

1 概要

BCH符号(ハミング符号含む)の符号化/復号化モジュールです。2ビットエラー訂正/3ビットエラー検出(Single Error Correction/Double Error Detection:SEC/DED)、または、1ビットエラー訂正/2ビットエラー検出(Double Error Correction/Triple Error Detection:DEC/TED)を行うことができます。サポートする入力ビットなどのパラメータを、表 1 に示します。

表 1:サポート状況

データ幅	パリティ幅	エラー訂正	エラー検出
16	11	2	3
32	13	2	3
16	6	1	2
32	7	1	2

2 ブロック構成

符号化モジュール(bch_enc)のブロック図を図 1 に、復号化モジュール(bch_dec)のブロック図を図 2 に示します。

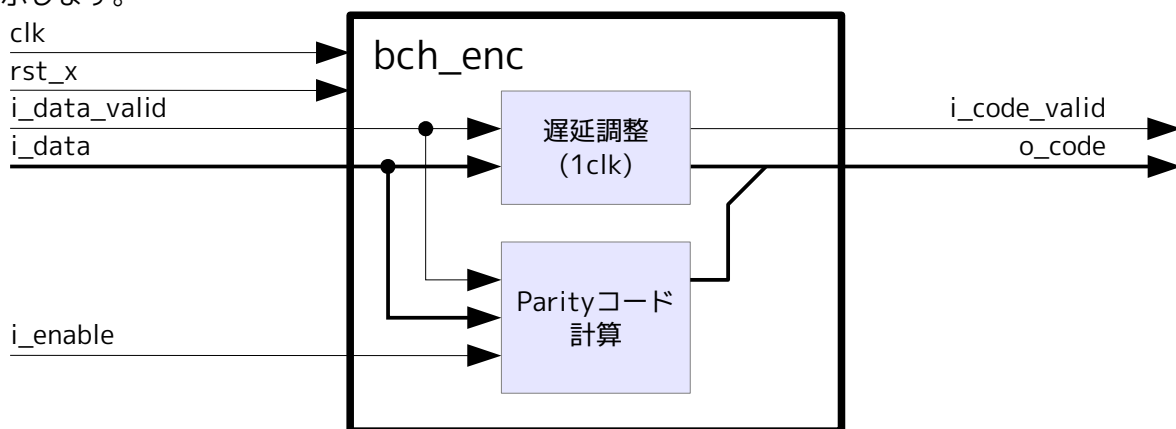


図 1: ブロック図(符号化モジュール)

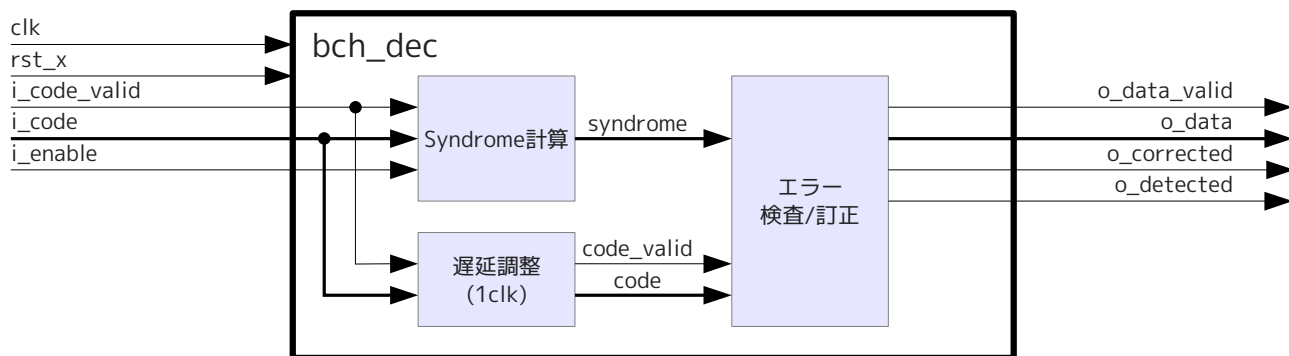


図 2: ブロック図(復号化モジュール)

3 パラメータ/入出力ポート

3.1 パラメータ

パラメータの一覧を表 2 に示します。

表 2:パラメーター一覧

パラメータ名	設定値	説明
pDataWidth	16/32	符号化入力/復号化出力データ幅です。
pErrorNum	1/2	訂正可能なエラーの個数です。
pExtend0n	0/1	pErrorNum + 1 個のエラー検出を可能にする、拡張符号化の 0n/0ff です。 0 : 拡張符号化 0ff 1 : 拡張符号化 0n (現状、1 のみ設定可能です。)
pCodeWidth	表 3 参照	符号化出力/復号化入力コード幅(データ幅 + パリティ幅)です。 設定値は自動的に設定されるので、インスタンス時に上書きしないで下さい。

- pCodeWidth の値は、pDataWidth, pErrorNum, pExtend0n から自動的に設定されます。設定値は表 3 を参照してください。

表 3:pCodeWidth 値一覧

pDataWidth	pErrorNum	pExtend0n	pCodeWidth
16	2	1	27
32	2	1	45
16	1	1	22
32	1	1	39

3.2 入出力ポート

符号化モジュールの入出力ポート一覧を表 4 に、復号化モジュールの入出力ポート一覧を表 5 に示します。

表 4: 符号化モジュール入出力ポート一覧

ポート名	I/O	ビット幅	説明
clk	Input	1	クロックです。
rst_x	Input	1	非同期リセットです。(Low Active)
i_enable	Input	1	符号化イネーブルです。符号化無効時は、パリティビットの生成を行いません。 0 : 符号化無効 1 : 符号化有効
i_data_valid	Input	1	入力データバリッドです。
i_data	Input	pDataWidth	入力データです。 i_data_valid = 1 の時に有効です。
o_code_valid	Output	1	出力コードバリッドです。
o_code	Output	pCodeWidth	出力コードです。 o_code_valid = 1 の時に有効です。

表 5: 復号化モジュール入出力ポート一覧

ポート名	I/O	ビット幅	説明
clk	Input	1	クロックです。
rst_x	Input	1	非同期リセットです。(Low Active)
i_enable	Input	1	復号化イネーブルです。復号化無効時は、エラー訂正/検出を行わずに、入力をそのまま出力します。 0 : 復号化無効 1 : 復号化有効
i_code_valid	Input	1	入力コードバリッドです。
i_code	Input	pCodeWidth	入力コードです。 i_code_valid = 1 の時に有効です。
o_data_valid	Output	1	出力データバリッドです。
o_data	Output	pDataWidth	出力データです。 o_data_valid = 1 の時に有効です。
o_corrected	Output	1	o_corrected = 1 でエラー訂正を行ったことを示します。 o_data_valid = 1 の時に有効です。
o_detected	Output	1	o_detected = 1 で pErrorNum + 1 ビットのエラーを検出したことを示します。 o_data_valid = 1 の時に有効です。

4 動作概要

4.1 入出力コードビットアサイン

入出力コードは、上位 pCodeWidth ビットにデータ部、残りの下位ビット(pCodeWidth - pDataWidth)にパリティ部が配置されています。ビットアサインのイメージを図 3 に示します。

