

## **32-point FFT (The Cooley-Tukey algorithm)**

Design:

The main operation of the FFT is

$Out1 = I0 + constant * I1$   
 $Out2 = I0 - constant * I1$

Expanding the Equations:

$Out1_{REAL} = ((I0_{real} + constant_{real} * I2_{REAL})_{FirstClk} - CONSTANT_{IMAG} * I2_{IMAG})_{SecondClk}$   
 $Out2_{REAL} = ((I0_{real} - constant_{real} * I2_{REAL})_{FirstClk} + CONSTANT_{IMAG} * I2_{IMAG})_{SecondClk}$   
 $Out1_{IMAG} = ((I0_{imag} + constant_{imag} * I2_{REAL})_{ThirdClk} + CONSTANT_{Real} * I2_{IMAG})_{FourthClk}$   
 $Out2_{IMAG} = ((I0_{imag} - constant_{imag} * I2_{REAL})_{ThirdClk} - CONSTANT_{Real} * I2_{IMAG})_{FourthClk}$

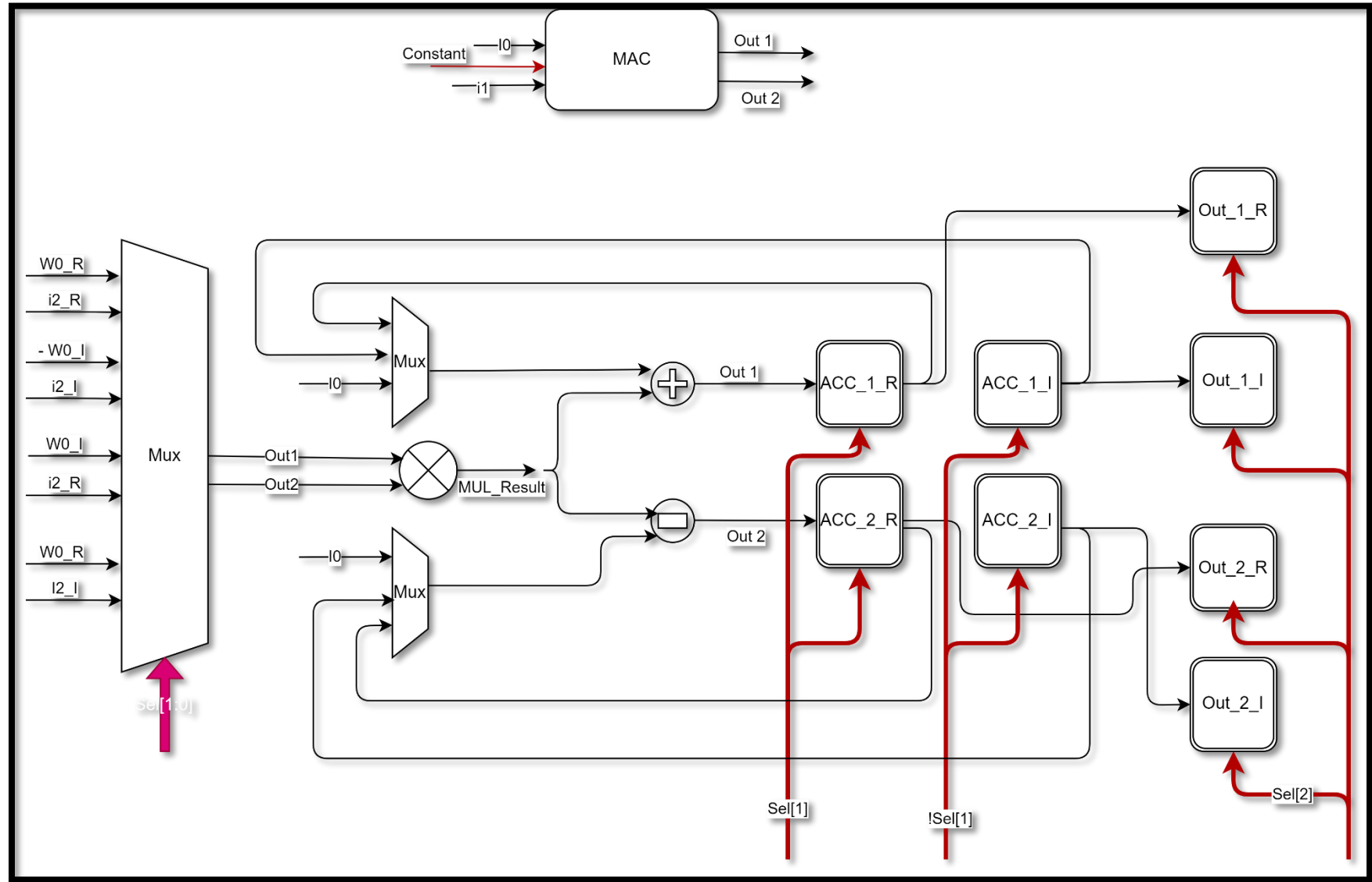
As shown, there is common operands between Out1Real & Out2Real same for Imaginary Part so in first 2 clks Real part is calculated (in first Clk the first part is calculated then in second clk the second part is accumulated on the Previous Result). Using Sel signals to select which inputs and which registers to write.

Same Idea applies for the Imaginary Part

In the 5<sup>th</sup> clk we store the 4 Results of the output and become the input of the second Stage

Here is the design showing all control signals (Red Lines) and the data path

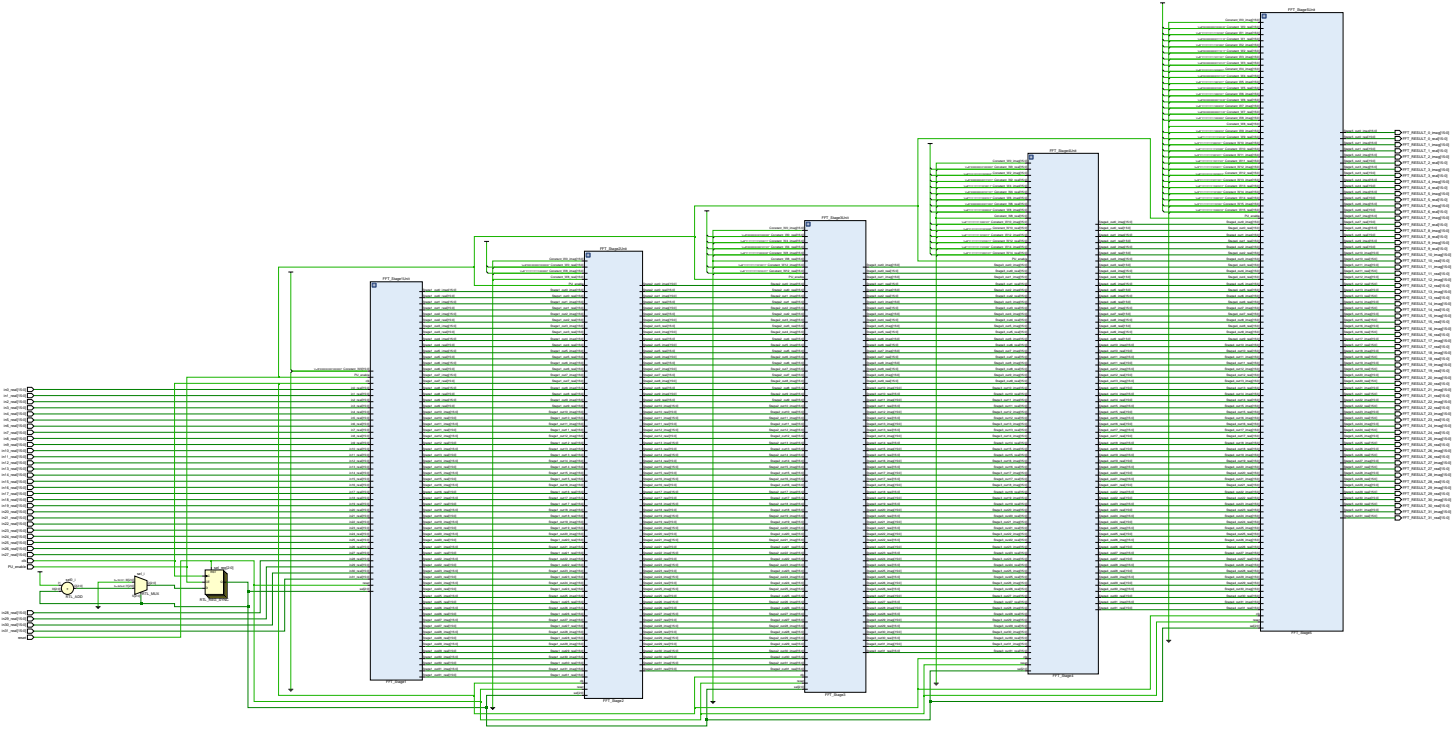
Mac:



