All Status Report

مقدمه: در این ازمایش می خواهیم کد نرم افزاری را به زبان سخت افزار تبدیل کنیم و در هر مورد pipeline و Unroll کردن لوپ را ببینیم و این کاد را با استفاده از Catapult انجام می دهیم

کد:C++

```
#include <ac_int.h>

void Adding (ac_int<8,false> A, ac_int<8,false> B, ac_int<8,false> C, ac_int<8,false> D, ac_int<8,false> E, ac_int<64,false> &Temp)
{
      ac_int<64,false> Temp2;
    out: for(int i=0;i<11;i++)
     {
       in: for(int i=0;i<6;i++)
        {
          Temp2+=(A+A+B+C+D)/4;
        }
       Temp2*=E*E;
     }
     Temp = Temp2;
}
```

با استفاده از Catapult:

section A: (Min Latency) 1

Frequency: 50 MHz

Register count:179

Logic element count:395

Area: 2033.57

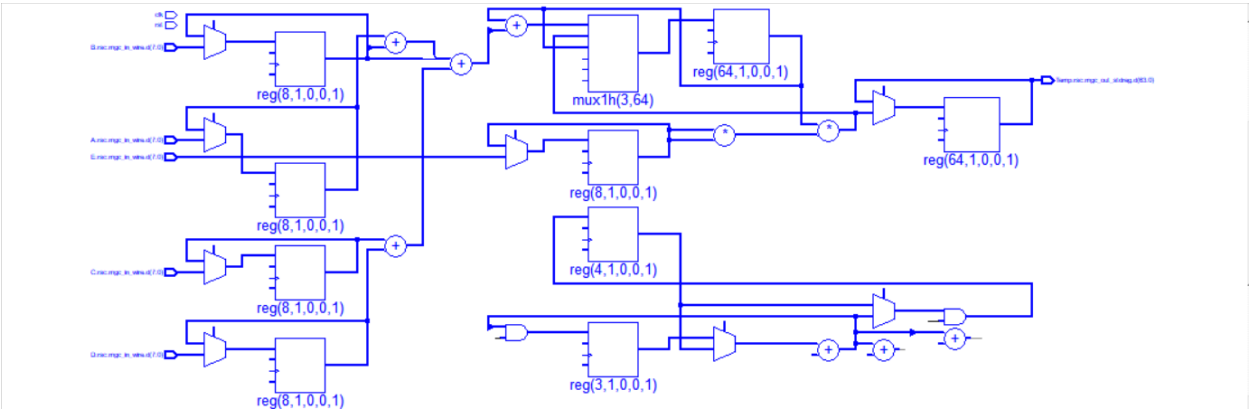
Latency cycle: 78

Latency time: 1560.00

Throughput cycle: 79

Throughput time: 1580.00

Data path:



All Status Report

section B:

این قسمت pipeline با II=1 امکان پذیر نبود

main Loop Pipelining (II=2) 2

Frequency: 50Mhz

Register count: 194

Logic element count: 532

Area: 1902.86

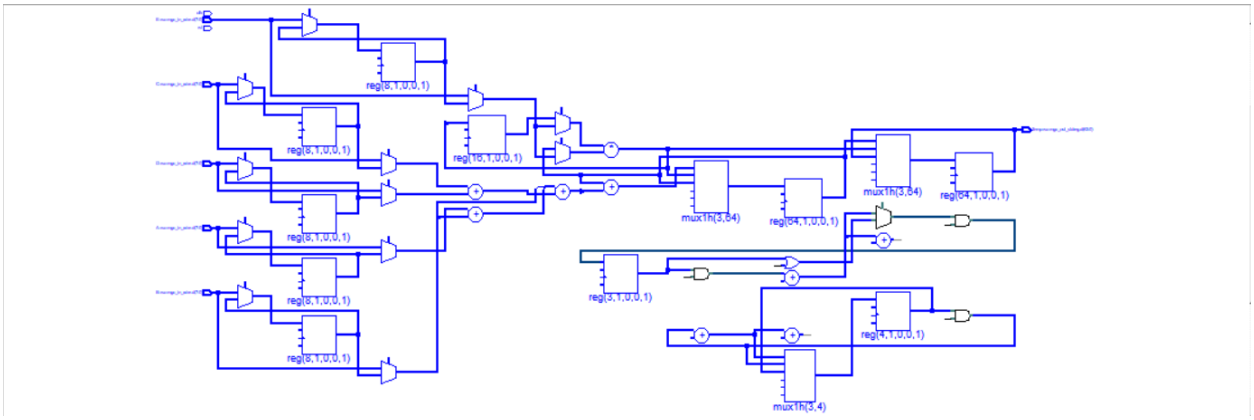
Latency cycle: 132

Latency time: 2640.00

Throughput cycle: 132

Throughput time: 2640.00

Data path:



outer Loop Pipelining (II = 2) 3

Frequency:50Mhz

Register count: 197

Logic element count: 489

Area:1855.71

Latency cycle: 133

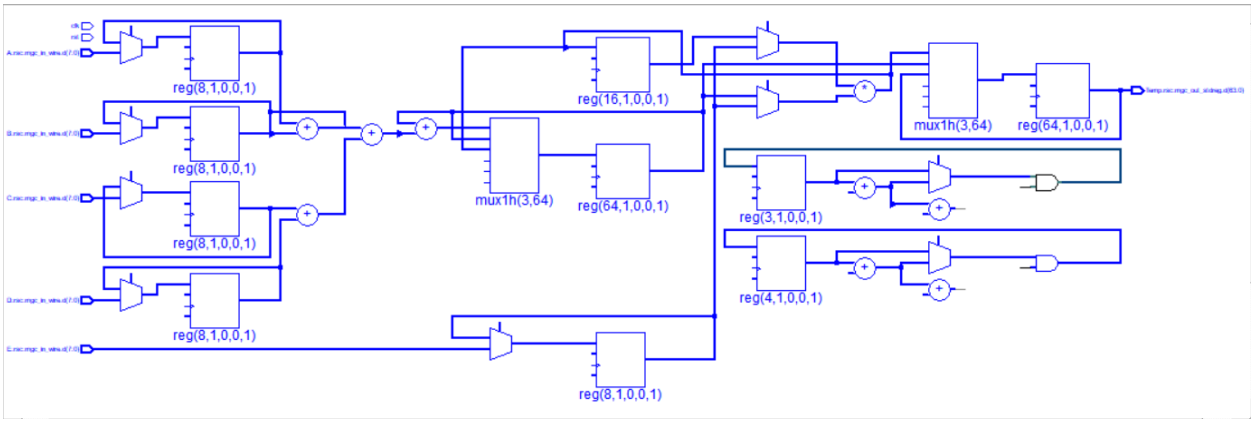
Latency time: 2660.00

Throughput cycle: 134

Throughput time: 2680.00

All Status Report

Data path:



inner Loop Pipelining (II = 2) 4

Frequency: 50Mhz

Register count: 181

Logic element count: 398

Area:2036.65

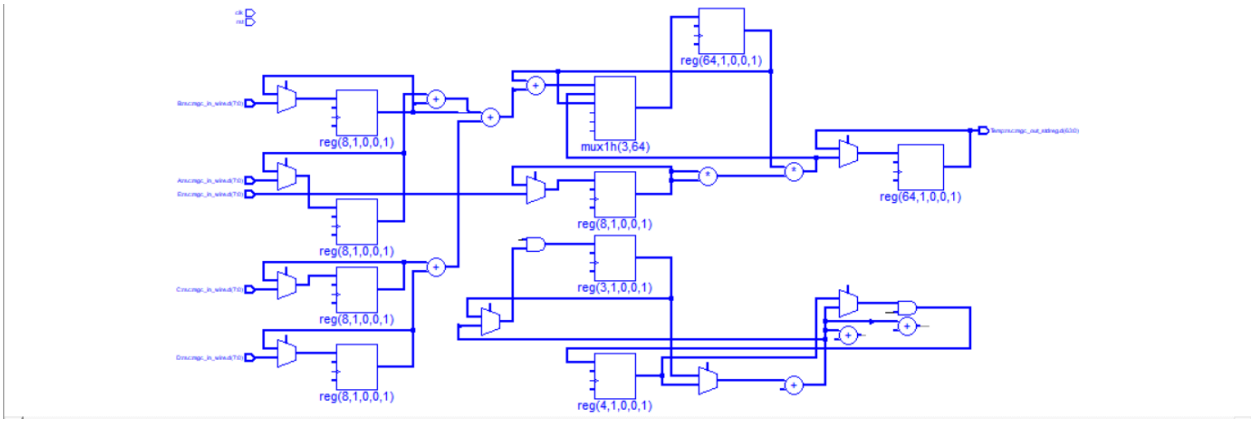
Latency cycle: 144

Latency time: 2880.00

Throughput cycle: 145

Throughput time: 2900.00

Data path:



عملکرد Pipeline: با Pipeline کردن مدار می توانیم با Area ی کمتر Latency بهتری بگیریم که به معنی سریع تر بودن مدار است و در بین این 3 مورد Pipeline کردن main از بقیه بهتر بوده است

section C:

outer loop unrolling 5

Frequency: 50Mhz

Register count: 274

Logic element count: 579

Area: 2156.89

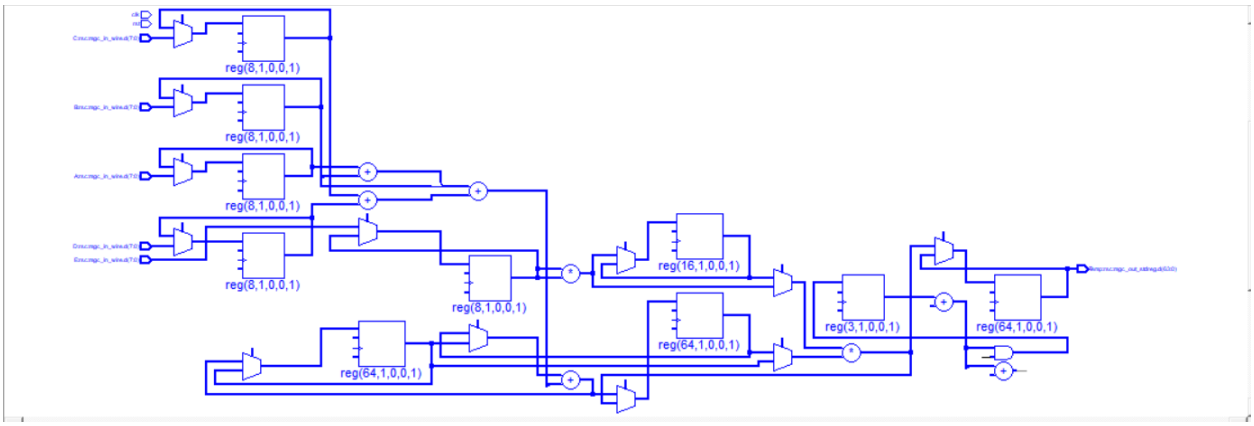
Latency cycle: 78

Latency time: 1560.00

Throughput cycle: 79

Throughput time: 1580.00

Data path:



