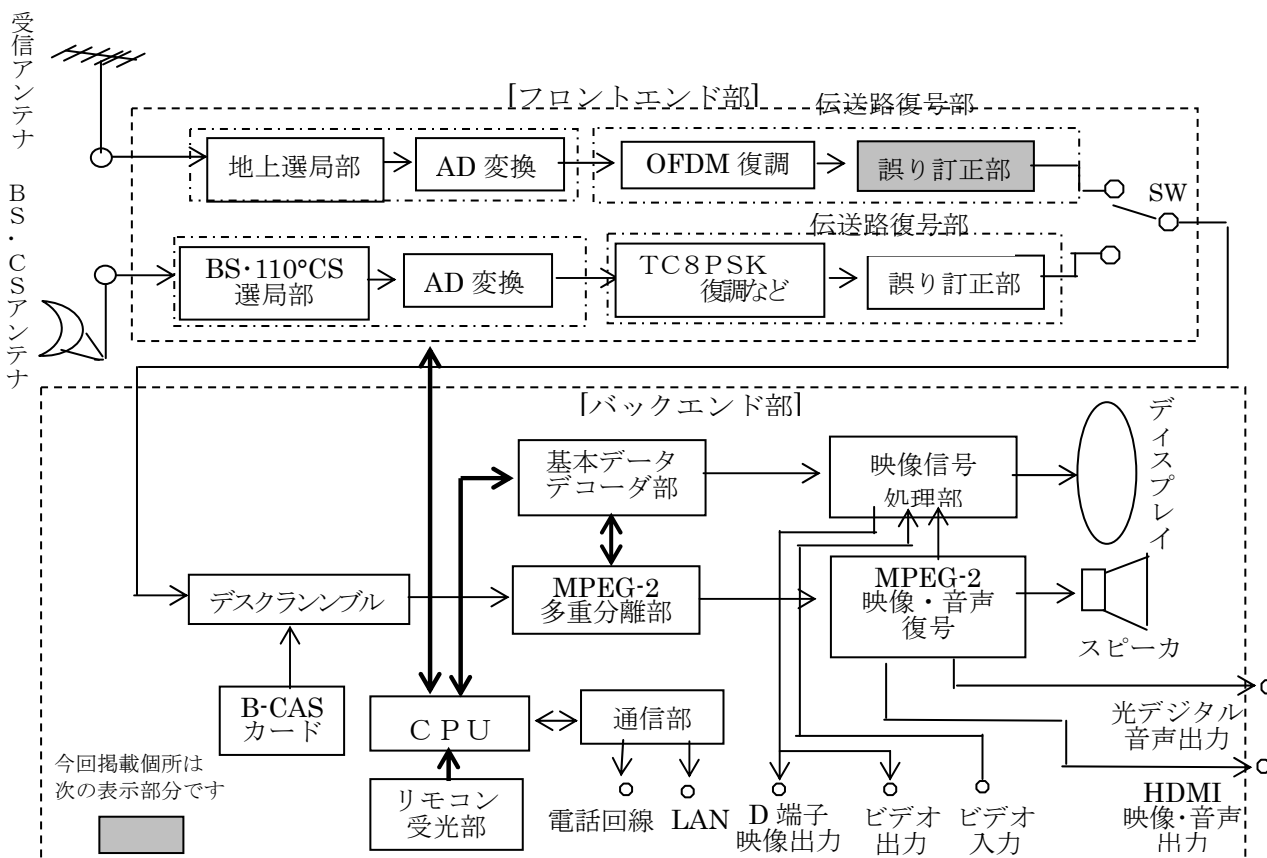


< 地上デジタル放送受信機 (その7・誤り訂正3) >



[参考図] 実際の地上デジタル放送受信機の回路構成図

☆ ブロック符号のわかりやすい事例 (その2)

次に、もう少し複雑なブロック符号による誤り訂正の方式について検討します。

図 10 をご覧ください。

1 ブロック 8 ビットとしましょう。さらに 8 ビットの信号に 4 ビットの CRCC が後についています。

したがって 8 ビットの中に誤りがあれば、横の CRCC(末尾参照)

8 ビットの情報信号	4 ビットの CRCC
8 ビットの情報信号	4 ビットの CRCC
8 ビットの情報信号	4 ビットの CRCC
8 ビットの情報信号	4 ビットの CRCC
たて方向のパリティ	たて方向のパリティ

図 10 ブロック符号の例 (その2)