Equations of stages:

Index	inputs	Stage1	Stage 2	Stage 3	Stage 4	Stage 5 (Result)
0	I0	S01= I0 + W0 * I16	S02= S01 + W0 * S21	S03=S02 + W0*S42	S04=S03+W0*S83	S05=S04+W0*S164
1	I16	S11= I0 - W0 * I16	S12= S11 + W8 * S31	S13=S12 + W4*S52	S14=S13+W2*S93	S15=S14+W1*S174
2	I8	S21= I8 + W0 * I24	S22= S01 - W0 * S21	S23=S22 + W8*S62	S24=S23+W4*S103	S25=S24+W2*S184
3	I24	S31= I8 - W0 * I24	S32= S11 - W8 * S31	S33=S32 + W12*S72	S34=S33+W6*S113	S35=S34+W3*S194
4	I4	S41= I4 + W0 * I20	S42= S41 + W0 * S61	S43=S02 - W0*S42	S44=S43+W8*S123	S45=S44+W4*S204
5	I20	S51= I4 - W0 * I20	S52= S51 + W8 * S71	S53=S12 - W4*S52	S54=S53+W10*S133	S55=S54+W5*S214
6	I12	S61= I12 + W0 * I28	S62= S41 - W0 * S61	S63=S22 - W8*S62	S64=S63+W12*S143	S65=S64+W6*S224
7	I28	S71= I12 - W0 * I28	S72= S51 - W8 * S71	S73=S32 - W12*S72	S74=S73+W14*S153	S75=S74+W7*S234
8	I2	S81= I2 + W0 * I18	S82= S81 + W0 * S101	S83=S82+W0*S122	S84=S03-W0*S83	S85=S84+W8*S244
9	I18	S91= I2 - W0 * I18	S92= S91 + W8 * S111	S93=S92+W4*S132	S94=S13 - W2*S93	S95=S94+W9*S254
10	I10	S101= I10 + W0 * I26	S102= S81 - W0 * S101	S103=S102+W8*S142	S104=S23 - W4*S103	S105=S104+W10*S264
11	I26	S111= I10 - W0 * I26	S112= S91 - W8 * S111	S113=S112+W12*S152	S114=S33 - W6*S113	S115=S114+W11*S274
12	I6	S121= I6 + W0 * I22	S122= S121 +W0*S141	S123= S82- W0*S122	S124=S43 - W8*S123	S125=S124+W12*S284
13	I22	S131= I6 - W0 * I22	S132=S131 + W8*S151	S133= S92- W4*S132	S134=S53 - W10*S133	S135=S134+W13*S294
14	I14	S141= I14 + W0 * I30	S142= S121 -W0*S141	S143= S102- W8*S142	S144=S63 - W12*S143	S145=S144+W14*S304
15	I30	S151= I14 - W0 * I30	S152=S131 - W8*S151	S153= S112- W12*S152	S154=S73 - W14*S153	S155=S154+W15*S314
16	I1	S161= I1 + W0 * I17	S162=S161 + W0*S181	S163 = S162 + W0* S202	S164=S163 +W0*S243	S165=S04 - W0*S164
17	I17	S171= I1 - W0 * I17	S172=S171 + W8*S191	S173 = S172+ W4*S212	S174=S173+W2*S253	S175=S14 - W1*S174
18	I9	S181= I9 + W0 * I25	S182=S161 - W0*S181	S183= S182+ W8*S222	S184=S183+W4*S263	S185=S24 - W2*S184
19	I25	S191= I9 - W0 * I25	S192=S171 - W8*S191	S193= S192+ W12*S232	S194=S193+W6*S273	S195=S34 - W3*S194
20	I5	S201= I5 + W0 * I21	S202=S201 + W0*S221	S203 = S162 - W0* S202	S204=S203+W8*S283	S205=S44 - W4*S204
21	I21	S211= I5 - W0 * I21	S212=S211 + W8*S231	S213 = S172 - W4*S212	S214=S213+W10*S293	S215=S54 - W5*S214
22	I13	S221= I13 + W0 * I29	S222=S201 - W0*S221	S223= S182 - W8*S222	S224=S223+W12*S303	S225=S64 - W6*S224
23	I29	S231= I13 - W0 * I29	S232=S211 - W8*S231	S233= S192 - W12*S232	S234=S233+W14*S313	S235=S74 - W7*S234
24	I3	S241= I3 + W0 * I19	S242=S241 + W0*S261	S243=S242 + W0*S282	S244=S163 -W0*S243	S245=S84 - W8*S244
25	I19	S251= I3 - W0 * I19	S252=S251 + W8*S271	S253=S252 + W4*S292	S254=S173-W2*S253	S255=S94 - W9*S254
26	I11	S261= I11 + W0 * I27	S262=S241 - W0*S261	S263=S262 + W8*S302	S264=S183-W4*S263	S265=S104 - W10*S264
27	I27	S271= I11 - W0 * I27	S272=S251 - W8*S271	S273=S272 + W12*S312	S274=S193-W6*S273	S275=S114 - W11*S274
28	I7	S281= I7 + W0 * I23	S282=S281 + W0 *S301	S283=S242 - W0*S282	S284=S203-W8*S283	S285=S124 - W12*S284
29	I23	S291= I7 - W0 * I23	S292= S291+W8*S311	S293=S252 - W4*S292	S294=S213-W10*S293	S295=S134 - W13*S294
30	I15	S301= I15 + W0 * I31	S302=S281 - W0 *S301	S303=S262 - W8*S302	S304=S223-W12*S303	S305=S144 - W14*S304
31	I31	S311= I15 - W0 * I31	S312= S291-W8*S311	S313=S272 - W12*S312	S314=S233-W14*S313	S315=S154 - W15*S314

# Schematic

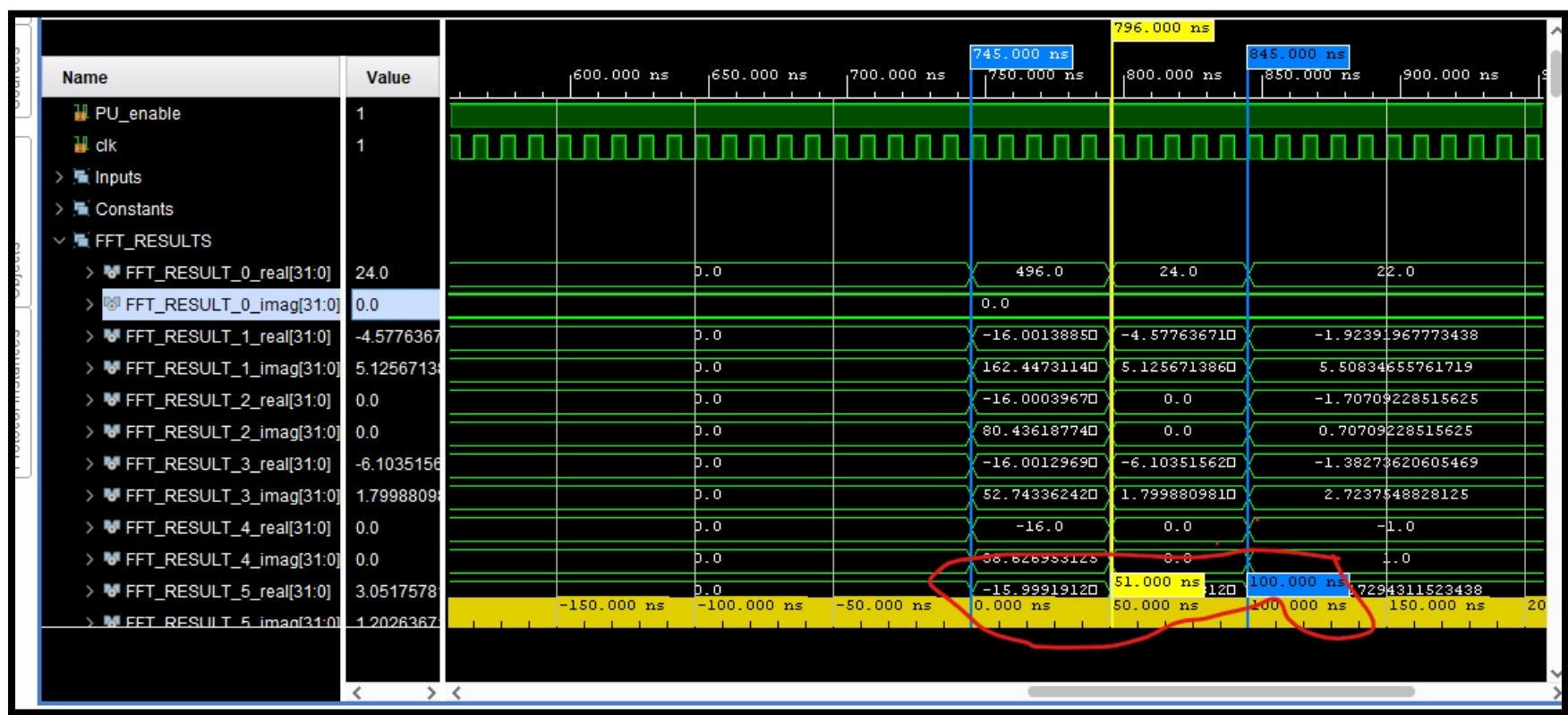






Pipeline Result:

FFT block Output is with 20 MHz we inserted 3 different input with 20 MHZ and here is the output showing the time between each output is 50 ns



Timing Results:

Setup	Hold	Pulse Width
Worst Negative Slack (WNS): 3.195 ns	Worst Hold Slack (WHS): 0.079 ns	Worst Pulse Width Slack (WPWS): 4.650 ns
Total Negative Slack (TNS): 0.000 ns	Total Hold Slack (THS): 0.000 ns	Total Pulse Width Negative Slack (TPWS): 0.000 ns
Number of Failing Endpoints: 0	Number of Failing Endpoints: 0	Number of Failing Endpoints: 0
Total Number of Endpoints: 20483	Total Number of Endpoints: 20483	Total Number of Endpoints: 10244
All user specified timing constraints are met.		

General Information	Name	Waveform	Period (ns)	Frequency (MHz)
Timer Settings	clk	{0.000 5.000}	10.000	100.000
Design Timing Summary				
Clock Summary (1)				

# MATLAB Check:

Equation.m fd.m +

1 x=[0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31];

2 y=fft(x);

3 z=transpose(y);

4

Result	FFT
0	496.000000000000 + 0.000000000000000i
1	-16.0000000000000 + 162.450726201742i
2	-16.0000000000000 + 80.4374318740136i
3	-16.0000000000000 + 52.7449313430131i
