Introduction to Embedded Systems
EHB326E
Lectures

Prof. Dr. Müştak E. Yalçın

Istanbul Technical University

mustak.yalcin@itu.edu.tr

Datapath

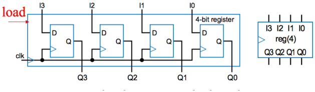
A datapath: storing and manipulating data
Datapath (Register-transfer level) components:

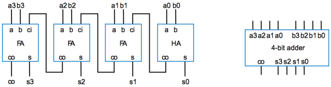
- Combinational Comp. (Multeplexor,Decoder, n-bit comporator, ALU)
- Sequential Comp.(n-bit Register, shift register, counter)

Course Prerequisites: EHB2015 Introduction to Logic Design (BLG 231E Digital Circuits) : Combinational Logic (Combinational logic design (Karnaugh map,...)), Sequential Logic (Sequential logic design)

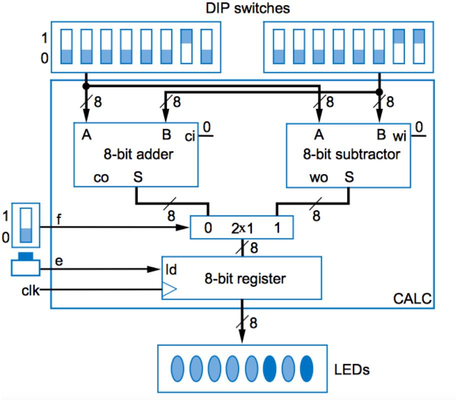
Datapath (Frank Vahid, Digital Design (Chapter 4))

Register & Adder

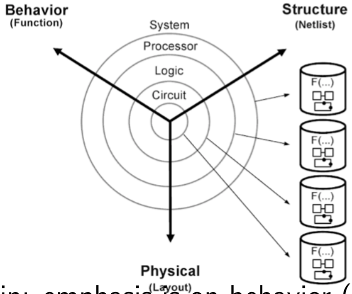




Adding/Subtracting Calculator



Y-chart (Embedded System Design Modeling, Synthesis and Verification, Gajski et al., Chapter 1)



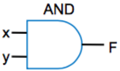
- Functional Domain: emphasis is on behavior (input - output functionality), without any reference to the particular way in which this behavior is implemented.
- Structural Domain: the specification is in terms of hierarchy of interconnected functional components.
- Physical Domain: the specification is in terms of physical placement in space and physical characteristics without any elements to functionality.

Circuit & Logic Level Abstraction

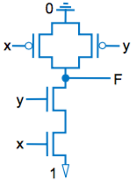
Behavior

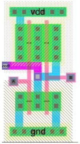
Structure

Physical



x	y	F
0	0	0
0	1	0
1	0	0
1	1	1

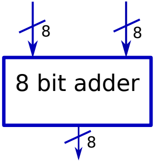


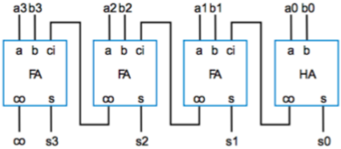


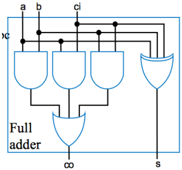
EHB 205E Introduction to Logic Design

EHB 322E Digital Electronic Circuits

EHB 413E VLSI Circuit Design I



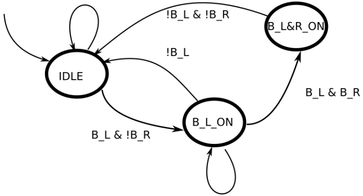




$S = A + B$, EHB 205E Introduction to Logic Design

Processor Level Abstraction

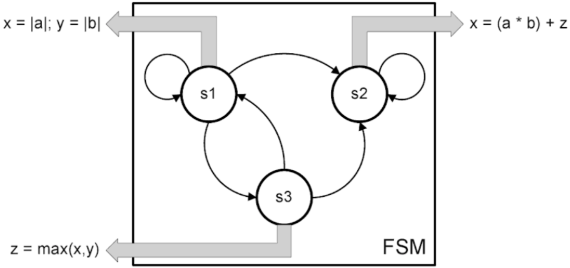
Computational components or Processing Elements (PEs) are defined and designed!



