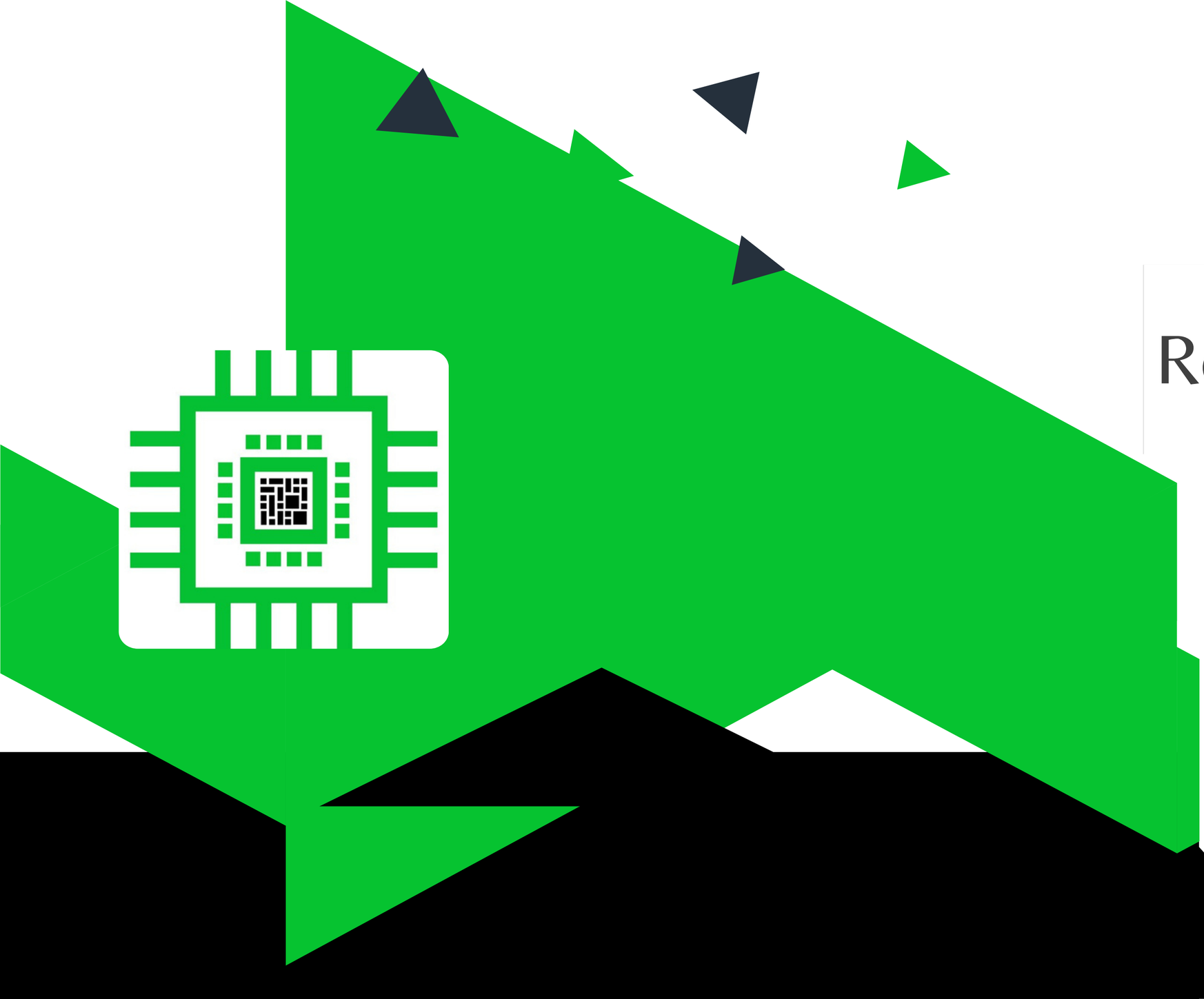


Research Update 10

Ebinesh K
IC Design Intern

20th March 2025

Hamming Code

Hamming Code Specifications:

Hamming (7,4)

Message length: 4

Parity Bits: 3

Total Bits: 7

Initial idea - CRC + BCH

Why not CRC + BCH?

Increased Latency:

- BCH decoding is iterative and takes multiple clock cycles, leading to higher delay in real-time applications.
- Hamming (7,4) allows for faster single-step correction.

No Built-in Correction in CRC:

- CRC can only detect errors, but it cannot correct them.
- Additional BCH decoding is required, increasing design complexity.

High Hardware Complexity:

- BCH decoding requires Galois field arithmetic, which increases circuit complexity.
- More logic gates and memory are needed compared to Hamming codes**.**

Higher Power Consumption:

- BCH requires complex polynomial division and multiple iterations, leading to higher power consumption.
- Hamming (7,4) with interleaving is a lightweight and power-efficient alternative.