により誤りの検出が出来ます。図 10 では、このような組が 4 組あり、さらに、この 4 組に対してたて方向のパリティがあります。

たて方向のパリティとは、4 組の情報信号のそれぞれ第 1 番目のビットの和をパリティの第 1 番としてしています。2～8 番目も同様です。

送信等をするときは、これら全部を送信等を行います。したがって 4 組の情報は、それぞれ CRCC で割り切れ、たて方向のシンδροームは各ビットの順番ごとに全て 0 になっています。

この送信等の形は、8 ビットの情報と、それにつづく 4 ビットのパリティで 1 ブロックになっています。言い換えれば、4 つのブロックは、それぞれ誤り検出が可能で、さらに、4 つのブロックにパリティを付け、シンδροームが 0 になっていることです。この方式では、1 ブロックの誤り訂正が可能です。どうして 1 ブロックの誤り訂正が可能なのか原理を説明します。

いま、第 1 ブロックに誤りがあり、第 2、第 3、第 4 ブロックに誤りが無いとしましょう。したがって、第 1 ブロックでは誤りがあったことが CRCC により検出可能です。言い換えれば、誤りブロックの位置が確認できるわけです。

誤りブロックでは、1 ビット誤りのこともあるでしょうし、また、全ビットが誤っているかもしれません。しかし、誤ったビットを含むシンδροームは、必ず 1 になっています。

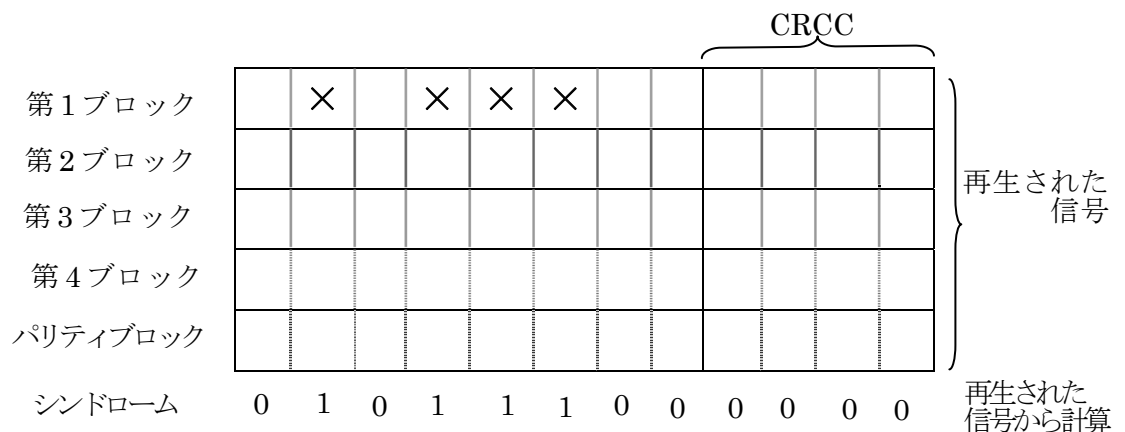


図 11 ダイブロックビット誤りとシンδροーム

図 11 をご覧ください。第 1 ブロックの中に誤りのあったところだけがシンδροームは、1 となっています。訂正するためには、シンδροーム

に表われた

0 1 0 1 1 1 0 0 0 0 0 0

をそのまま第1ブロックにモジロ2で加えればよいのです。

ビット誤りは、0か1に、また、1が0になったのですから、1を加えれば元に戻ります。また、誤りのないビットは、0を加えても関係ありません。

このように第1ブロックの中に1ビット誤りがあったときは、シンδροームの中に1が1個だけとなり、8ビット誤ったときは8個となりますから、一つのブロックの中に誤りが何個あっても、シンδροームを加えることにより訂正できるわけです。CRCCは、シンδροームをどのブロックに加えるかを指示するポインターのような役目をしています。

バースト誤りが発生する系統では、この訂正方式は極めて便利な方法といえるでしょう。

バースト誤りが発生すると全てのビットが誤っているのではなく、各ビットは誤りがあつたりなかつたりするわけです。このような誤りに対して、どのビットが正しいかどのビットが誤りかなど一つ一つの検出ではなく、シンδροームを加えることによって全てOKとなるこの方式は素晴らしい方式といえます。

ただし、この方式は1ブロックのみ訂正が可能です。2ブロック誤りがあるかどうかはCRCCで割り切れないブロックの数が二つあることとなりますから、この時はシンδροームの加算は行いません。したがって訂正は出来ないこととなります。