inner loop unrolling 6

Frequency: 50Mhz

Register count: 256

Logic element count: 427

Area: 1681.82

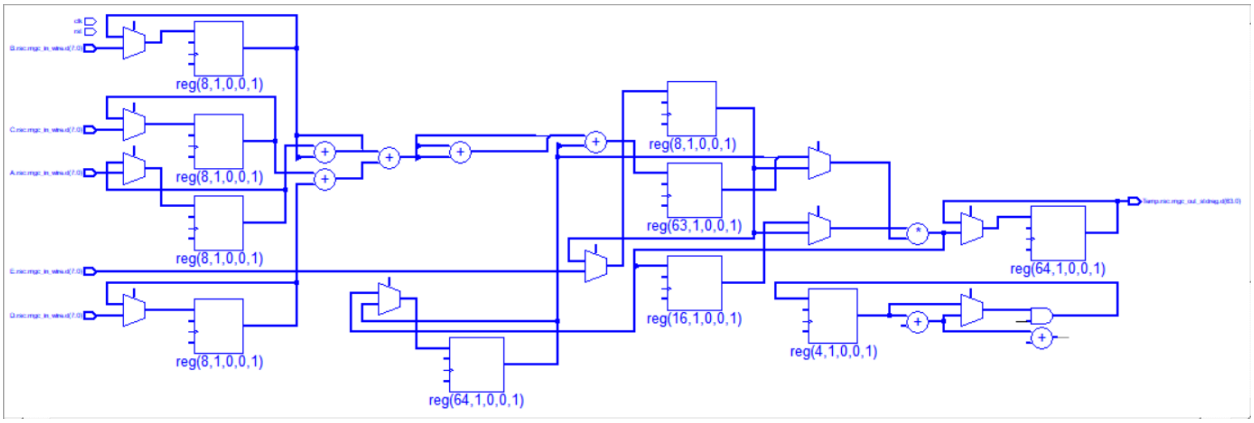
Latency cycle: 23

Latency time: 460.00

Throughput: 24

Throughput time: 480.00

Data path:



عملکرد Unroll: با Unroll کردن loop های مدار ما می توانیم Iteration ها را با هم اجرا کنیم و این کار Latencyمدار را کم می کند و سرعت بالا می رود ولی

Section D: (goal = Area) 7

Frequency: 50 MHz

Register count:197

Logic element count:492

Area: 1767.57

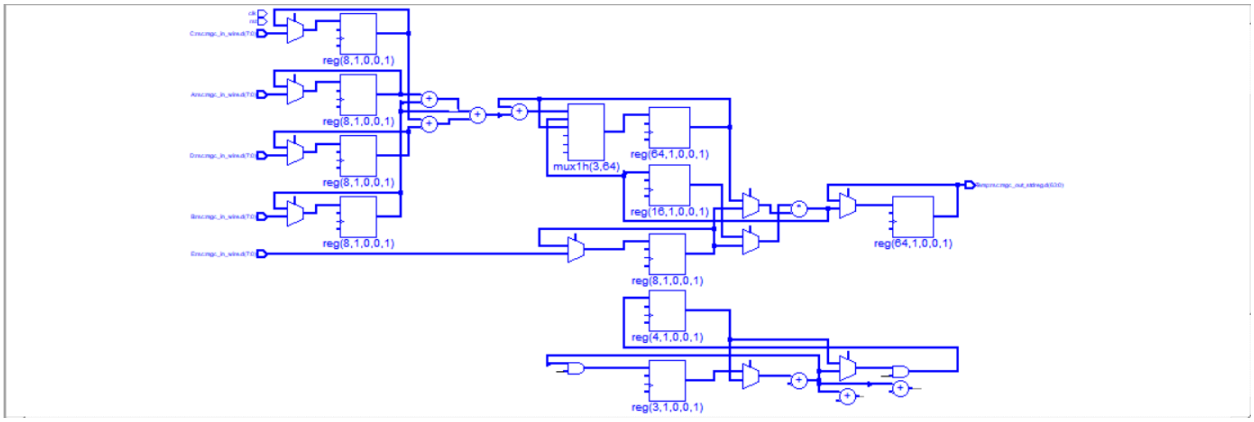
Latency cycle: 89

Latency time: 1780.00

Throughput cycle: 90

Throughput time: 1800.00

Data path:



All Status Report

section D_B:

