
CSE 20221: Logic Design

PS/2 Keyboard Protocol

Interface to the Basys Board

Basys Board

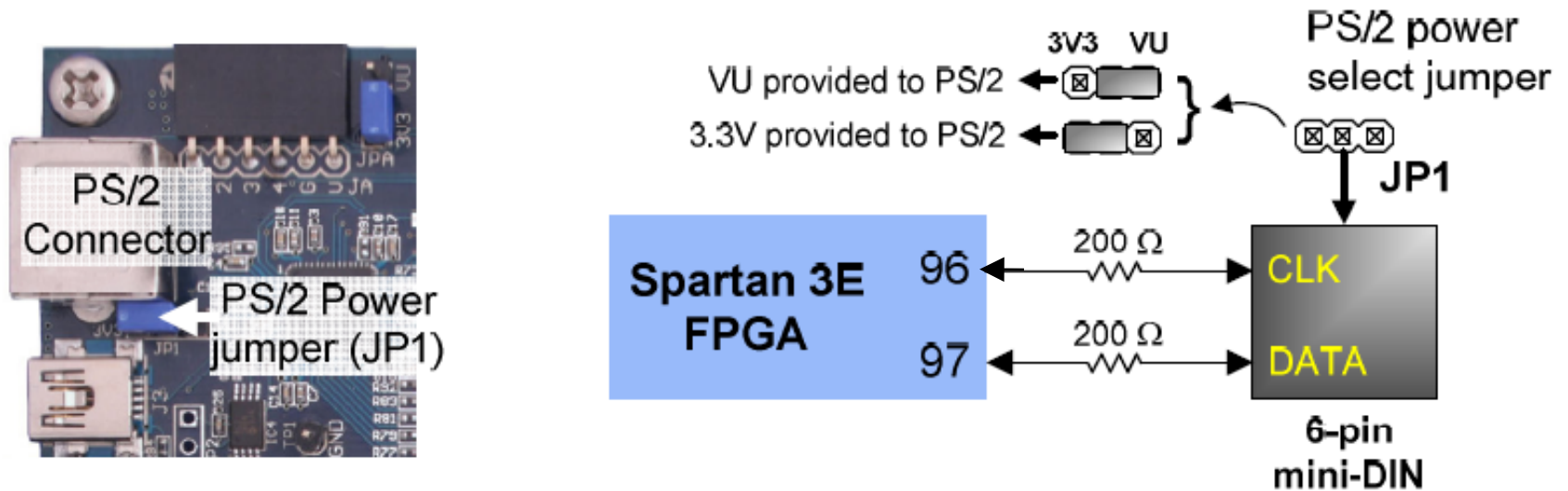
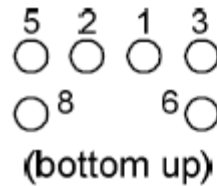
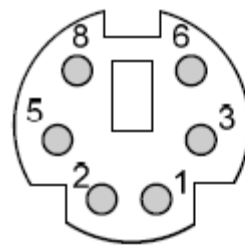


Figure 9. PS/2 connector and Basys PS/2 circuit

Basys Board Connector

- The Basys Board can interface to either a mouse or keyboard

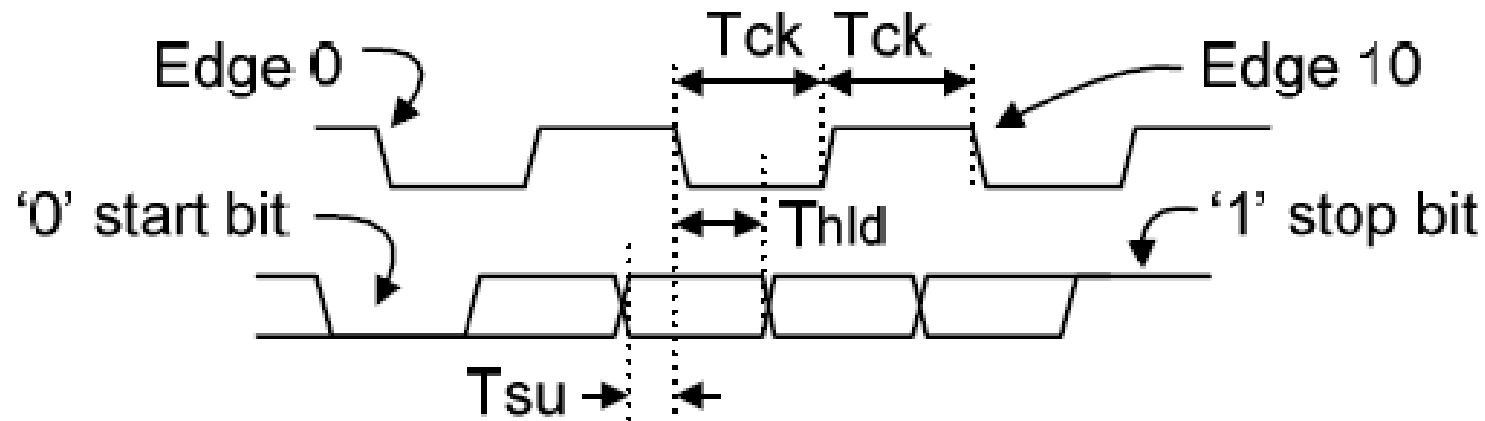


Pin1: Data
Pin2: Data
Pin3: GND
Pin5: Vdd
Pin6: Clock
Pin8: Clock

PS/2 I/O Signals

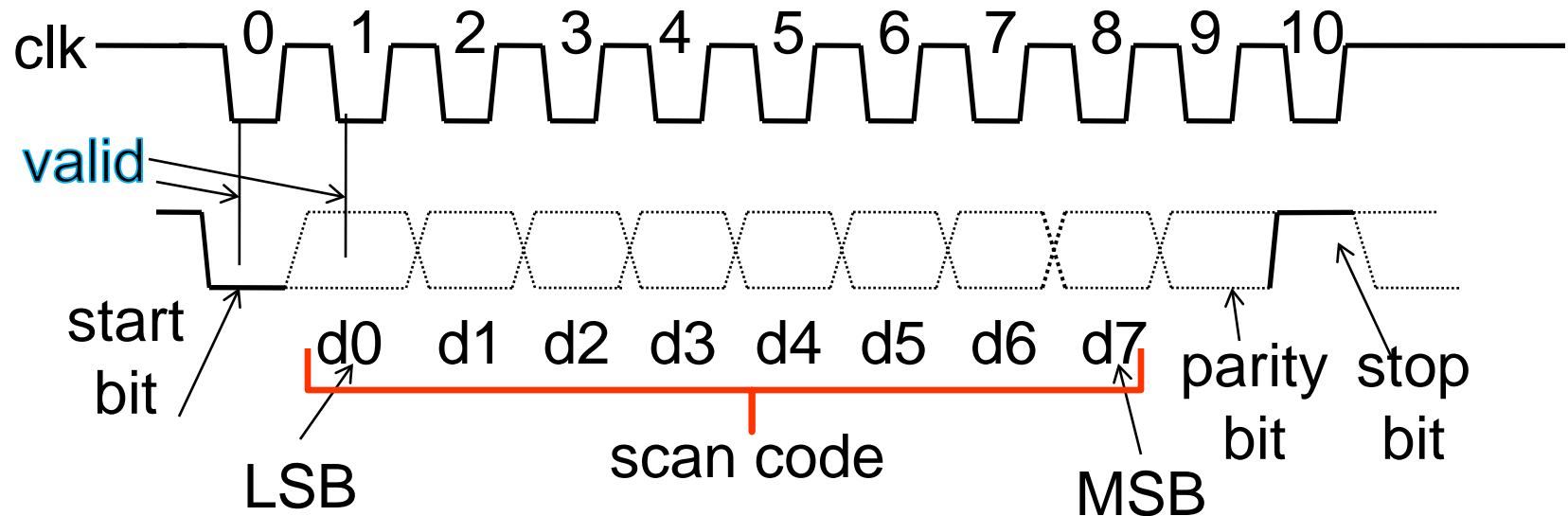
- clock: used for synchronization to indicate when data is valid.
 - Bidirectional, i.e., from keyboard to host or vice versa
- Data:
 - From keyboard: 8-bit (byte) scan code to indicate which key was pressed.
 - From host: keyboard configuration commands
- In this course we will only consider the direction from keyboard to host.

PS/2 Timing Specifications



Symbol	Parameter	Min	Max
T_{CK}	Clock time	30us	50us
T_{SU}	Data-to-clock setup time	5us	25us
T_{HLD}	Clock-to-data hold time	5us	25us

Signal Protocol



Notes

data is valid on negative edge of clock

clock frequency: 20 – 30 KHz

parity is odd

Scan Codes in Hex

ESC 76	F1 05	F2 06	F3 04	F4 0C	F5 03	F6 0B	F7 83	F8 0A	F9 01	F10 09	F11 78	F12 07	↑ E0 75	
~ 0E	1! 16	2@ 1E	3# 26	4\$ 25	5% 2E	6^ 36	7& 3D	8* 3E	9(46	0) 45	-_ 4E	=+ 55	BackSpace ← 66	→ E0 74
TAB 0D	Q 15	W 1D	E 24	R 2D	T 2C	Y 35	U 3C	I 43	O 44	P 4D	[{ 54]} 5B	\ 5D	← E0 6B
Caps Lock 58	A 1C	S 1B	D 23	F 2B	G 34	H 33	J 3B	K 42	L 4B	:: 4C	'" 52	Enter ↵ 5A	↓ E0 72	
Shift 12	Z 1Z	X 22	C 21	V 2A	B 32	N 31	M 3A	,< 41	>. 49	/? 4A	⬆ Shift 59			
Ctrl 14	Alt 11	Space 29						Alt E0 11	Ctrl E0 14					

Sequence of Events

- key pressed: 11-bit code is sent
 - start bit, 8-bit scan code, odd parity bit, stop bit
- key released: two 11-bit code is sent
 - first scan code is FO
 - second scan code is the released key code
- key held down: 11-bit code sent every 100 ms.
 - scan code is pressed key code

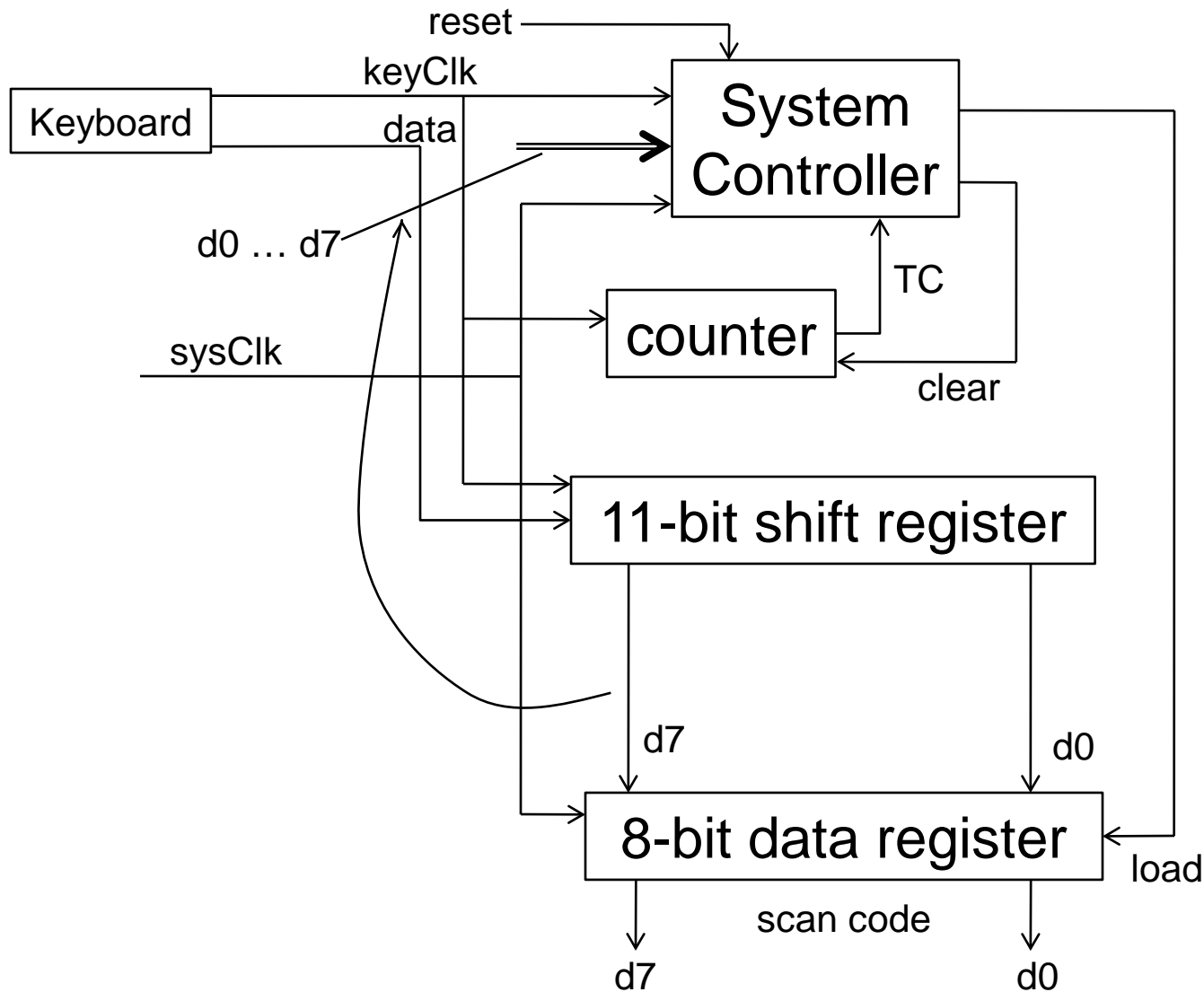
Additional Events

- Shift, Caps Lock, Ctrl: scan code is sent like any other key, host must keep track of the mode.
- Extended keys: E0 is sent, followed by the scan code. When an extended key is released, E0 F0 is sent.

Design Process Steps

1. Identify inputs and outputs and describe their function.
2. Write a functional (behavioral) description that incorporates the I/O.
3. Sketch a block diagram of the system. This should include the FSM controller and all interacting signals, modules and hardware devices – this is the architecture.
4. Identify all the unique states. Refer to the functional description to help identify the different states
5. Draw the state diagram(s). There may be multiple FSMs.
6. Write the Verilog modules for each block identified in step 3.
7. Verify consistency between the Verilog description and the documentation generated in steps 1 – 5.
8. Write test benches for each module.
9. Simulate and verify correct functionality.
10. Make necessary design changes.
11. Integrate the module in a top-level module or schematic and repeat steps 8 – 10.
12. Implement the prototype and verify and validate the design.

Block Diagram



System Controller States

- Initial
 - reset counter, wait for start bit
- Start
 - enable counter, shift in data, wait for stop bit
- checkCode
 - evaluate scan code for FO
 - must eventually consider EO and special keys like shift
- SaveCode
 - load scan code into register
- keyUp
 - ignore FO on key released
- keyPressed
 - indicate scan code is ready for key that was pressed
- Ignore
 - ignore scan code that occurs after FO on key released

System Controller State Diagram