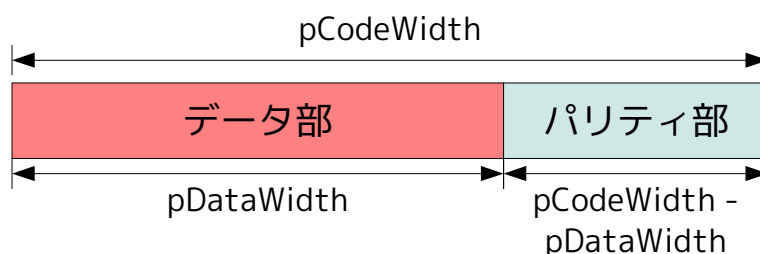


図 3: 入出力コードビットアサインイメージ

4.2 符号化モジュール動作概要

i_data_valid = 1 の時の入力データ(i_data)に対してパリティビットを生成し、符号化を行います。

i_data を入力した 1clk 後に、o_code_valid をアサートし、符号化コード(o_code)を出力します。

また、符号化無効(i_enable = 0)時は、パリティビットの生成を行わず、図 3 のパリティ部を 0 として、符号化コードを出力します。

符号化モジュールのタイミングチャートを図 4 に示します。

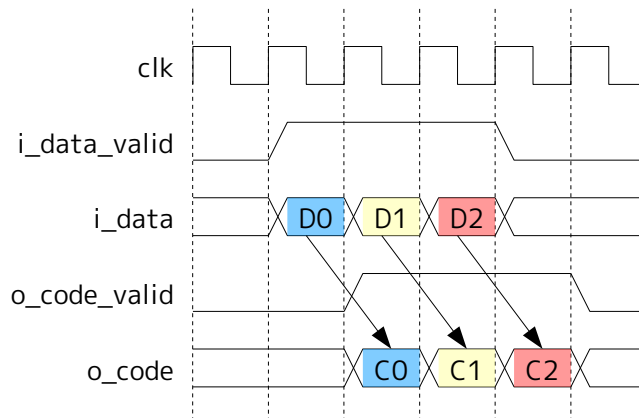


図 4: 符号化モジュールタイミングチャート

4.3 復号化モジュール動作概要

i_code_valid = 1 の時の入力コード(i_code)に対して、復号化を行います。i_code を入力した 3clk 後に、o_data_valid をアサートして、復号化データ(o_data)を出力します。同時に動作ステータス信号(o_corrected, o_detected)も出力し、エラー訂正を行った時は o_corrected を、pErrorNum + 1 ビットのエラーを検出した時は o_detected をアサートします。エラーなし/エラー検出時の復号化データは図 3 のデータ部をそのまま、エラー訂正時の復号化データはエラー訂正を施した図 3 のデータ部を出力します。

また、復号化無効(i_enable = 0)時は、エラー訂正/検出は行わず、図 3 のデータ部をそのまま出力し、動作ステータス信号の出力も行いません。

復号化モジュールのタイミングチャートを図 5 に示します。図中の C0 がエラーなし、C1 が訂正可能なエラーあり、C2 が検出可能なエラーありとします。

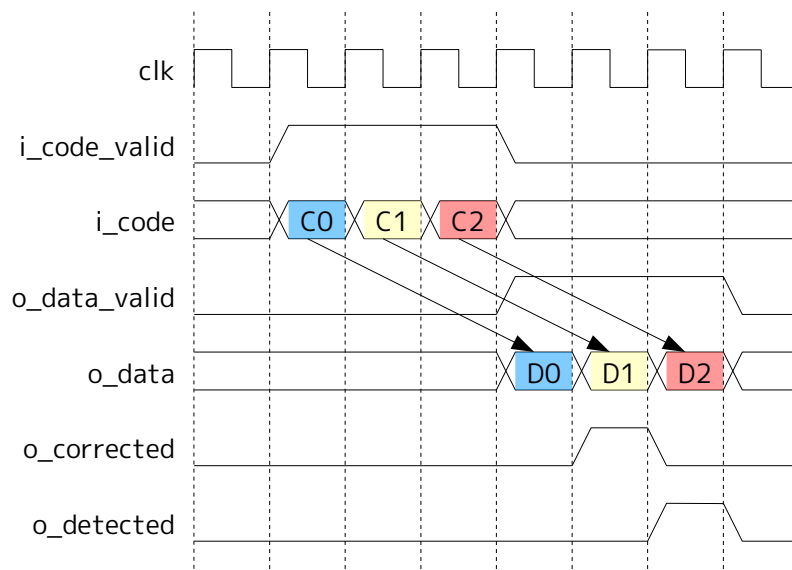


図 5: 復号化モジュールタイミングチャート