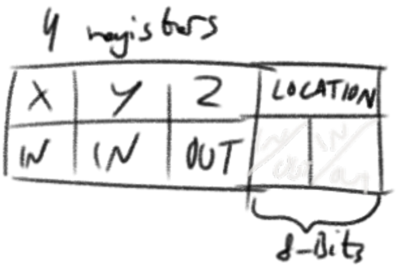
Pixel Brush Processing Unit

Monday, 3 May 2021 09:36

Mark 1

$2^8 = 256$

0-f



Hex	Binary	OP code	Inputs	Name	Description
0	0000	NOP	0	No operation	No operation is executed
1	0001	ADD	0	Add	X and Y are added, the result is written to Z
2	0010	SUB	0	Subtract	X and Y are subtracted, the result is written to Z
3	0011	WT1	0-f	Write to Loc1	The first 4 Bits of the Location Byte are written to
4	0100	WT2	0-f	Write to Loc2	The last 4 Bits of the Location Byte are written to
5	0101	WTX	0-f	Write to X	The X Byte is written to
6	0110	WTY	0-f	Write to Y	The Y Byte is written to
7	0111	WTZ	0-f	Write to Z	The Z Byte is written to
8	1000	PTZ	0	Write Z to Loc	Put Z variable in location specified by X and Y

Example Code

Pseudo	Hex	Meaning
NOP 0000	0 0	No operation
WTX 0101	5 5	Write 5 to X
WTY 1010	6 A	Write A to Y
ADD 0000	1 0	Add X and Y
WT1 0000	3 1	Write 0 to Loc1
WT2 0001	4 0	Write 1 to Loc2

Result:
X = 5 (5)
Y = 10 (A)
Z = 15 (F)
Location = 1 (1)

Mark 2

Hex	Binary	OP code	Inputs	Name	Description
0	*0000*	NOP	*0*	No operation	No operation is executed
1	*0001*	ADD	*0*	Add	X and Y are added, the result is written to Z
2	*0010*	SUB	*0*	Subtract	X and Y are subtracted, the result is written to Z
3	*0011*	WT1	*0-f*	Write to Loc1	The first 4 Bits of the Location Byte are written to
4	*0100*	WT2	*0-f*	Write to Loc2	The last 4 Bits of the Location Byte are written to
5	*0101*	WTX	*0-f*	Write to X	The X Byte is written to
6	*0110*	WTY	*0-f*	Write to Y	The Y Byte is written to
7	*0111*	WTZ	*0-f*	Write to Z	The Z Byte is written to
8	*1000*	ZTR	*0*	Write Z to RAM	Put Z variable in location specified by X and Y
9	*1001*	RTZ	*0*	Read from RAM	Writes the Value of the current RAM location to Z
a	*1010*	PC1	*0-f*	Program Counter To (1)	Sets Program Counter to desired value (part 1)
b	*1011*	PC2	*0-f*	Program Counter To (2)	Sets Program Counter to desired value (part 2)
c	*1100*	JMP	*0*	Jump to PC	Uses the values set by PC1 and PC2 to set the Program counter to the desired value, only activates if Z = 1

Example Code

Binary	Hex	Pseudo	Meaning
0000 0000	0 0	NOP	No operation
0011 0000	3 0	WT1 0	Set Loc1 to 0
0100 0000	4 0	WT2 0	Set Loc2 to 0
0111 1001	7 9	WTZ 9	Write 9 to Z
1000 0000	8 0	ZTR	Write Z to RAM
0100 0010	4 2	WT2 2	Set Loc2 to 2
0111 1001	7 9	WTZ 9	Write 9 to Z
1000 0000	8 0	ZTR	Write Z to RAM
0100 0011	4 3	WT2 3	Set Loc2 to 3
0111 0110	7 6	WTZ 6	Write 6 to Z
1000 0000	8 0	ZTR	Write Z to RAM

Assuming 0-3 are lines 1-4 on the LED Matrix, this should draw a smiley.

Bugs:

- JMP must jump to odd address, or else op code/ number reading will be messed up

Not implemented:

- JMP only activates when Z = 1

Mark 3

Hex	Binary	OP code	Inputs	Name	Description
0	*0000*	NOP	*0*	No operation	No operation is executed
1	*0001*	ADD	*0*	Add	X and Y are added, the result is written to Z
2	*0010*	SUB	*0*	Subtract	X and Y are subtracted, the result is written to Z
3	*0011*	WT1	*0-f*	Write to Loc1	The first 4 Bits of the Location Byte are written to
4	*0100*	WT2	*0-f*	Write to Loc2	The last 4 Bits of the Location Byte are written to
5	*0101*	WTX	*0-f*	Write to X	The X Byte is written to
6	*0110*	WTY	*0-f*	Write to Y	The Y Byte is written to
7	*0111*	WTZ	*0-f*	Write to Z	The Z Byte is written to
8	*1000*	ZTR	*0*	Write Z to RAM	Put Z variable in location specified by X and Y
9	*1001*	RTZ	*0*	Read from RAM	Writes the Value of the current RAM location to Z
a	*1010*	PC1	*0-f*	Program Counter To (1)	Sets Program Counter to desired value (part 1)

Example Code

Binary	Hex	Pseudo	Meaning
0000 0000	0 0	NOP	No operation
		WTX	
		USC	

b	*1011*	PC2	*0-f*	Program Counter To (2)	Sets Program Counter to desired value (part 2)
c	*1100*	JMP	*0*	Jump to PC	Uses the values set by PC1 and PC2 to set the Program counter to the desired value, only activates if Z = 1
d	*1101*	RTX	*0*	RAM to X	Loads current RAM location to X
e	*1110*	RTY	*0*	RAM to Y	Loads current RAM location to Y
f	*1111*	USC	*0*	Use Carry	Toggles carry bit

Bugs:

- PPU writing changed to accommodate 8x8 grid, can cause issues in legacy programs
- PPU Memory is difficult to write to
- Carry Bit gets overwritten by new Calculation

Not implemented:

- IO