これまでは、12ビット1ブロックで、4ブロックにそれぞれパリティを付加する方式について説明しましたが、1ブロックが何ビットでも、何ブロックにパリティをつけてもこの原理は同じです。

一般に、1か0に、または、0が1に変化したとき、それは符号が誤りを生じたといいますが、このブロックの考え方では、符号が正しいか誤っているかを各ビットについて判別する必要がありません。シンδροームがなければ1が0になったのか、1が1になったのかもわかりません。このようなとき符号に誤りがあるとか誤りがないとか言うのではなく、

符号が消失（イレージャー）したと言います。シンδροームにより消失した信号が生き返ってくると考えるのです。

このような考え方により、この訂正方式をイレージャー訂正方式と呼んでいます。

☆ ブロック符号のわかりやすい事例（その3）

これまで述べたように、1ブロックのみが誤った場合のブロック訂正（イレージャー訂正）は、理解できたと思いますが、時々、大きなドロップアウト（大きな連続して生ずる誤りのこと。）により、数ブロックにわたるエラーを生ずることがあります。今度は、この場合の訂正について考えて見ましょう。

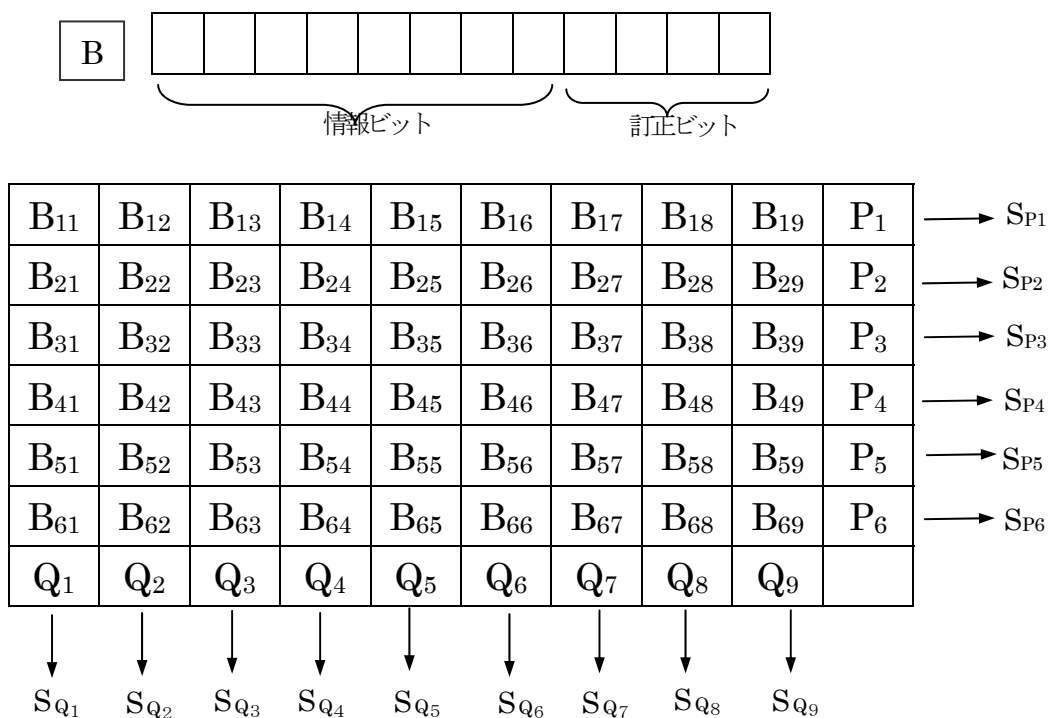


図 12 鎖状符号構成の原理

図 12 をご覧ください。これは鎖状符号構成と呼ばれる方式です。

まず、図 12 の B₁₁ について説明しましょう。B₁₁ は一つのブロックを意味しています。たとえば、8 ビットの情報に 4 ビットの検出信号が付いていれば、B₁₁ は 12 ビットになっています。B₁₁ は何ビットでも良い

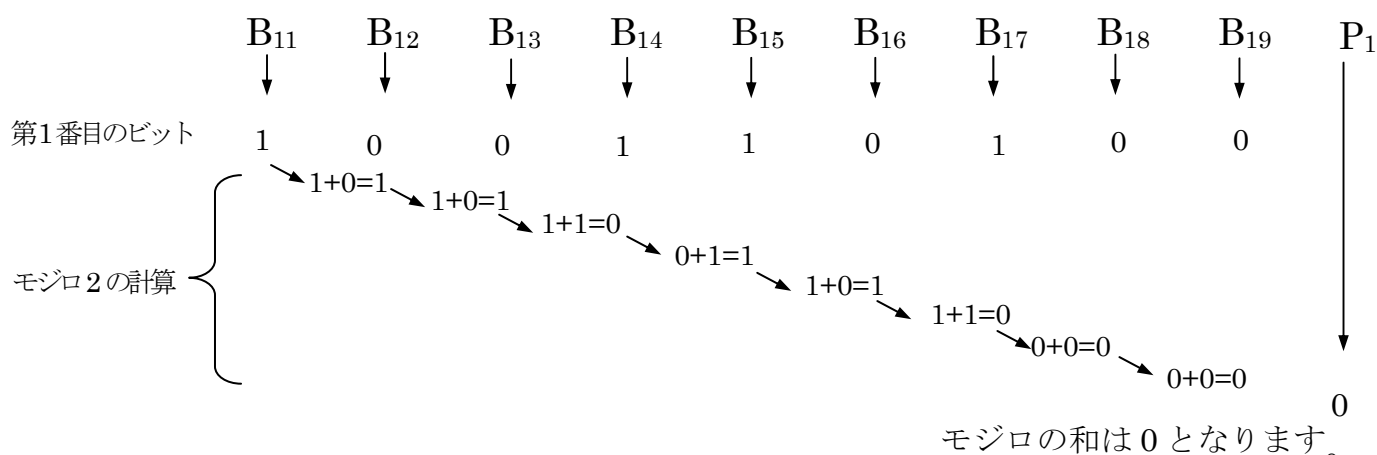
のですが、必ず誤り検出が付いていることが重要です。

$B_{11} \sim B_{69}$ は、全て同じ構成になっています。

話をわかりやすくするため B は 12 ビットで構成されているとして話を進めましょう。 B_{12} は 12 ビットですから、 $B_{12} \sim B_{19}$ も 12 ビットです。しかも、情報ビットは 8 ビット、検出ビットは 4 ビットでブロックごとに誤り検出は可能となっています。

次に、 P_1 の作り方を説明します。 P_1 も 12 ビットです。 P_1 の 1 番目のビットは、 B_{11} 、 B_{12} 、 B_{13} 、 B_{14} 、 B_{15} 、 B_{16} 、 B_{17} 、 B_{18} 、 B_{19} の 1 番目のビットのモジロの和です。

実例にてモジロの和の計算をしてみましょう。各ブロックの第 1 番目のビットが次のようになっていたとします。



P_2 の 2 番目から 12 番目までのビットも同様に作られます。

P_2 から P_6 についても P_1 と同じように計算されて作られています。

つぎに、 Q_1 については、 Q_1 の 1 番目のビットは B_{11} 、 B_{21} 、 B_{31} 、 B_{41} 、 B_{51} 、 B_{61} の 1 番目のビットのモジロの和とします。2 番目から 12 番目についても Q_1 と同じように計算されて作られます。

伝送等をする場合は、 $B_{11} \sim B_{69}$ と共に P_1 から P_6 、 Q_1 から Q_9 も同様に伝送します。ここで聡明な皆さんはお気づきのように、横方向のシンδροーム (S_p) とたて方向のシンδροーム (S_q) が 0 になるように送信等をしているわけです。

では、このような伝送等した信号の誤り訂正について考えましょう。

まず、 $B_{11} \sim B_{69}$ のブロックのうち一つだけブロック誤りがあったと考えます。たとえば B_{11} のブロックが誤ったと考えましょう。各ブロックには誤り検出の 4 ビットの CRCC 符号が付加されていますので、 B_{11} のブロックに誤りがあることはわかります。この時、横方向のシンδροーム S_{P1} とたて方向のシンδροーム S_{Q1} は 12 ビットのうちオール 0 にはならず必ず 1 が含まれます。1 ブロックのみ誤りの場合に S_{Q1} は使用しません。 S_{P1} のみ使用します。すなわち、 S_{P1} を B_{11} にモジロ 2 で加えれば全てのビットを正しく訂正されます。

次に、 B_{11} と B_{12} の 2 つのブロックに誤りが検出されたときを考えます。このとき S_{P1} は動作せず、 S_{Q1} で B_{11} の誤り訂正を行い、 S_{Q2} で B_{12} の誤り訂正を行うのです。

横方向のブロックに誤りが一つだけのときは、 $S_{P1} \sim S_{P6}$ で誤り訂正が可能ですし、また、たて方向のブロックに誤りが一つだけのときは、 $S_{Q1} \sim S_{Q9}$ で誤り訂正を行います。

S_P でも S_Q でも訂正可能なら S_P で訂正を行いますが、その手順は次のようになります。

まず、横方向に誤りブロックが一つなら S_P で訂正しますが、誤りブロックが二つ以上なら S_P は何もしません。また、訂正したブロックは、もう誤りがなかったこととして、次に S_Q に訂正させます。 S_Q では S_P で訂正されなかったブロックを訂正します。この訂正されなかったブロックがたて方向に 1 ブロックだけなら訂正は可能になります。

つぎに、 B_{11} 、 B_{12} と B_{21} の 3 ブロックが誤ったときを考えます。シンδροーム S_{P1} では B_{11} 、 B_{12} との二つのブロックのため訂正はできません。ところが、 B_{12} の誤りは S_{Q2} で訂正できるのです。 B_{12} が正しくなれば S_{P1} によって B_{11} も訂正可能です。また、 B_{21} は S_{P2} によって訂正できますからロジックは複雑になりますが、極めて高い訂正能力をもっていることがわかります。

図 12 の方式で訂正が不可能となるのは、 B_{11} 、 B_{12} 、 B_{21} 、 B_{22} のブロックが同時に誤ったときなどです。すなわち、横方向、たて方向同時に二

つ以上のブロックに誤りがあったときは訂正は不可能になります。

図 12 で構成した符号を実際に伝送等をするため時間軸に沿って並べれば図 13 のようになります。

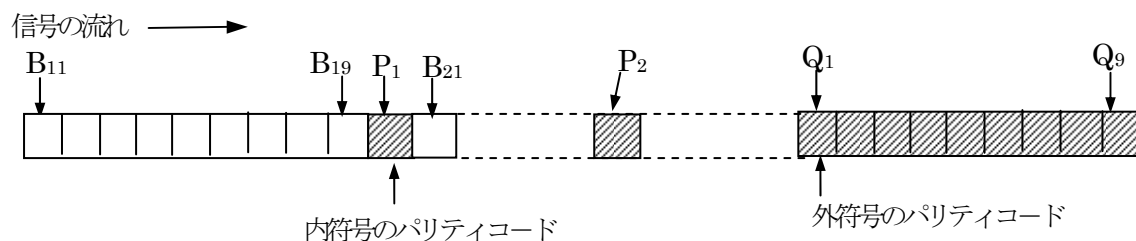


図 13 鎖状符号構成の信号

この図では、情報信号が 9 ブロックあり、そのあと P パリティがあり、それが 6 回繰り返すと Q パリティがあります。この形はちょうど鎖がつながっているように見えますから鎖状符号構成と呼ばれているのです。

鎖状符号構成の P パリティのことを内符号パリティコード、Q パリティのことを外符号パリティコードと呼んでいます。

これまでブロック符号による誤り訂正の概要について説明しましたが、ブロック符号は、隣接符号、リードソロモン符号へと進化します。隣接符号では、T マトリックスを用い、リードソロモン符号では多項式の根を用いて高度な誤り訂正が可能となります。

地上デジタルテレビにおいては、MPEG-2TS を構成する 188 バイト(1 バイト 8 ビット、1504 ビット)の packets 単位にリードソロモン符号化しています。これは 1 ブロックのデータ 204 バイトのうち、情報データが 188 バイトで、その後ろに 16 バイトのチェックビットが付加することにより、RS (204,188) は、最大 8 バイトまでの誤りを訂正することができます。

[もっと知りたい方のために]

CRCC (巡回符号 : Cyclic Redundancy Check Code)

(サイクリック・リダンダンシー・チェック・コード)

送信等行った信号の情報ビットが伝送途中で誤りを発生したかどうかをチェックするための付加符号で、送信等の側では情報ビットにさらに数ビットのCRCCを付加して送ります。CRCCとは、受信等の側でそのブロック符号について定められた符号列の割り算を行って割り切れるようなCRCCを付加して送信するものです。割り切れれば伝送途中での誤りはないと判断できます。割り切れない場合は、誤りはありますので誤り訂正作業を行うこととなります。

パリティチェックによる誤りチェックについてこれまで紹介してきましたが、CRCCを付加する方法は1段と高級な伝送符号列のチェックができます。