

C_011

ECC 回路内蔵 16 ビットマイコンの設計 Design of 16 bit Microcomputer with ECC Circuit

姜 珂 野平 江梨 鈴木 五郎
Ke Jiang Eri Nohira Goro Suzuki

北九州市立大学情報メディア工学科
Department of Information Science, University of Kitakyushu

1. はじめに

現在、携帯電話、デジタルカメラをはじめ、様々なデジタル家電製品の普及が著しくなっている。100nm を切る deep sub- μ を使った信号処理用 LSI では、(1) α 線によるソフトエラーに対する高信頼性化[1]、(2)量込み演算の高速化、(3)低消費電力化、が重大な課題となっている。これらの課題を解決する 16 ビットマイコンの設計を行った[2]ので報告する。

2. アーキテクチャ

16 ビットマイコンの基本的なアーキテクチャは次の通りである。

- register 方式：汎用的な 8 種類の register (R0~R7)
- 命令長：16bit 固定命令長 (図 1)

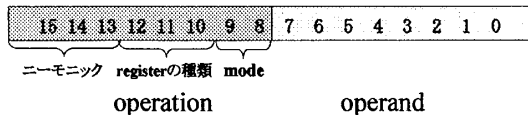


図 1 1 命令の構成

- アドレッシングモード：direct/immediate/indirect (register)
- CPI (cycle per instruction) : 8
 - 1-4 サイクル：命令フェッチ/命令デコード
 - 5-8 サイクル：データフェッチ/命令実行
- Machine cycle (clock cycle) : 5 nsec (200MHz)
- 命令セット：148 種類
- ブロックダイアグラム (図 2)

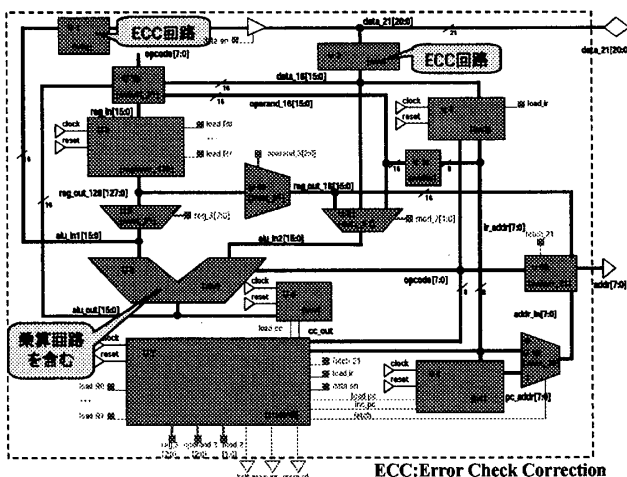


図 2 ブロックダイアグラム

3. 高信頼性化

ソフトエラーの対策には ECC(Error Correction Code)回路を具備することとした。パースト誤りの訂正が可能な巡回符号方式[3]とし、割り算回路にはサイクル数の増加が避けられる組み合わせ回路方式を採用した。

ECC 回路[4]は、誤り箇所の特定を行うために必要な 5bit の check code を生成する ECG(Error Check-code Generator)回路 (図 2 のモジュール U1) と、メモリから読み込んだデータに誤りがないかを計算し、誤りがあった場合にはその検出と訂正を行う CAC(Check And Correction)回路 (図 2 のモジュール U2) から構成することにした。

巡回符号を扱うには、1, 0 を多項式で表現し、data を $W_d(x)$ 、check data を $W_c(x)$ 、メモリに書き込むデータを $W(x)$ とし、generator polynomial を $G(x)$ 、quotient polynomial を $Q(x)$ とする。ECG では、

$$\overbrace{W_d(x)}^{m \text{ bit}} \overbrace{W_c(x)}^{n-m \text{ bit}} \quad n = 21, \quad m = 16$$

$$x^5 \cdot W_d(x)/G(x) = Q(x) + W_c(x)$$

$$(W(x) = x^5 \cdot W_d(x) + W_c(x) = G(x) \cdot Q(x))$$

$$G(x) = x^5 + x^2 + 1$$

のように、check data を決定した。

メモリから読み込んだデータ $R(x)$ にエラー $e(x)$ が加わったと仮定する。CAC では、

$$\begin{aligned} R(x)/G(x) &= \{W(x) + e(x)\}/G(x) \\ &= W(x)/G(x) + e(x)/G(x) \\ &= e(x)/G(x) \end{aligned}$$

の式より剰余が 0 なら誤りなしと判定し、0 でない場合は誤りがあると判定できる。剰余と誤り箇所とは一対一に対応することから、エラー箇所が同定でき、該当ビット (パーストエラーを含む) の修正を行う。

cyclic code で必要となる割り算回路には組み合わせ論理を用いて、検討した。組み合わせ論理を用いた回路構造を図 3 に示す。

制御論理

load_ir

clock

instruction register

in[15:0]

ld clk

out[15:0]

チップ外から供

clock

load_ir

ld_clk

in[15:0]

out[15:0]

トグル低減:
最大1/8

全てのF/1
に適用

180