4	-16.0000000000000 + 38.6274169979695i
5	-16.0000000000000 + 29.9338945886302i
6	-16.0000000000000 + 23.9456922026478i
7	-16.0000000000000 + 19.4960564094076i
8	-16.0000000000000 + 16.0000000000000i
9	-16.0000000000000 + 13.1308606532586i
10	-16.0000000000000 + 10.6908582067088i
11	-16.0000000000000 + 8.55217817521267i
12	-16.0000000000000 + 6.62741699796952i
13	-16.0000000000000 + 4.85354693771748i
14	-16.0000000000000 + 3.18259787807452i
15	-16.0000000000000 + 1.57586245371462i
16	-16.0000000000000 + 0.000000000000000i
17	-16.0000000000000 - 1.57586245371462i
18	-16.0000000000000 - 3.18259787807452i
19	-16.0000000000000 - 4.85354693771748i
20	-16.0000000000000 - 6.62741699796952i
21	-16.0000000000000 - 8.55217817521267i
22	-16.0000000000000 - 10.6908582067088i
23	-16.0000000000000 - 13.1308606532586i
24	-16.0000000000000 - 16.0000000000000i
25	-16.0000000000000 - 19.4960564094076i
26	-16.0000000000000 - 23.9456922026478i
27	-16.0000000000000 - 29.9338945886302i
28	-16.0000000000000 - 38.6274169979695i
29	-16.0000000000000 - 52.7449313430131i
30	-16.0000000000000 - 80.4374318740136i
31	-16.0000000000000 - 162.450726201742i

**Note:** We got the same results approximately due to fixed point usage (quantization error)