The functionality of a PE can be specified with an algorithm or with a flow chart. Finite State Machine (FSM) is a mathematical model of computation (for simple control functionality, (good for several hundred states,binary variable for I/O, EHB2015)

Finite State Machine with Datapath

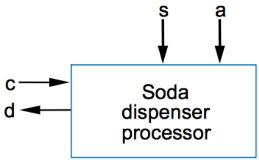
FSM model can be extended (Finite State Machine with Datapath (FSMD)) using standard integer or floating-point variables and computing their values in each state or during each transition by a set of arithmetic expressions or programming statements.

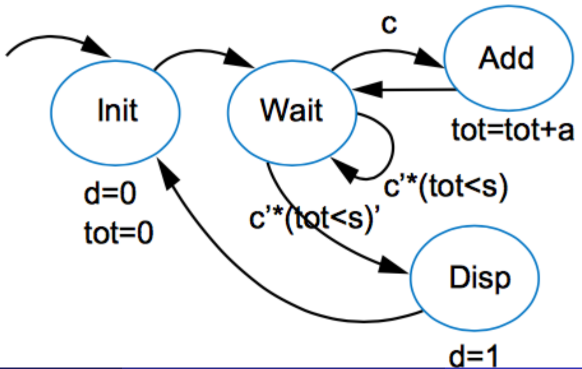


FSMs allow only binary I/O and don't allow local data storage (only state!).

Example 5.1 (Vahid, page 227) : Soda dispenser

- c: bit input, 1 when coin deposited
- a: 8-bit input having value of deposited coin
- s: 8-bit input having cost of a soda
- d: bit output, processor sets to 1 when total value of deposited coins equals or exceeds cost of a soda





Register Transfer Level (RTL) Design Method

RTL design is one of the methods for processor design!

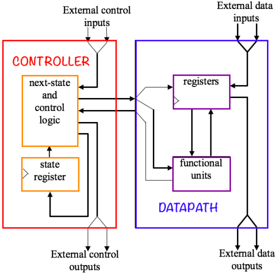
Designer specifies the register of a design, describes the possible transfers and operations perform on input,output or register data and defines the control that specifies when to transfer and operate on data !

Step$_1$: Describe the system's desired behaviour as a finite state machine with data.

Step$_2$: Create a datapath to carry data operations.

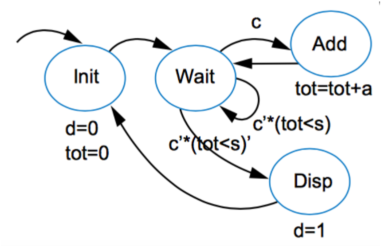
Step$_3$: Connect the datapath to a control unit.

Step$_4$: Drive the controller's FSM (replacing data operations with setting and reading of control signal to and from the datapath)



Digital Design, F. Vahid, Chapter 5

Step$_2$



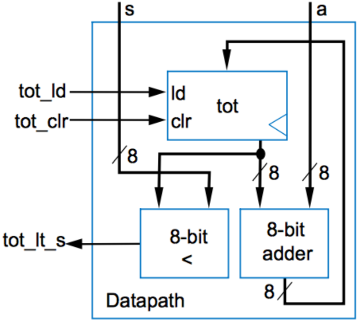
External Data Inputs: a, s

Operations : adder, comparator

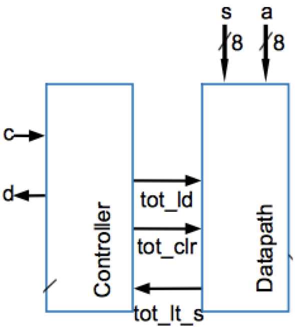
Registers: tot

Control Input: tot_load, tot_clear

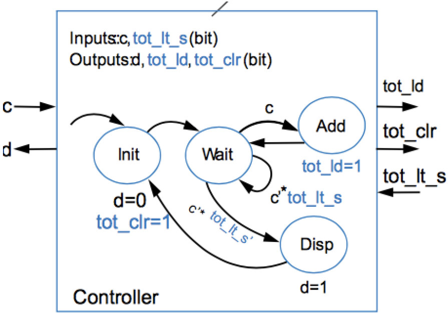
Control Output: tot_compare_result



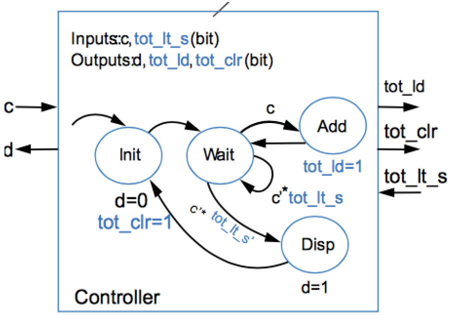
Step3

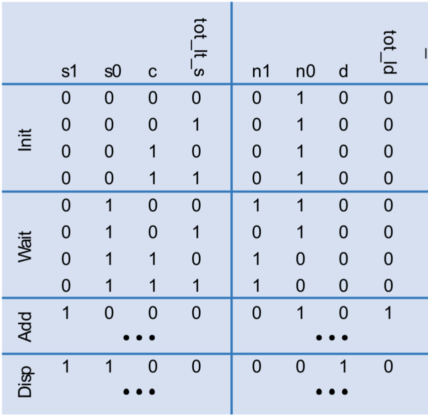

s
a
8
8
c
d
Controller
tot_ld
tot_clr
tot_lt_s
Datapath

Step4


Inputs:c, tot_lt_s (bit)
Outputs:d, tot_ld, tot_clr (bit)
c
d
tot_ld
tot_clr
tot_lt_s
Init
Wait
Add
Disp
c
tot_ld=1
c'*tot_lt_s
c'* tot_lt_s'
d=0
tot_clr=1
d=1
Controller

Step$_4$



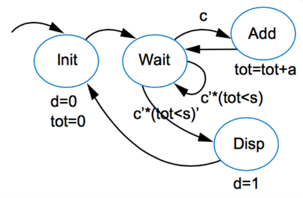


	s1	s0	c	tot_lt_s	n1	n0	d	tot_ld
Init	0	0	0	0	0	1	0	0
	0	0	0	1	0	1	0	0
	0	0	1	0	0	1	0	0
	0	0	1	1	0	1	0	0
Wait	0	1	0	0	1	1	0	0
	0	1	0	1	0	1	0	0
	0	1	1	0	1	0	0	0
	0	1	1	1	1	0	0	0
Add	1	0	0	0	0	1	0	1
		•••				•••		
Disp	1	1	0	0	0	0	1	0
		•••				•••		

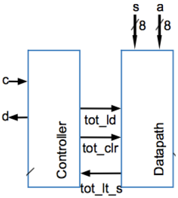
Implementation of controller's state table: Sequential logic design(EHB205 Int. Logic Design)

Processor Level Abstraction

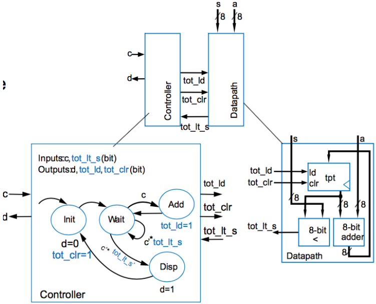
Behavior



Structure



Physical



More to read : P. Schaumont, A Practical Introduction to Hardware/Software Codesign (Chapter 4)