Hamming Code

Structure of (7,4) Hamming Code:

- 4 Message bits: m1 ,m2, m3,m4
- 3 Parity bits: P1 ,P2, P3
- Total 7 bits: P1, P2, m1, P3, m2, m3, m4

P1 ---> c3 (1,3,5,7)
P2---> c2 (2,3,6,7)
P3---> c1 (4,5,6,7)

So the parity bits P1, P2 and P3 are placed at positions 1, 2 and 4

At receiver side

	c1	c2	c3
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Hamming Code

Bit Position (1 to 7)	Bit Type	Covered by Parity
1	P_1	Covers: 1, 3, 5, 7 (i.e., P_1, m_1, m_2, m_4)
2	P_2	Covers: 2, 3, 6, 7 (i.e., P_2, m_1, m_3, m_4)
3	m_1	Message bit
4	P_3	Covers: 4, 5, 6, 7 (i.e., P_3, m_2, m_3, m_4)
5	m_2	Message bit
6	m_3	Message bit
7	m_4	Message bit

The parity bits are calculated as follows:

P1=m1⊕m2⊕m4
P2=m1⊕m3⊕m4
p3=m2⊕m3⊕m4
(⊕ represents XOR operation)

Hamming Code

Example: Encoding

Let's say we want to encode the message bits m1,m2,m3,m4=1,0,1,1.

Step 1: Compute Parity Bits

P1=1⊕0⊕1=0
P2=1⊕1⊕1=1

P3=0⊕1⊕1=0

Step 2: Construct Codeword

So, the final (7,4) Hamming codeword is:
P1,P2,m1,P3,m2,m3,m4 = 0 1 1 0 0 1 1

Hamming Code

Error Detection & Correction:

When the received 7-bit codeword has a single-bit error, the receiver can detect and correct it using syndrome bits (C).

Step 1: Compute Syndrome (C) Bits

At the receiver, syndrome bits C1, C2, C3 are calculated:

- $C1 = P1 \oplus m1 \oplus m2 \oplus m4$
- $C2 = P2 \oplus m1 \oplus m3 \oplus m4$
- $C3 = P3 \oplus m2 \oplus m3 \oplus m4$

ThesyndromevalueC=C1C2C3gives the position of the error (if nonzero).

Step 2: Error Correction

- If C=000, no error.
- If $C \neq 000$, the binary value of C gives the position of the erroneous bit.
- Flip that bit to correct the error.

Hamming Code

Example: Error Correction

Suppose we receive 0 1 1 0 1 1 1 instead of 0 1 1 0 0 1 1.

Substituting values from the received codeword (0110111):

- $C1 = 0 \oplus 1 \oplus 1 \oplus 1 = 1$
- $C2 = 1 \oplus 1 \oplus 1 \oplus 1 = 0$
- $C3 = 0 \oplus 1 \oplus 1 \oplus 1 = 1$

Thus, the syndrome value is:
$C = C1C2C3 = B(101) = 5$

Error Detection:

- The syndrome 101 (binary) = 5 (decimal).
- This means the error is in bit position 5.

Hamming Code

Correcting the Error:

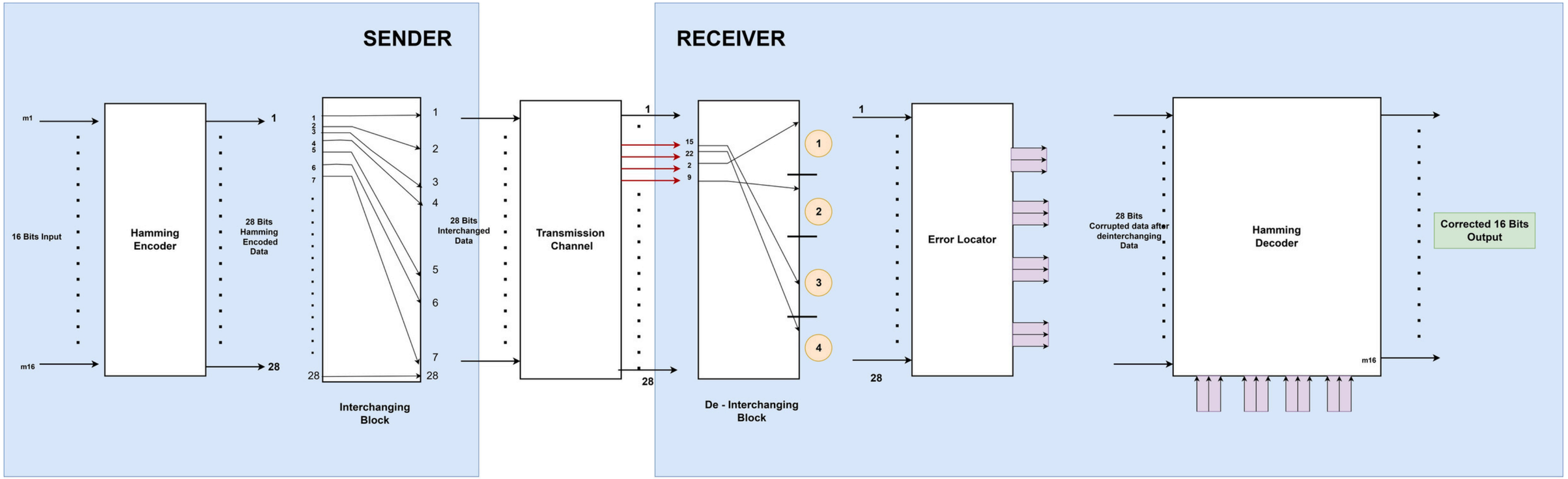
- The received codeword was: 0110111
- The error is at bit 5 (m_2).
- Flipping m_2 from $1 \rightarrow 0$, the corrected codeword is: 0110011

Hamming Code

Summary:

- The received codeword had an error at bit 5 (m_2).
- Using the syndrome bits, we detected the error location
- After flipping bit 5, the corrected codeword is 0110011.
- The original message bits are 1011.

