

Block Diagrams – A Quick Overview

University of Washington - Department of Electrical Engineering
James K. Peckol

Introduction

This document gives a quick overview of constructing a block diagram for a system that we may design. Such a diagram gives us an early picture of the design of our system.

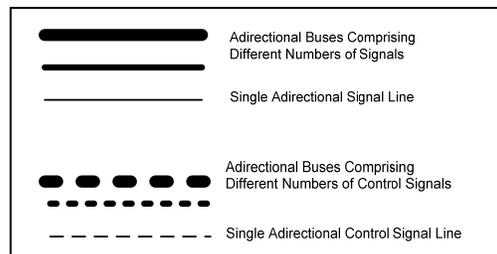
The information that we wish to identify and capture through a block diagram includes the following,

1. The signals flowing into and out of our system.
2. The major high-level functional blocks that comprise the internal architecture of our system.
3. The signals flowing among the major functional blocks.

Signals

Let's look first at the signals. Within a design we move signals as an individual signal or as an aggregate that we call a *bus*. Busses may reflect signals moving in one or two directions. Such signals express data and control.

We begin with these basic ideas as illustrated in following graphics



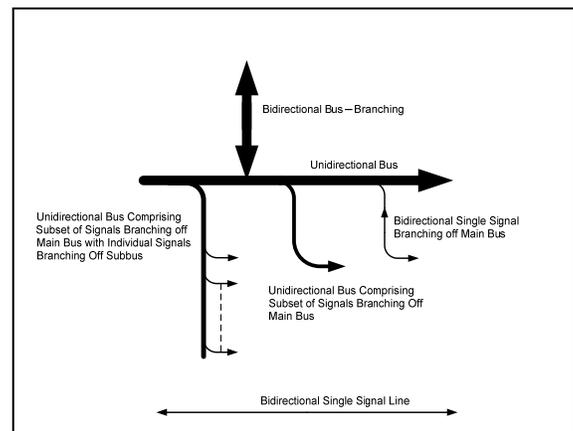
As the graphic shows, we distinguish between data and control flow using a solid or dotted or dashed line. We identify the relative number of signals comprising a bus – the width of the bus

When such signals / busses are expressed in a diagram, we annotate each with brief description of the nature of the comprising signals. Annotation can be general such as

- Address Bus
- Data Bus

or more specific such as

- Image Data
- High Alarm Warning



Let's now look at a collection of signal paths as they might appear in diagram reflecting more detail. The diagram illustrates the different sizes and directions of the 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 the nature of comprising signals.

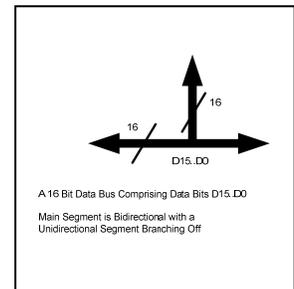
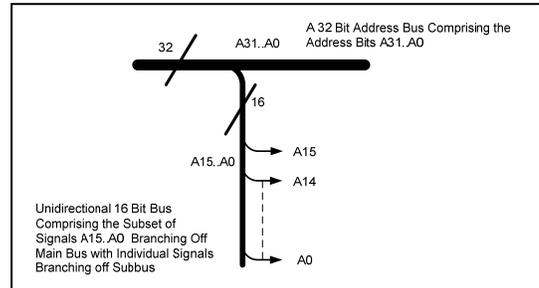
The next level of refinement provides specific information as to number of comprising signals. The signal cardinality in a bus is indicated by labeled forward slash crossing the bus.

Further, the annotation in the graphics below is written in *big endian* notation. The annotation suggests that the uppermost or the leftmost individual wire is the *most* significant, A31 or A15 in this example.

If the annotation written as:

A0..A31 or A0..A15

little endian notation assumed and the notation suggests uppermost or leftmost individual wire is the *least* significant.



Expressing Components

Consider design of an engine management system for a TGV train in France – the new one. The main top-level functions include,

User interface Display

Supporting all user input and output. Information is managed as normal and alarm or warning information.

Measurement

Collecting pressure and temperature data.

Data Collection

Supporting functionality such as image collection and position information from a GPS subsystem.

Communications

For sending and receiving commands from the base station and interpreting commands when received.

Diagnostics

Which implements functionality associated with collecting, interpreting and displaying relevant information.

Control system

Manage temperature within passenger cabin and the engine.

Each top-level function may be made up of a variety of related pieces of functionality. Once we have captured major functional blocks as well as specific pieces of functionality that

make up each of those blocks, we move to formalize our design. That is, we take the first cut at the architecture. That first cut is expressed in block diagram.

It is important to note, that a block diagram can be used to express the architecture of our software or hardware components. Typically, though, the hardware block diagram occurs first.

As with other diagrams, the initial version may abstract away most details and focus on the major physical blocks of functionality and the signal flow between or among such blocks.

Structure

The high-level structure of a block diagram expresses signal I/O and flow. Inputs are generally on the left and outputs are generally on the right. Signal flow is left to right and top to bottom. Signals into and out of modules follow same pattern

Components

Circuit elements such as diodes, transistors, LEDs, relays, connectors etc. may be included in limited cases for clarity.

Boxes or blocks are used to indicate modules or more complex pieces of functionality.

The following block diagram gives high-level architecture for the TGV system that must

- Capture and store image data
- Sample several analog signals
 - Convert these to digital form
 - Store the results
- Receive GPS information
- Send and receive serial information and data
- 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 are interconnected via a system bus. The system bus components are not elaborated at the current stage of the design. Such elaboration can occur if necessary, as the design is refined.

Specifically observe that data and control flows not distinguished and that for the most part inputs appear on the left and outputs appear on the right. The one exception is the serial communications. That module supports bidirectional communication and is grouped with inputs subset.

Notice that we are using a visual grouping of like items. Specifically, the Serial communication and GPS blocks and the Analog measurement and image capture blocks comprise components that perform similar types of functions.

