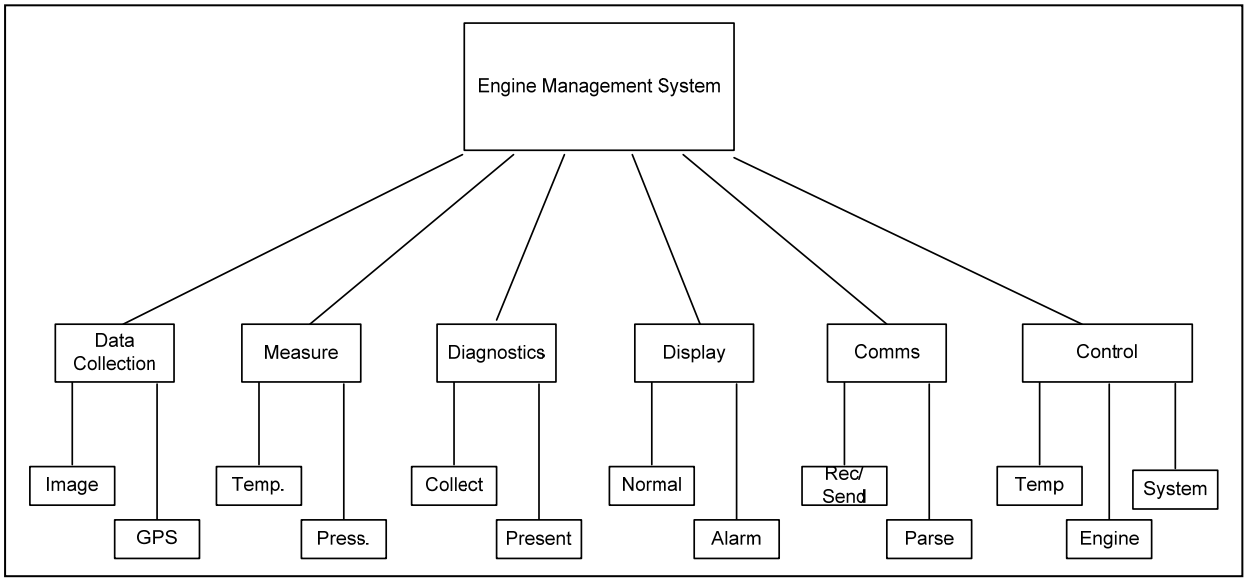
As long as we're traveling
Consider design of an engine management system
For TGV train in France – the new one

Main top level functions - main things we need to do

- User interface Display
  - Supporting all user input and output
    - Information managed as
      - Normal and alarm or warning information
- Measurement
  - Collecting
    - Pressure and temperature data
- Data Collection
  - Supporting functionality such as
    - Image collection
    - Position information
      - From GPS subsystem
- Communications
  - For sending and receiving commands
    - From base station
  - Interpreting commands when received
- Diagnostics
  - Implements functionality associated with
    - Collecting
    - Interpreting and displaying relevant information
- Control
  - System
  - Temperature within passenger cabin
  - Engine

Each top level function
Made up of variety of related pieces of functionality

Portion of functional hierarchy given in following graphic
Graphic below does not include
Nature or extent of communication
Between functions



Expressing Components – Part 2

As our trip planning matures
We decide which
Restaurants, clubs, or musical venues
Want to try to see

Similar process occurs with design

Once we have captured major functional blocks
As well as specific pieces of functionality
Make up each of those blocks
Move to formalize design - think about architecture
Take first cut at the architecture

First cut at such an architecture
Expressed in block diagram

Block diagram can be used to express architecture
    Software or hardware components

Typically hardware block diagram occurs first
    As with other diagrams
        Initial version may abstract away most details
            Focus on
                - Major physical blocks of functionality
                - Signal flow between or among such blocks
        First cut at identifying
            What is hardware
            What is software

Structure
    High level structure of block diagram
        Signal I/O and flow
            Inputs generally on right
            Outputs generally on left
            Signal flow is
                Left to right
                Top to bottom
    Signals into and out of modules
        Follow same pattern

Components
    Components included in block diagram
        Circuit elements as appropriate
            Diodes, transistors, LEDs, relays, connectors etc.
            Used only in limited cases for clarity
        Boxes or blocks
            Used to indicate
                Modules or more complex pieces of functionality

The Block Diagram

Following block diagram

Gives high level architecture for system that must

- Capture and store image data
  - Digital and software
- Sample several analog signals
  - Convert these to digital form
    - Analog to digital converter
  - Store the results
    - Memory of some kind
- Receive GPS information
  - External sensor
- Send and receive serial information and data
  - Communication system
- Control
  - System control
  - Manage and coordinate operation
  - Composite elements
  - Specific control
  - Motor speed
    - Based upon pressure measurements
  - Heating, ventilation, and air conditioning
    - Using measured temperature data
- Display information
  - Normal information
  - Extraordinary condition annunciation

Components interconnected via system bus

System bus components not elaborated

At current stage of design

Initially high level view

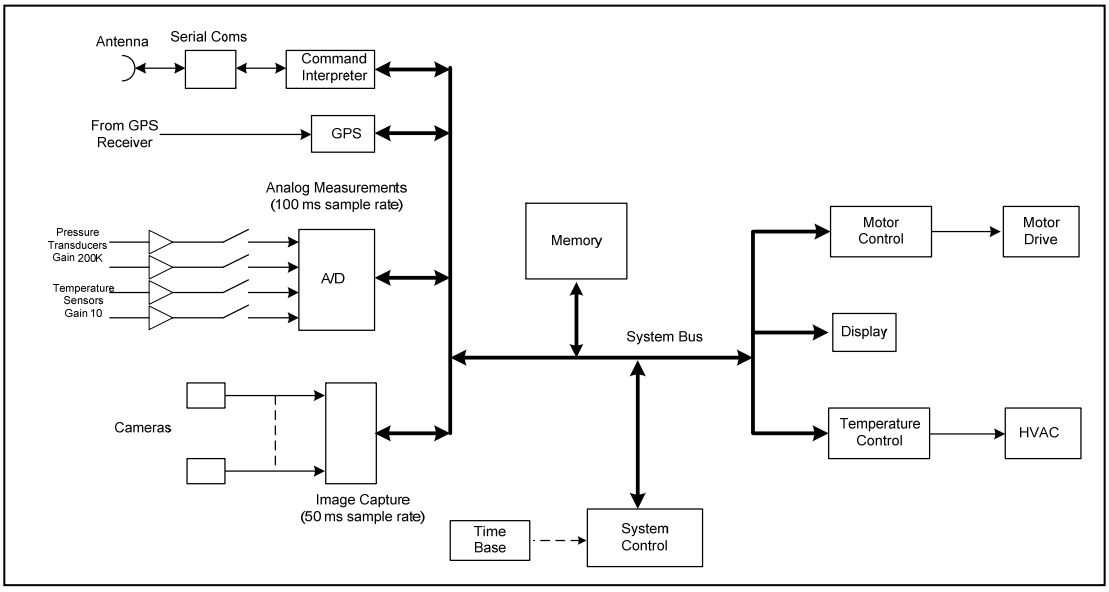
Elaboration can occur if necessary

As design is refined

Specifically observe

Data and control flows not distinguished

Moving to RTL - *Register Transfer Level - Data Flow*



Observe that

For most part

- Inputs appear on left
- Outputs appear on right

Exception is serial communications

Module supports bidirectional communication

Grouped with inputs subset

Visual grouping

Serial communication and GPS blocks

Analog measurement and image capture

Perform similar types of function

Grouped together

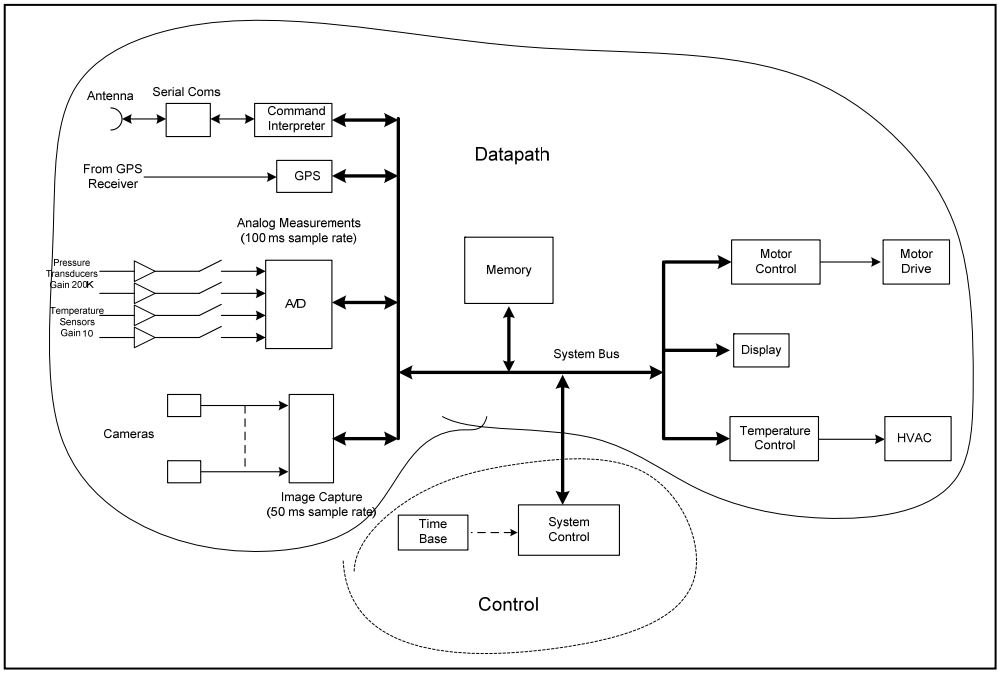
Use such a scheme to clarify presentation

Refining the Design

The Datapath and Control

Examine previous block diagram

In greater detail



We find that there are two major subsections in diagram

- ✓ Datapath
- ✓ Control

Datapath enclosed in larger circled area

Data flows into system

From several sources

Takes different paths through system

Ultimately appears as output as

Display

Communications

Two control functions

Control expressed in circled area comprising

Time base

System control

As data moves along datapath
- Affected by number of common operations
  - Store
  - Transfer
    - Select
    - Direct or route
  - Modify
- Operations affected by components
  - Data passes through

How we
- Connect datapath components together
- Manage control of those components
  - These are our busses

- To enable us to solve problem
  - According to specification
  - Reliably
  - At low cost
  - Efficiently
- What we call creativity
  - This is the challenge
  - This is the exciting and fun part of engineering

Datapath
- Datapath comprises
  - Route(s) data takes through system
  - Components through which data passes
    - As it moves through system

- Data Path Components
  - Today designs often incorporate
    - Analog as well as digital components
    - Transport mechanism

Such components include
- Digital
  - SSI and MSI type functions
    - Mix of gate level and RTL
    - Registers
      - Basic storage
        - Parallel in – parallel out
      - Shift
        - Serial / parallel load
        - Shift left, right, or both
        - Polynomial – LFSR
      - Files
    - Multiplexers and demultiplexers
    - Encoder / decoder
    - Arithmetic
    - Logical
    - Comparison
    - Counters
      - Up – down
    - Timers – Dividers
    - Communication

  - LSI and VLSI type functions
    - Discrete packaged objects
      - Memory
      - Special purpose processors
        - DSP
        - Graphics
    - IP
      - Form of soft cores
      - HDL modules
- Analog
  - Small scale
    - A/D – D/A converters
    - Multiplexers
    - References
    - Buffers – Amplifiers
    - Level shifters

- Large scale
  - Receiver – transmitter
  - Communications
  - Power management

- Infrastructure and Transport
  - Under this category of components we include
    - Physical means by which signals flow
      - Into, out of, or through system
  - Here such components include
    - Input and output ports
    - Busses or physical interconnections between components

- Mechanical and miscellaneous
  - Many of today's designs extend beyond
    - Traditional analog and digital components
  - Not unusual to find
    - Mechanical
      - Switches
    - Electromechanical
      - Relays or motors
    - Other types of hardware
      - Lights
      - Displays
      - Audio capture or presentation

Control

- Components along Datapath
  - Perform tasks of
    - Moving, manipulating, storing data
- How, when, where, and how often such operations occur
  - Responsibility of control portion of system
- Control very often based upon
  - Some form of FSM or
  - Collection of FSMs

Control Strategy

Two elements critical to formulating effective control strategy

Resources

Time

Let's examine each of these

Resources

Resources are components found along datapath

May have single or multiple copies of each resource

Some resources may be sharable others may not

In current context

- Sharable means

  Can be reused in/by different tasks at different times

  Example

  Register or I/O port

  Buss

- Nonsharable means

  The resource maintains state that can't be replicated

  Resource cannot reasonably be shared in design

  And or Or gate

  Dedicated I/O port or buss

Time

Two aspects of time important

- ✓ Coarse grained

  Constraints on functional timing

  Schedule and deadlines

- ✓ Fine grained

  Constraints on signal timing

  Delays and race conditions

Summary

In this lesson we

- ✓ Discussed an introductory approach to the design process.
- ✓ Examined a graphical view of design.
- ✓ Examined signals and components in a design.
- ✓ Introduced how signals and components can be expressed.
- ✓ Examined how to bring the pieces together in a block diagram.
- ✓ Introduced the concepts of datapath and control path.