Architecture



Interchanging the Hamming Encoder's output bits

Original Position	After Interchanging
1	1
2	8
3	15
4	22
5	2
6	9
7	16
8	23
9	3
10	10
11	17
12	24
13	4
14	11
15	18
16	25
17	5
18	12
19	19
20	26
21	6
22	13
23	20
24	27
25	7
26	14
27	21
28	28

Timing Verification

16-Bits	Power Consumption (mW)	Latency (ps)	Frequency (MHz)
Hamming Encoder	0.2485	168.14	-
Error Locator	0.42	252.21	-
Hamming Decoder	0.7608	74.08	-
Transmission Channel	0.1815	9	-
	1.6108	503.43	1986
Overall circuit	1.827		

Latency is calculated based on the critical path (using the delay of each gate on the longest path)

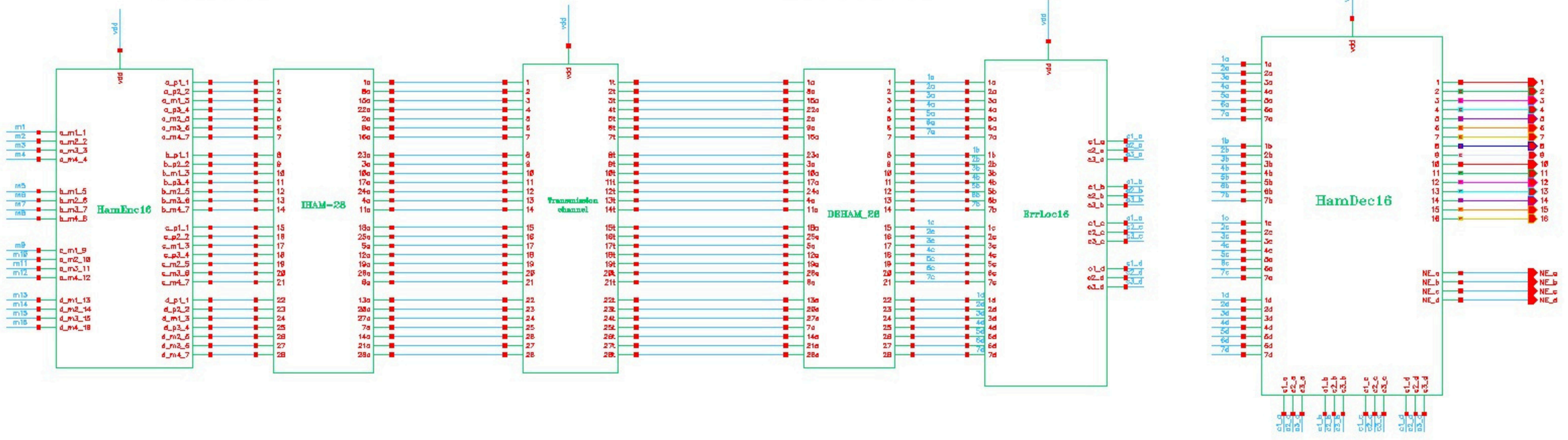
Timing Verification

Components	Rise delay (ps)	Fall delay (ps)	P.D (Latency) (ps)
Not gate	9.34	8.66	9
Or gate	21.31	65.72	43.52
And gate	38.35	27.23	32.9
Xor gate	98.73	69.4	84.07

Top level view

Sender

Receiver



Hamming Encoder - Propagation delay

Hamming Encoder	Output 1	Output 2	Output 3	Output 4	Output 5	Output 6	Output 7	Avg (ps)
Varying input bit	P.D (ps)	P.D (ps)	P.D (ps)	P.D (ps)	P.D (ps)	P.D (ps)	P.D (ps)	P.D (ps)
1	154.59	154.59	0	-	-	-	-	154.59
2	171.42	-	-	172.07	0	-	-	171.75
3	-	171.413	-	175.18	-	0	-	173.23
4	69.305	69.3025	-	71.83	-	-	0	70.14
Other input are kept high	' - ' indicates the outputs were always high (1)						Max	173.23

Error Locator - Propagation delay

Error Locator	P.D (ps)	P.D (ps)	P.D (ps)	P.D (ps)
Erroneous Bit	c1	c2	c3	Avg
1	246.57	-	-	246.57
2	-	244.02	-	244.02
3	-	246.521	258.6615	252.59
4	263.24	-	-	263.24
5	266.4755	-	156.624	211.55
6	163.096	171.574	-	167.33
7	81.0135	69.31	84.17	78.16
			Max	263.24

Hamming Decoder - Propagation delay

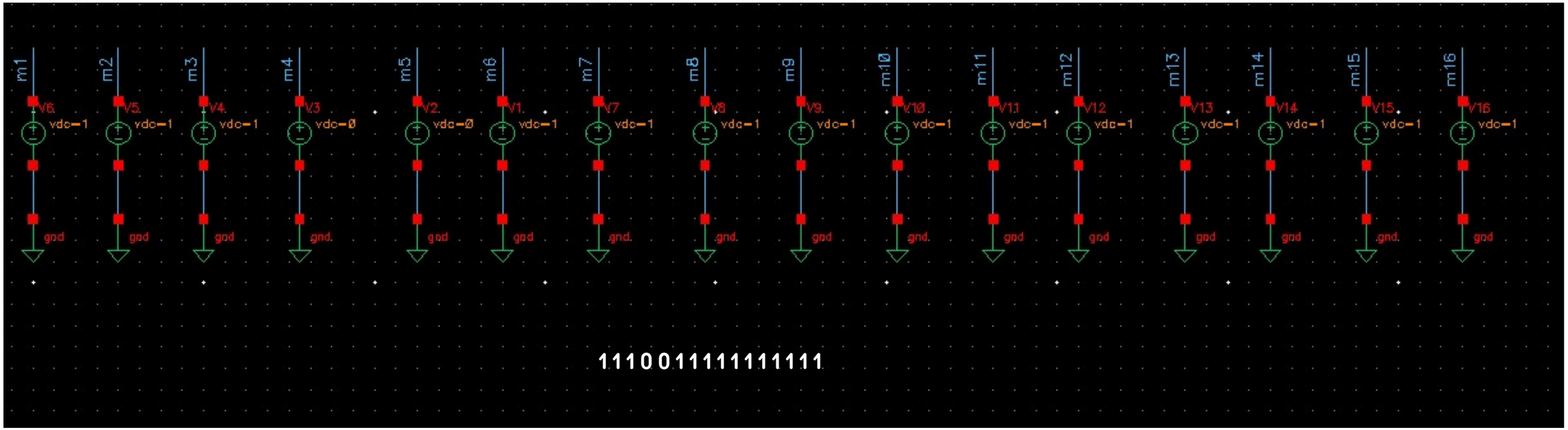
Bit to be corrected	P.D (ps)
1	153.19
2	184.56
3	153.195
4	184.68
5	153.19
6	184.45
7	153.2
Avg	166.64
Max	184.68

De-Interleave Block

- **The deinterleaving block will rearrange the bits back into their correct order, ensuring that no two errors occur within the same block.**
- **As a result, the four errors will be separated and distributed across four different blocks in the error locator stage.**

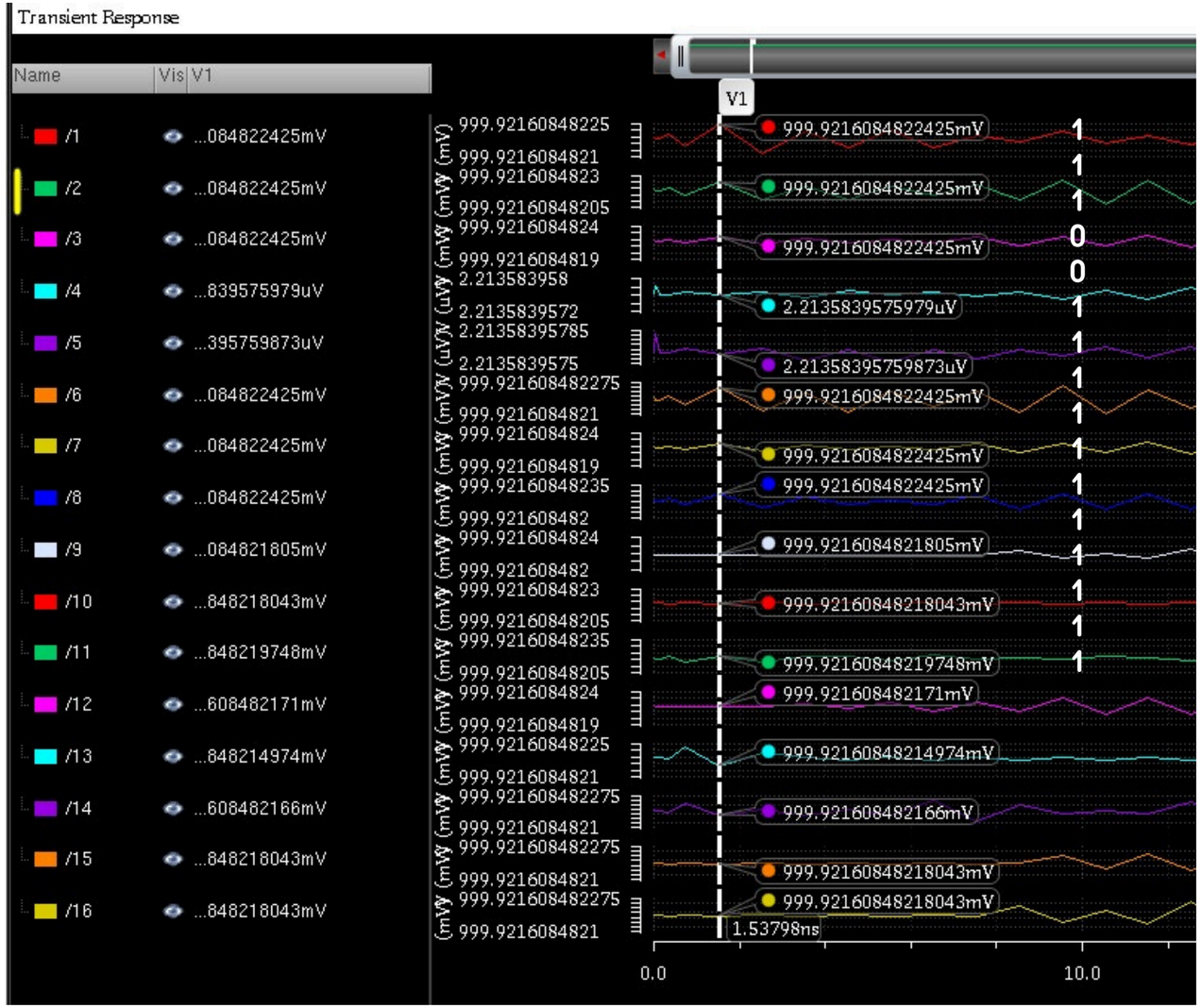
De-Interleave Block

Original Message

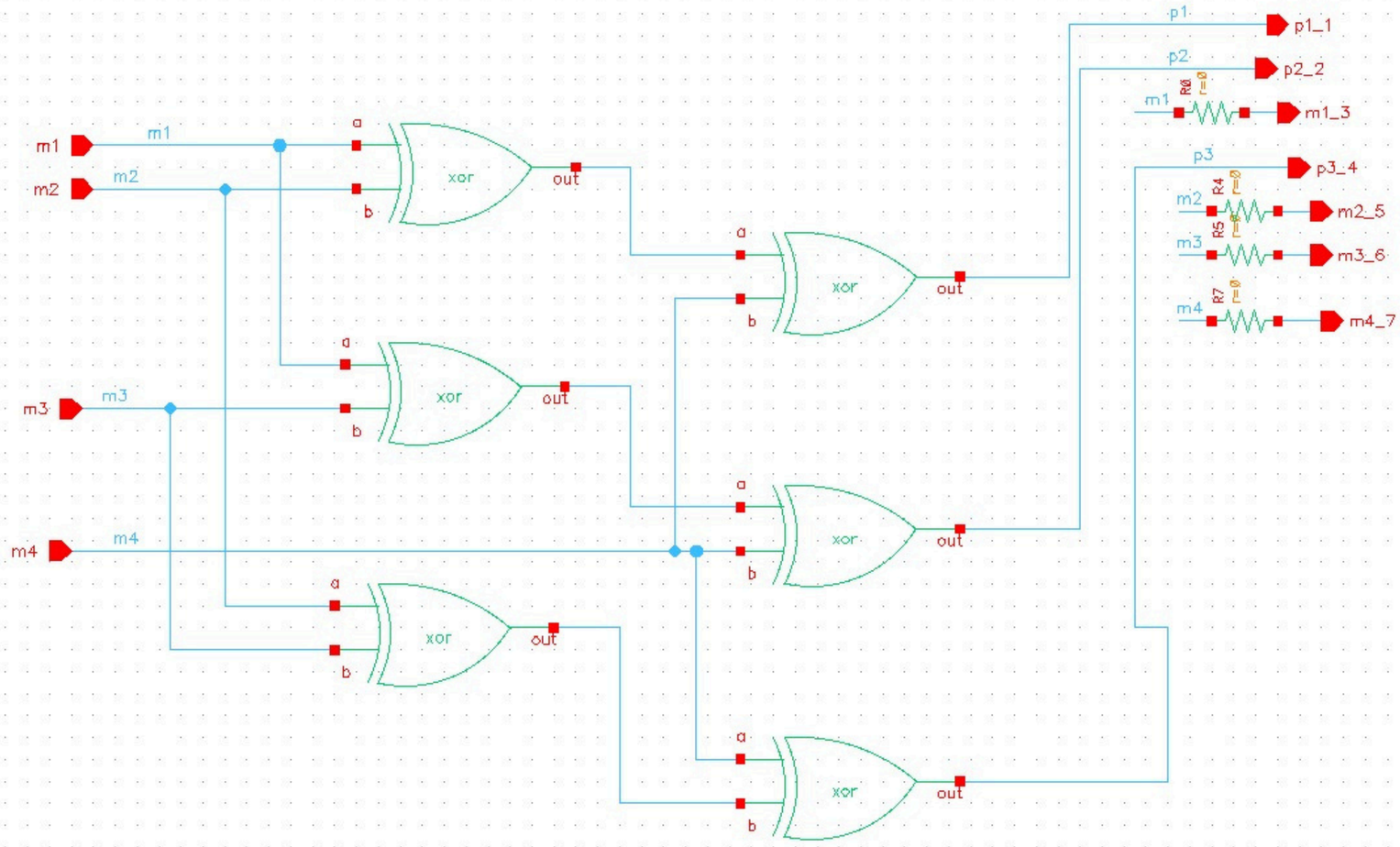


De-Interleave Block

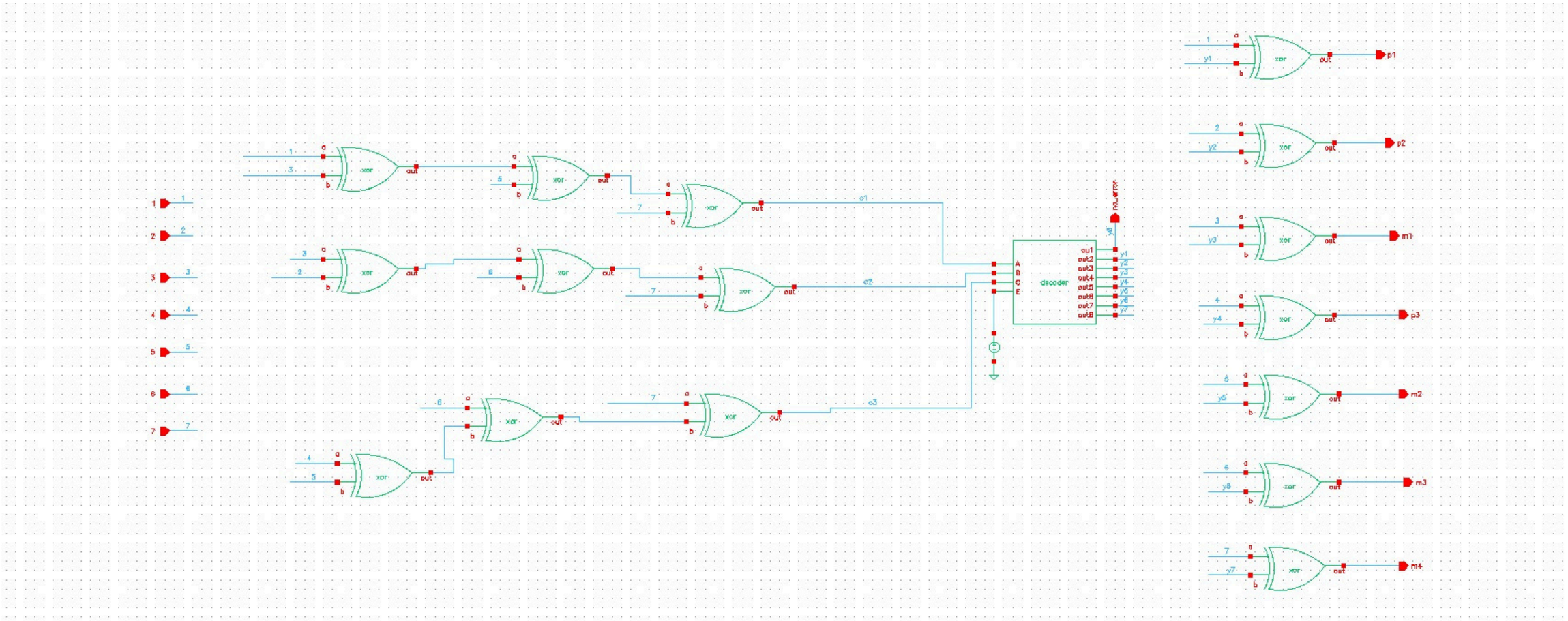
Final Ouput



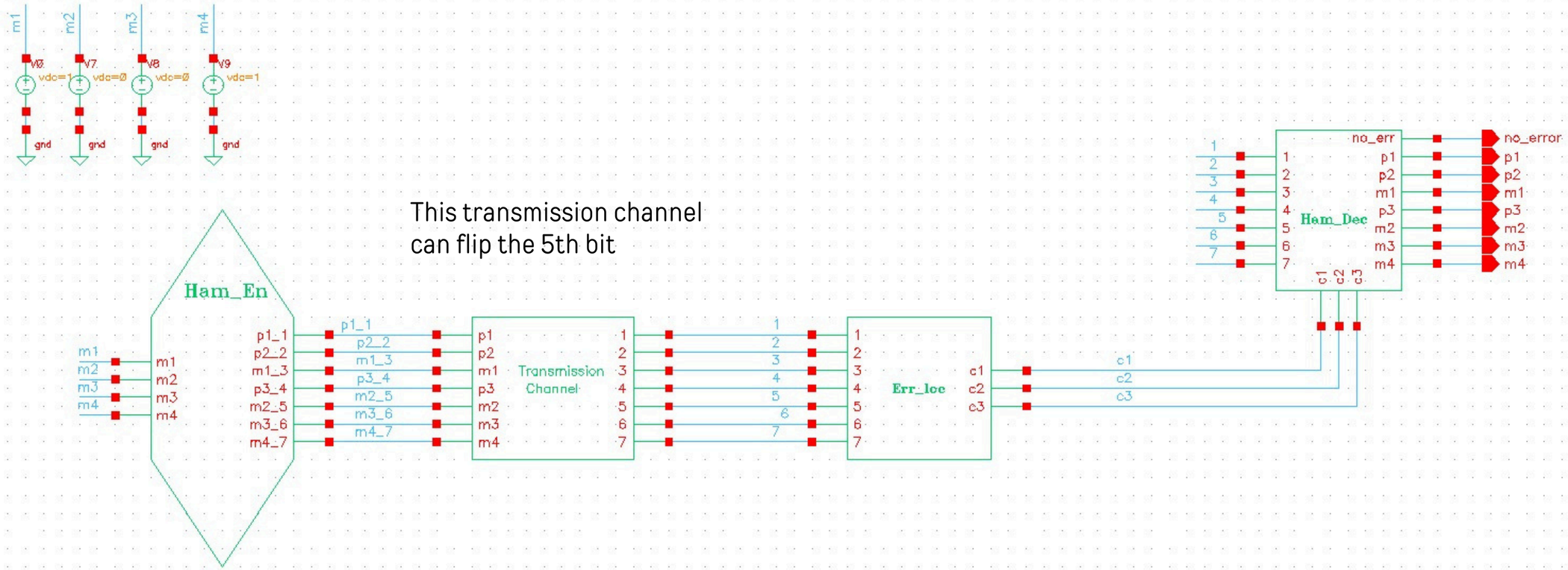
Hamming Encoder



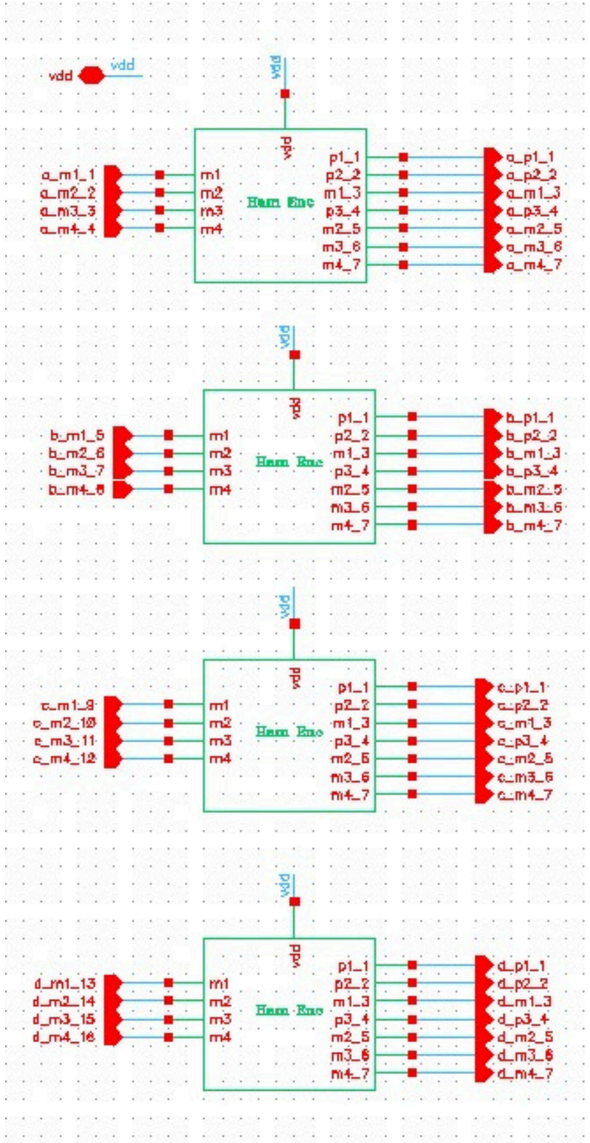
Hamming Decoder



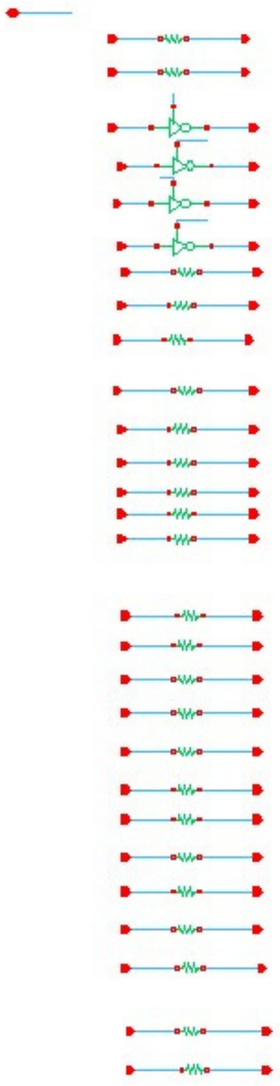
Hamming (7,4) Circuit

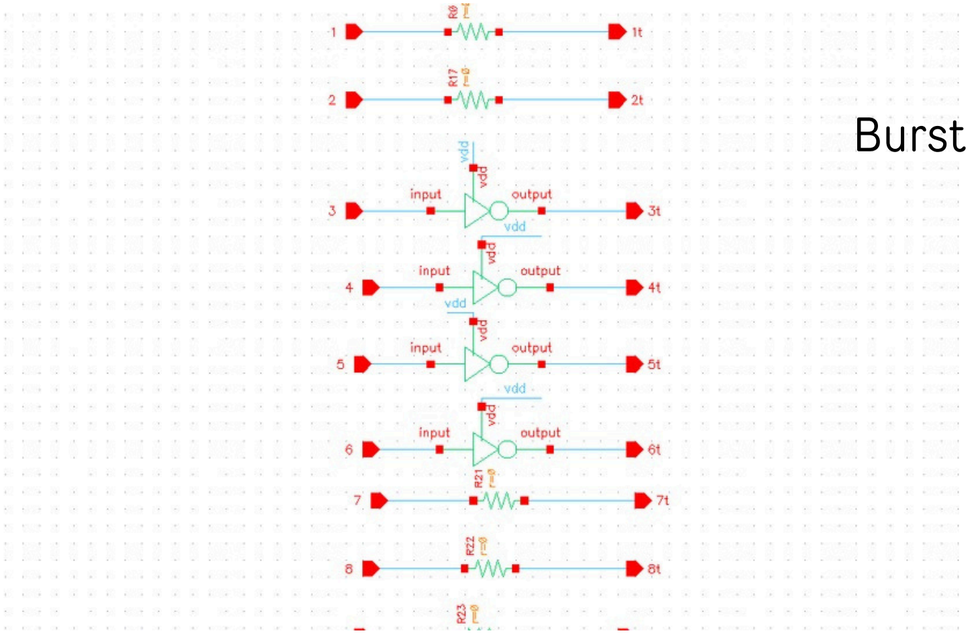


Hamming Encoder (16Bits)



Transmission Channel





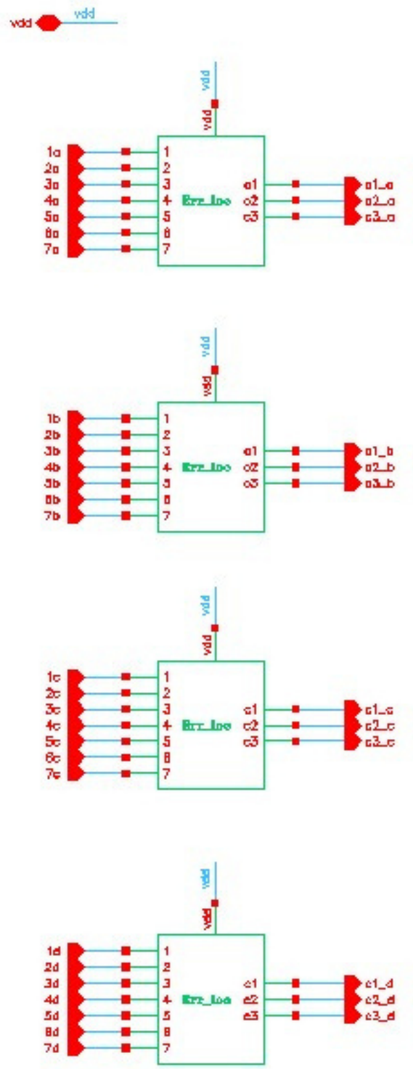
Burst Error (Consecutive Errors)

At bit 3,4,5,6

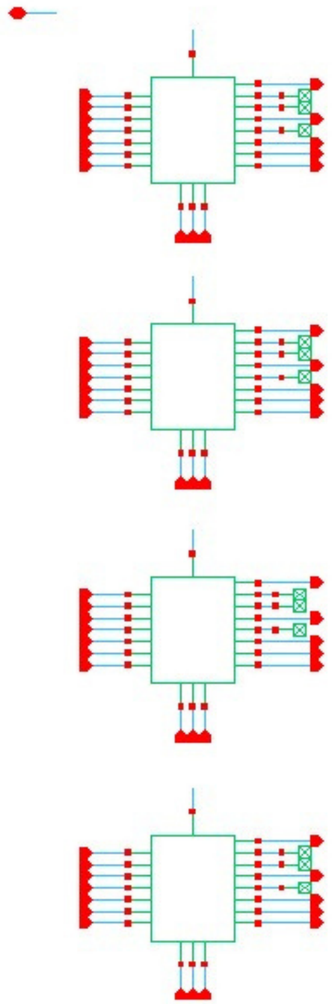
After Transmission Channel

The previous slide shows that the 3rd, 4th, 5th, and 6th bits have been flipped, indicating a consecutive error, which means a burst error has occurred.

Error Locator (16bit)



Hamming Decoder for 16 bits



Timing Verification for the whole circuit (from input till the final output)

# Bit	Rise delay (ps)	Fall delay (ps)	Propagation delay (ps)
1	71.32	88.21	79.765
2	53.5	79.73	66.615
3	53.5	25.14	39.32
4	53.5	79.611	66.5
5	53.5	79.7	66.6
6	53.48	79.61	66.546
7	53.48	79.61	66.546
8	53.5	79.7	66.6
9	53.5	79.78	66.64
10	53.5	79.72	66.61
11	53.5	79.8	66.65
12	53.5	79.79	66.645
13	53.5	79.56	66.53
14	53.5	79.56	66.53
15	53.5	79.51	66.505
16	53.5	75.75	64.625
		Avg	65.576
		Max	79.765