

# Product Code Iterative Decoder



Arif E. Nugroho

---

[arif.endro@opencores.org](mailto:arif.endro@opencores.org)



VLSI Research Group  
LabTek VIII Institut Teknologi Bandung  
Jl. Ganesha 10 Bandung 40141  
West Java, Indonesia

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Product Code . . . . .	1
1.2	Decoding Algorithm . . . . .	2
1.3	Circuit Schematic . . . . .	3
<b>2</b>	<b>Implementation</b>	<b>4</b>
2.1	Simulation . . . . .	4
2.2	Synthesize . . . . .	5
<b>A</b>	<b>Informations</b>	<b>7</b>
A.1	Warranty . . . . .	7
A.2	Tools . . . . .	7

# List of Figures

1-1	Sequence of Product Codes . . . . .	1
1-2	Product Code generations . . . . .	2
1-3	Schematic of Product Code Decoder . . . . .	3
2-1	Simulation waveform . . . . .	4

# 1

## Introduction

### 1.1. Product Code

Product code is also known as turbo code, this error correction methods is known to approach the Shannon Limit. This design uses iterative methods on decoding the product codes, see Figure 1-3, this design is based on Mr. Wada-san homepage[1].

This is two dimensional product code iterative decoder, there are four bits information followed by two row parity bits and two column parity bits. Each signal informations is represented in two's complement eight bit data, thus it indicate an integer value of -128 to 127 for each of information bit.

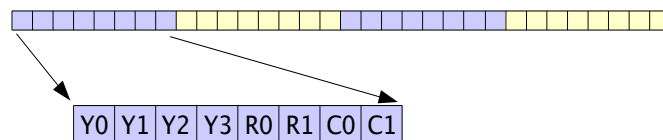


Figure 1-1: Sequence of Product Codes

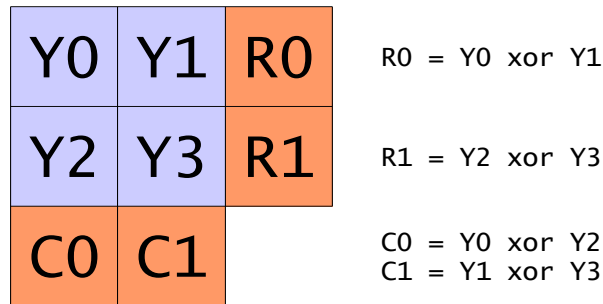


Figure 1-2: Product Code generations

## 1.2. Decoding Algorithm

$$\text{posteriori value} = \text{channel value} (Lch) + \text{priori value} + \text{external value} (Le) \quad (1-1)$$

$$\left. \begin{array}{l} \text{posteriori} = Lch + \text{priori} + Le \text{ (row parity)} \quad (0) \\ \text{posteriori} = Lch + \text{priori} + Le \text{ (column parity)} \quad (1) \\ \dots \\ \dots \\ \text{posteriori} = Lch + \text{priori} + Le \text{ (row parity)} \quad (n - 1) \\ \text{posteriori} = Lch + \text{priori} + Le \text{ (column parity)} \quad (n) \end{array} \right\} \quad (1-2)$$

$$Lch = Y0, Y1, Y2, Y3 \quad (1-3)$$

$$\text{priori} = \text{posteriori} (n - 1) \quad (1-4)$$

$$Le = \text{sgn}(a * b) * \min\{\text{abs}(a), \text{abs}(b)\} \quad (1-5)$$

$\text{sgn}(a * b)$  means the sign result of multiplication between operand  $a$  and  $b$ , and  $\text{abs}(x)$  means absolute value of operand  $x$ . The last value of posteriori is the decoded informations, i.e the posteriori value at  $n^{\text{th}}$  iterations. The decoded informations can be obtained from the sign of the last posteriori value, positive value is zero and negative

value is one, i.e this is the most significant bit of the posteriori value.

### 1.3. Circuit Schematic

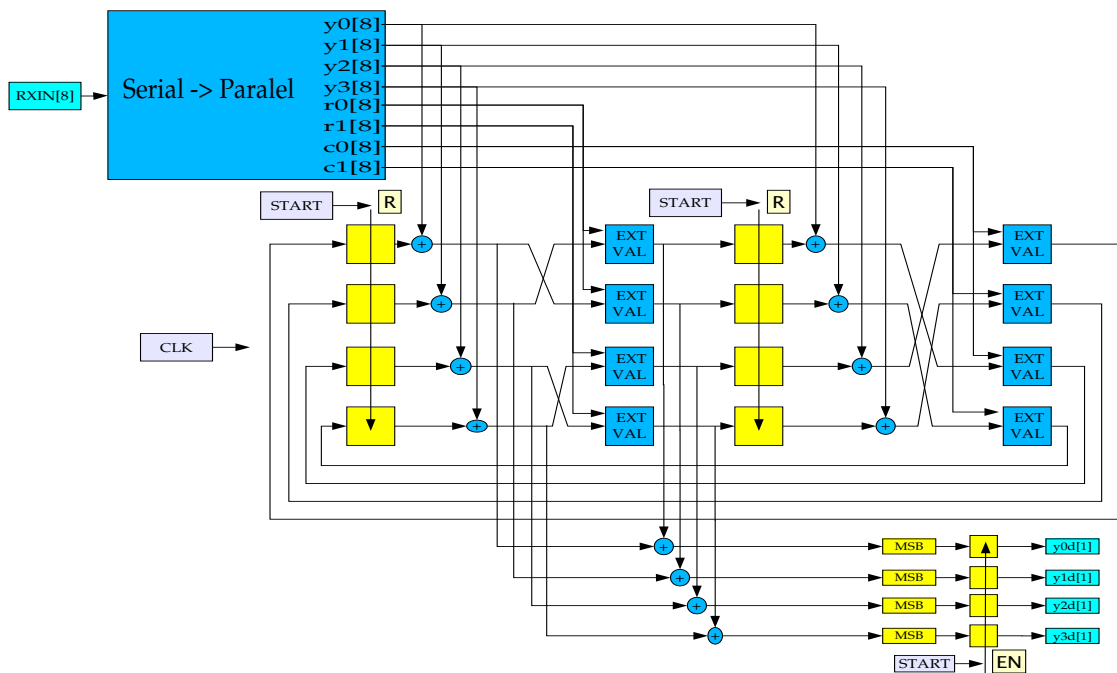


Figure 1-3: Schematic of Product Code Decoder

# Implementation

## 2.1. Simulation

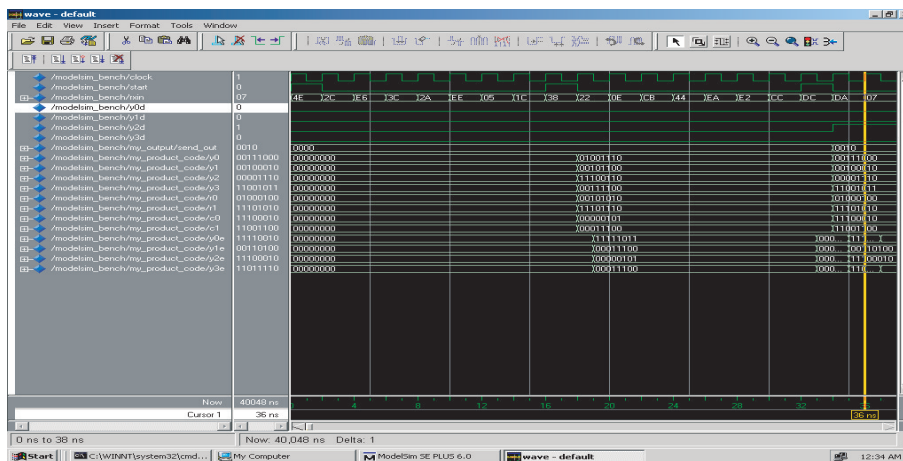


Figure 2-1: Simulation waveform

This design has been simulated using ModelSim 6.0 SE, here is the summary of bit errors on different signal to noise ratio (SNR) of input signal:

SNR(dB)	BIT ERRORS
100	0/10000
9	0/10000
6	12/10000
3	279/10000
0	1314/10000

Table 2-1: Bit errors on different SNR

signal with SNR 0 dB is signal with very big noise.

## 2.2. Synthesize

This design has been synthesized using ISE Xilinx 6.3i, here is the summary of the area utilization in FPGA Xilinx:

XC2V2000-FF896-4	
Slices	544/10752
Slices Flip Flops	203/21504
4 input LUT	922/21504
Total Equivalent gate count	<b>7294</b>

Table 2-2: Area utilizations summary

The maximum clock frequency is 64.045 MHz (Minimum period 15.614ns)



# Bibliography

- [1] Tom Wada, **2-D Product Code Iterative Decoder**,  
[http://www.ie.u-ryukyu.ac.jp/~wada/design06/spec\\_e.html](http://www.ie.u-ryukyu.ac.jp/~wada/design06/spec_e.html)  
October 1<sup>st</sup>, 2005

# A

## Informations

### A.1. Warranty

#### NO WARRANTY

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

### A.2. Tools

- ALLIANCE CAD SYSTEM developed by ASIM team at ©LIP6/Université Pierre et Marie Curie, <http://asim.lip6.fr/recherche/alliance>  
The primary VHDL Analyser for Synthesize
- ModelSim 6.0 The Simulator
- Xilinx 6.3i The Synthesizer

- **VIM** (Vi IMproved) The Editor
- **LaTeX** The Typesetter

**Version: 1.0**