Pipeline با II=1 قابل انجام نبوده است

main Loop Pipelining (II=2) 8

Frequency: 50MHz

Register count: 194

Logic element count: 532

Area: 1902.86

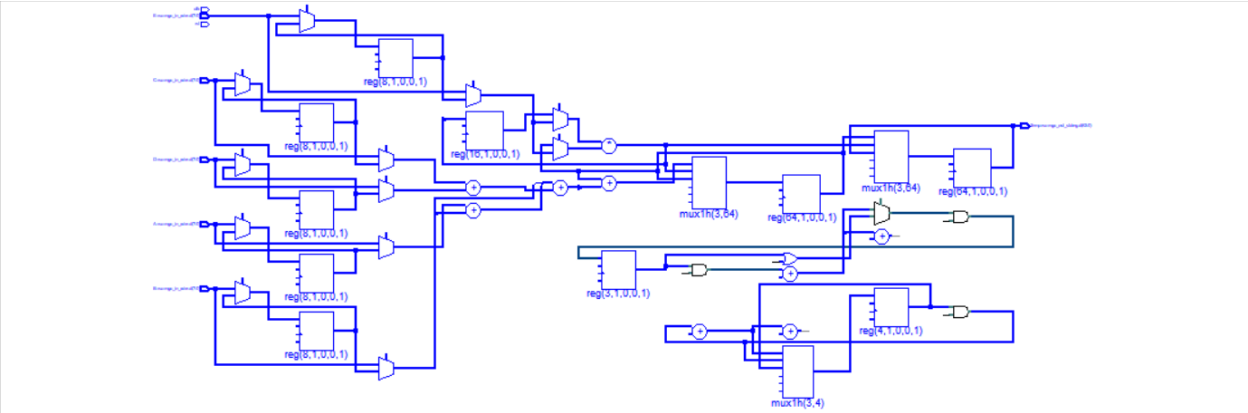
Latency cycle: 132

Latency time: 2640.00

Throughput cycle: 132

Throughput time: 2640.00

Data path:



outer Loop Pipelining (II=2) 9

Frequency:50MHz

Register count: 197

Logic element count: 489

Area:1855.71

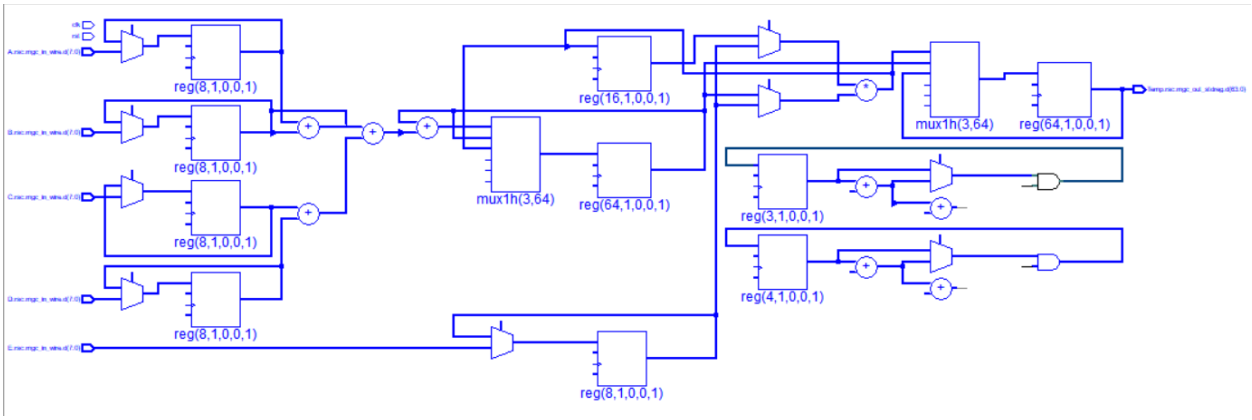
Latency cycle: 133

Latency time: 2660.00

Throughput cycle: 134

Throughput time: 2680.00

Data path:



inner Loop Pipelining (II = 2) 10

Frequency: 50MHz

Register count: 199

Logic element count: 196

Area:1771.38

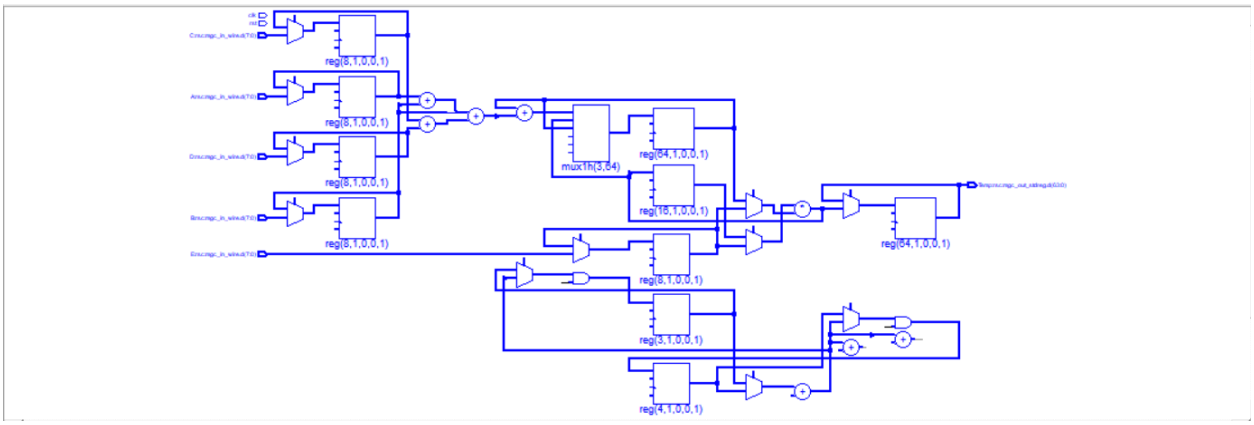
Latency cycle: 155

Latency time: 3100.00

Throughput cycle: 156

Throughput time: 3120.00

Data path:



All Status Report

section C:

outer loop unrolling 11

Frequency: 50MHz

Register count: 276

Logic element count: 589

Area: 1579.79

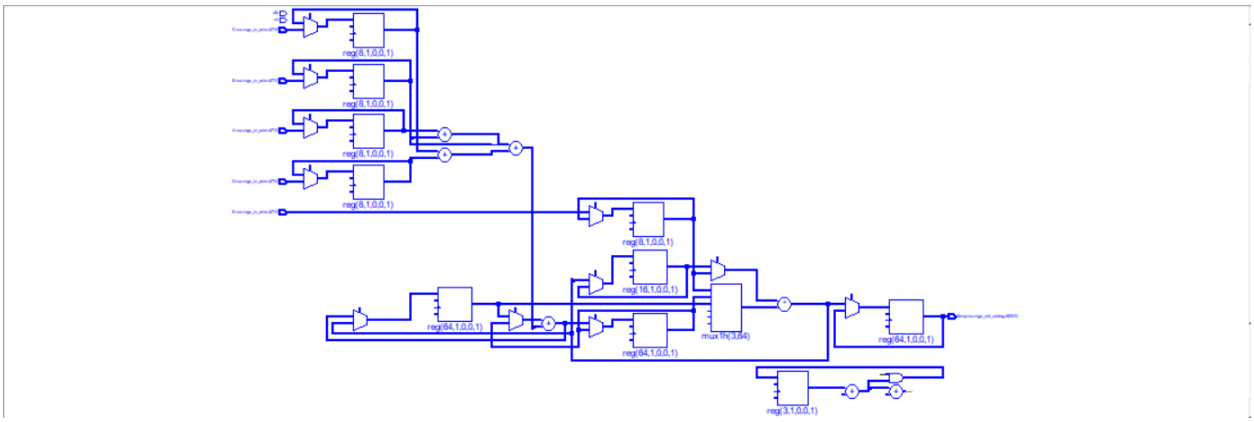
Latency cycle: 79

Latency time: 1898.79

Throughput cycle: 80

Throughput time: 1600.00

Data path:



inner loop unrolling 12

Frequency: 50MHz

Register count: 256

Logic element count: 427

Area: 1681.82

Latency cycle: 23

Latency time: 460.00

Throughput: 24

Throughput time: 480.00

All Status Report

Data path: