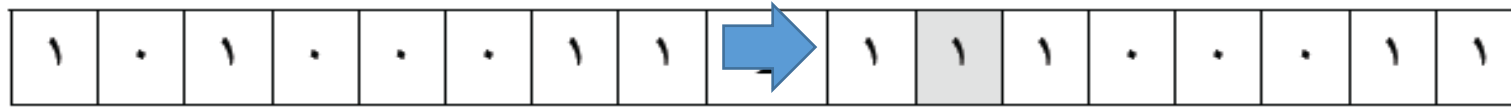


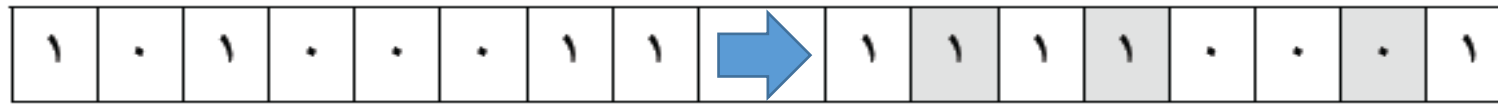
تشخيص و تصحيح خطا

Error Detection and Correction

احتمال رخداد خطا در انتقال بیت داده.
سه نوع خطا در هنگام انتقال داده وجود دارد:
۱ - تک بیت:



۲ - چند بیت:



۳ - انفجاری یا انبوه (بیت‌های متوالی):



افزونگی: جهت تشخیص خطای رخ داده، لازم است همراه داده، داده اضافی نیز ارسال شود که افزونگی گویند.

روش‌های مطرح تشخیص خطا:

1. Parity Check
2. Parity Check (row – column)
3. CRC (Cyclic Redundancy Check)
4. Checksum

Parity Check

برای تشخیص خطای تک بیتی، صد درصد تضمین.
توازن زوج: تعداد "۱" های موجود در داده زوج باشد.
توازن فرد: تعداد "۱" های موجود در داده فرد باشد.
یک بیت به فریم (هر ۸ بیت) داده در فرستنده اضافه شده و در گیرنده مجدد تست می‌شود.

Parity Check



پیغام ZYX	توازن فرد	بیت توازن زوج
۰۰۰	۱	۰
۰۰۱	۰	۱
۰۱۰	۰	۱
۰۱۱	۱	۰
۱۰۰	۰	۱
۱۰۱	۱	۰
۱۱۰	۱	۰
۱۱۱	۰	۱

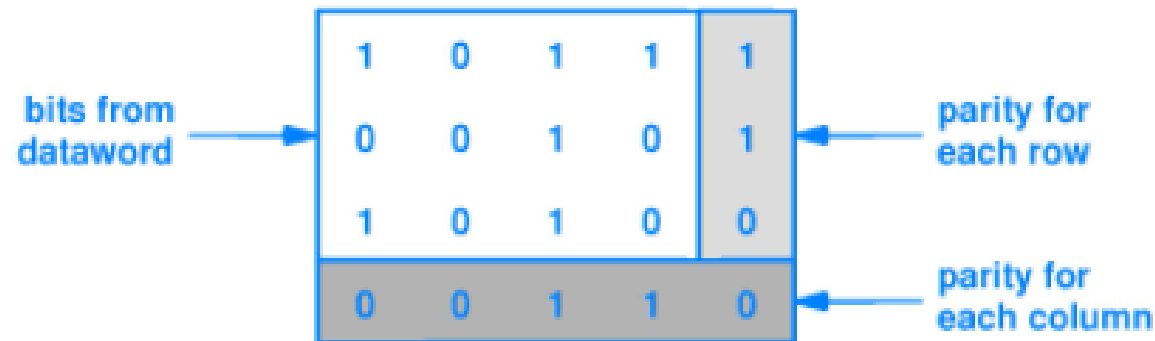
Parity Check (row – column)

بیت توازن فقط به صورت سطری محاسبه نمی‌گردد بلکه ستونی نیز بیت توازن ایجاد می‌شود.

علاوه بر تشخیص خطای یک بیت امکان تصحیح آن نیز وجود دارد.

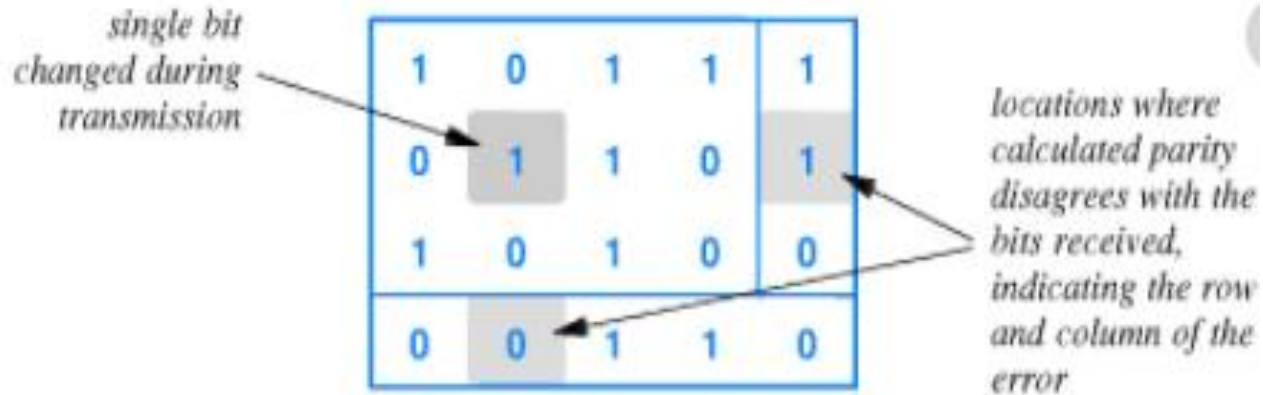
همچنین برای ