Rock Paper Scissors Digital Logic Documentation

PapilioDuo and Logic StartShield

Inputs:

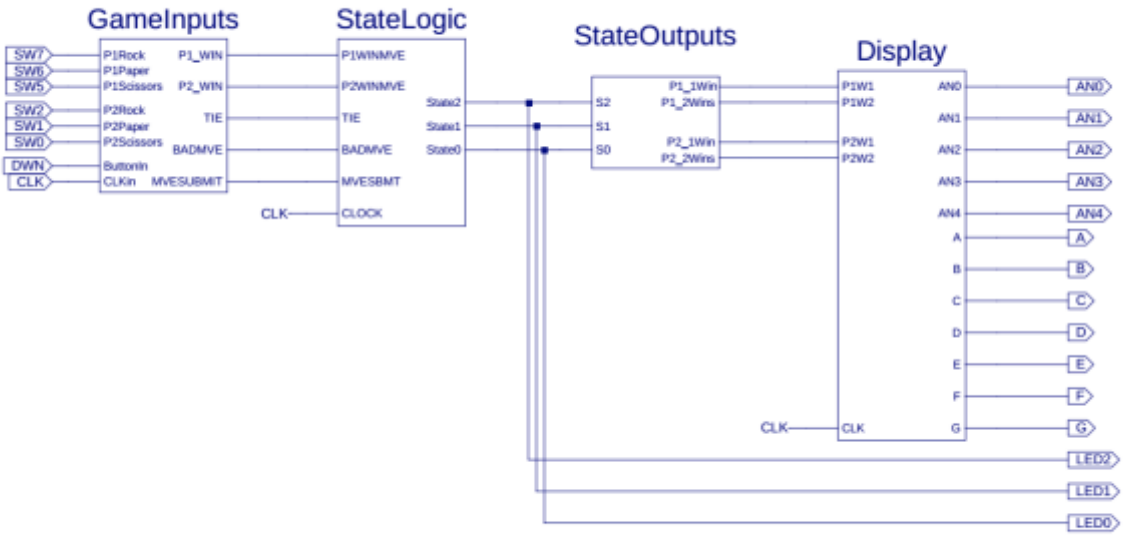
- Player One
  - Rock – SW7
  - Paper – SW6
  - Scissors – SW5
- Player Two
  - Rock – SW2
  - Paper – SW1
  - Scissors – SW0
- Submit – Down Button

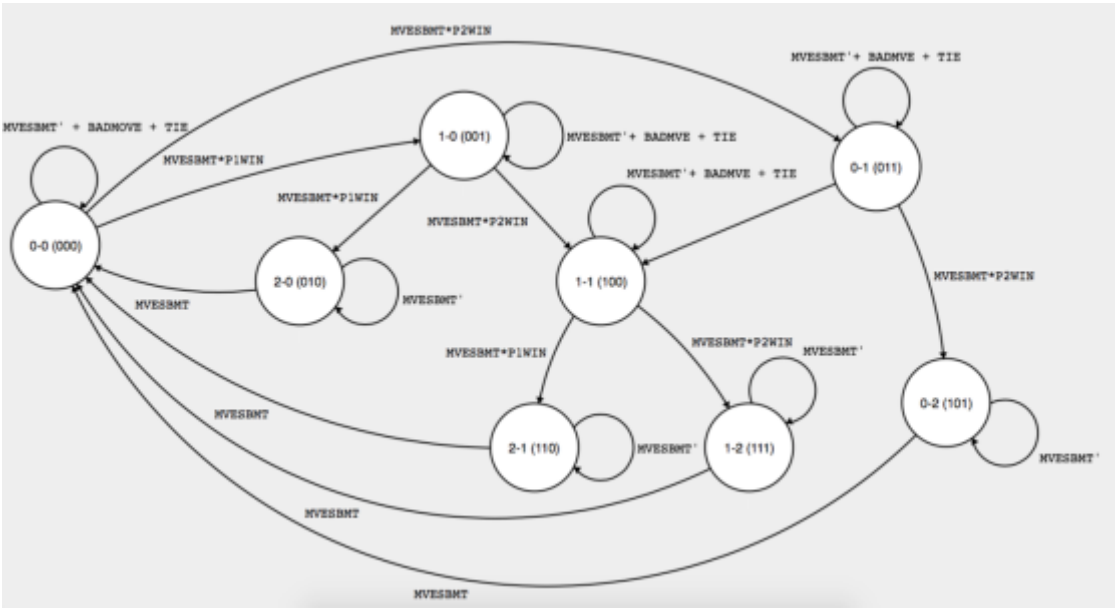
Outputs:

- Four Digit Seven Segment Display
  - AN0-AN4 – Anodes for the displays
  - A-G – Outputs for each segment in the display
- LEDs
  - LED 2 – State 2
  - LED 1 – State 1
  - Led 0 – State 0

Components:

- GameInputs
  - ButtonPulser
- StateLogic
  - N2Logic
  - N1Logic
  - N0Logic
  - REG3
- StateOutputs
- Display





GameInputs

Inputs:

- Player One
  - Rock – SW7
  - Paper – SW6
  - Scissors – SW5
- Player Two
  - Rock – SW2
  - Paper – SW1
  - Scissors – SW0
- Submit
  - Down Button

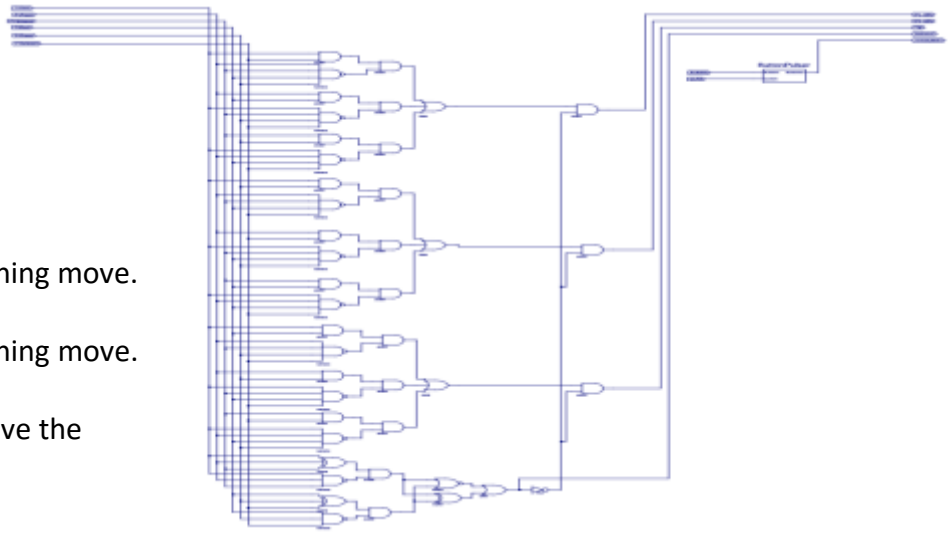
Outputs:

- P1_Win
  - High when P1 has a winning move.
- P2_Win
  - High when P2 has a winning move.
- TIE
  - High when P1 and P2 have the same move chosen.
- BadMove
  - High when an illegal mood is present.
- MVESUBMIT
  - High for only One (1) Clock Cycle after the Submit (Down Button) was pushed

Components:

- ButtonPulser

Purpose: Takes in P1 and P2's move choices, determines the result and outputs it.



GameInputs									
Inputs						Outputs			
P1Rock	P1Paper	P1Scissor	P2Rock	P2Paper	P2Scissor	P1Win	P2Win	Tie	Badmove
0	0	0	0	0	0	0	0	0	1
1	0	0	0	0	1	1	0	0	0
0	1	0	1	1	0	1	0	0	0
0	0	1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	1	0	0
0	1	0	0	0	1	0	1	0	0
0	0	1	0	0	0	0	1	0	0
1	0	0	1	0	0	0	0	1	0
0	1	0	0	1	0	0	0	1	0
0	0	1	0	0	1	0	0	1	0
Note: All other combinations cause "Badmove" to go high and cause "P1Win", "P2Win" and "Tie" to go low.									

ButtonPulser

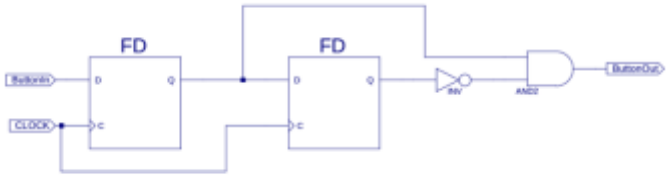
Purpose: Takes a button as an input and only outputs high for one clock cycle. This component will not output high until the button is released for two clock cycles and then pressed again.

Inputs:

- ButtonIn
- Clock

Outputs:

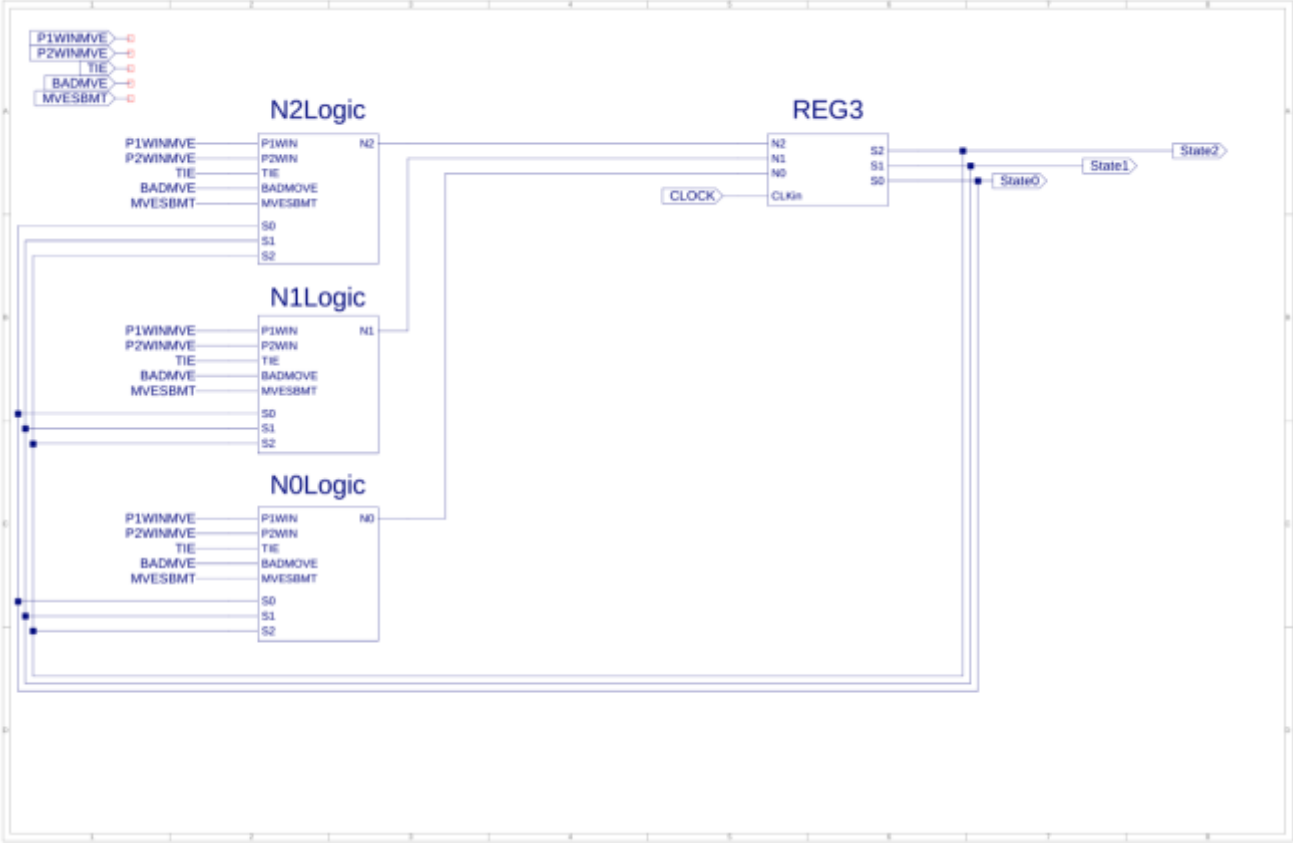
- ButtonOut
  - Goes high for only one clock cycle after the button is pushed



Button Pulser					
Inputs		Outputs			
ButtonIn	Clock	FD0	FD1	FD1'	ButtonOut
0	0	0	0	1	0
0	1	0	0	1	0
1	0	0	0	1	0
1	1	1	0	1	0
1	0	1	0	1	0
1	1	1	1	0	1
1	0	1	1	0	0
0	1	0	1	0	1
0	0	0	1	0	0
0	1	0	0	1	0

StateLogic

Purpose: Takes GameInputs' Result (P1 Winning Move, P2 Winning Move, Tie or Badmove and Move Submit) and the Current State (S2:0) as inputs and determines what the Next State would be. Once N2:0 is determined it is stored in a 3-Bit Register where it will become the Current State on the next rising edge of the clock.



Inputs:

- P1WINMVE
- P2WINMVE
- TIE
- BADMVE
- MOVESBMT
- Current State
  - S2:0

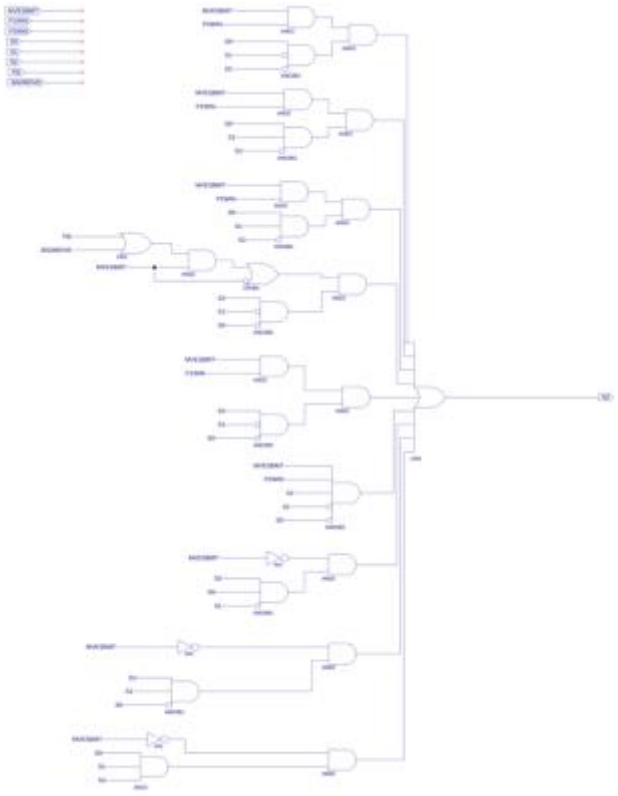
Outputs:

- Current State
  - State 2
  - State 1
  - State 0

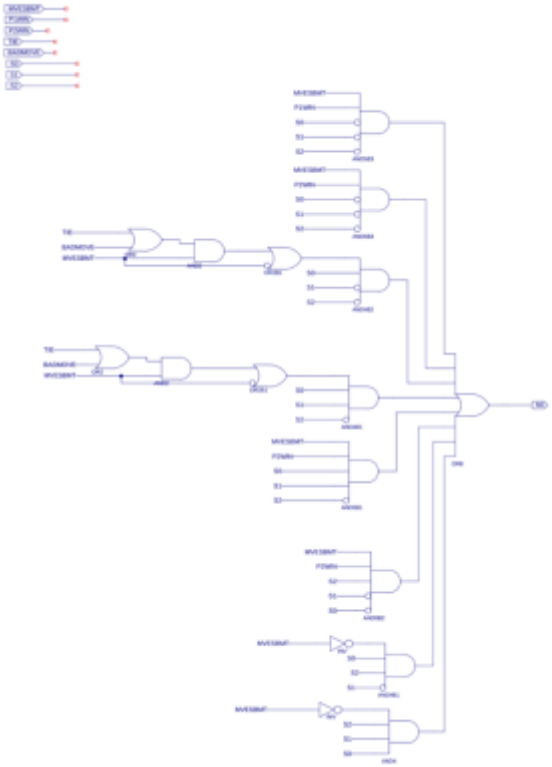
Components:

- N2Logic
- N1Logic
- N0Logic
- REG3

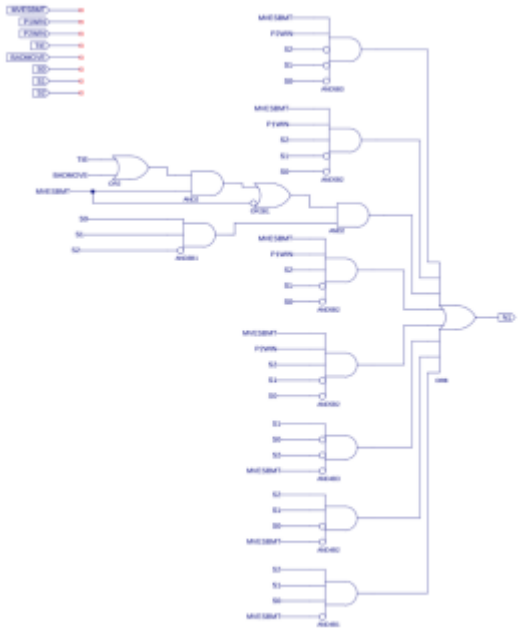
N2Logic



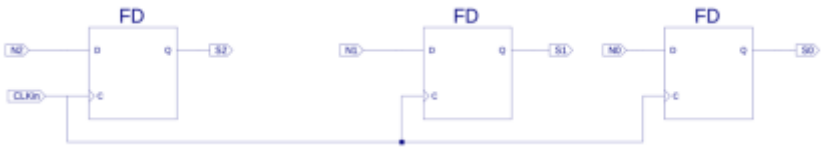
N0Logic



N1Logic



REG3



StateLogic

Inputs					Current State				Next State			
P1WINMVE	P2WINMVE	TIE	BADMVE	MVESBMT	S2	S1	S0	Score	N2	N1	N0	Next Score
0	0				0	0	0	0--0	0	0	0	0--0
1	0	0	0	1	0	0	0		0	0	1	1--0
0	1	0	0	1	0	0	0		0	1	1	0--1
0	0				0	0	1	1--0	0	0	1	1--1
1	0	0	0	1	0	0	1		0	1	0	2--0
0	1	0	0	1	0	0	1		1	0	0	1--1
0	0				0	1	1	0--1	0	1	1	0--1
1	0	0	0	1	0	1	1		1	0	0	1-1
0	1	0	0	1	0	1	1		1	0	1	0--2
0	0				1	0	0	1--1	1	0	0	1--1
1	0	0	0	1	1	0	0		1	1	0	2--1
0	1	0	0	1	1	0	0		1	1	1	1--2
				0	0	1	0	2--0	0	1	0	2--0
				1	0	1	0		0	0	0	0--0
				0	1	1	0	2--1	1	1	0	2--1
				1	1	1	0		0	0	0	0--0
				0	1	0	1	0--2	1	0	1	0--2
				1	1	0	1		0	0	0	0--0
				0	1	1	1	1--2	1	1	1	1--2
				1	1	1	1		0	0	0	0--0

StateOutputs

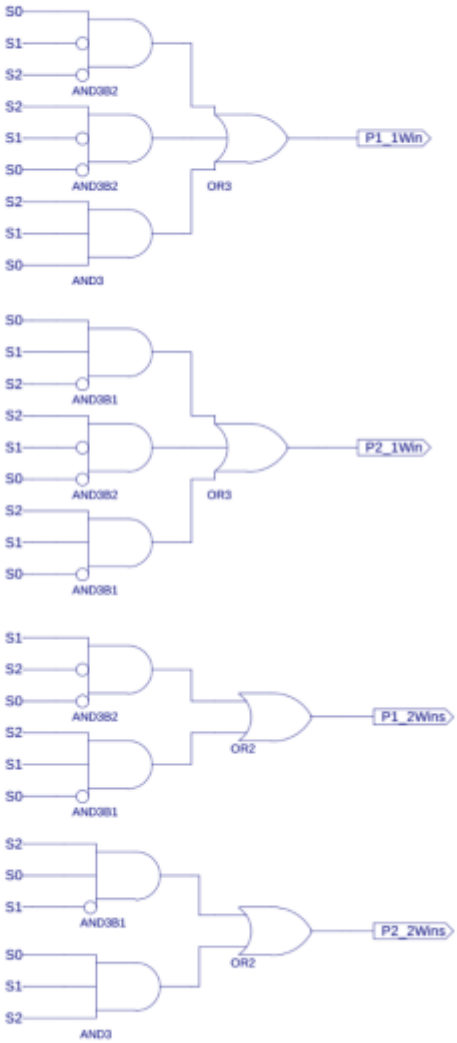
Purpose: Takes the Current State as an input, then determines how many wins each player has and outputs it.

Inputs:

- Current State
  - S2:0

Outputs:

- P1_1Win – High when P1 has ONE Win
- P2_1Win – High when P2 has ONE Win
- P1_2Wins – High when P1 has TWO Wins
- P2_2Wins – High when P2 has TWO Wins



StateOutputs							
Current State				Outputs			
S2	S1	S0	Score	P1_1Win	P2_1Win	P1_2Wins	P2_2Wins
0	0	0	0--0	0	0	0	0
0	0	0					
0	0	0					
0	0	1	1--0	1	0	0	0
0	0	1					
0	0	1					
0	1	1	0--1	0	1	0	0
0	1	1					
0	1	1					
1	0	0	1--1	1	0	1	0
1	0	0					
1	0	0					
0	1	0	2--0	0	0	1	0
0	1	0					
1	1	0	2--1	0	1	1	0
1	1	0					
1	0	1	0--2	0	0	0	1
1	0	1					
1	1	1	1--2	1	0	0	1
1	1	1					

Display

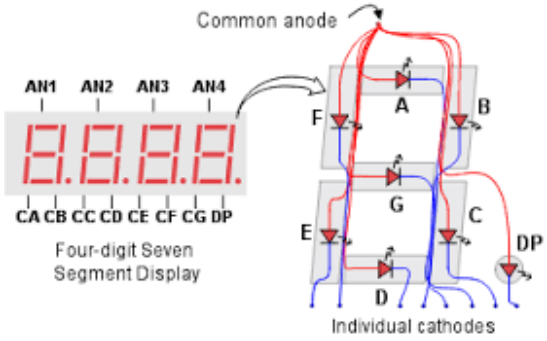
Purpose: Takes the number of wins each player has as an input and outputs the correct data to a four-digit seven-segment display.

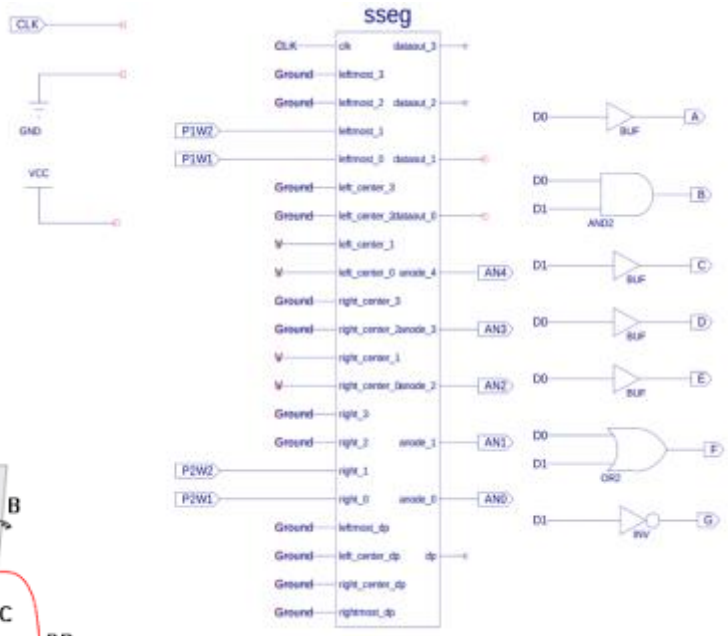
Inputs:

- P1W1
- P2W1
- P1W2
- P2W2

Outputs:

- AN0-AN4 – Anodes for the displays
- A-G – Outputs for each segment in the display





Display																
Inputs			Outputs													
# of Wins	D1	D0	Digit Shown	A	B	C	D	E	F	G	AN3 (L/P1)	AN2 (LC/Dash)	AN1 (RC/Dash)	AN0 (R/P2)	Displayed On	Displayed?
0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	Left	0_ _ _
											0	0	0	1	Right	_ _ _ 0
1	0	1	1	1	0	0	1	1	1	1	1	0	0	0	Left	1 _ _ _
											0	0	0	1	Right	_ _ _1
2	1	0	2	0	0	1	0	0	1	0	1	0	0	0	Left	2_ _ _
											0	0	0	1	Right	_ _ _2
	1	1	-	1	1	1	1	1	1	0	0	1	0	0	Left Center	_ - _ _
											0	0	1	0	Right Center	_ _ - _
Note: D1 & D0 are tied high for AN1 and AN2, therefore a Dash is always produced on the Left and Right Center Displays.																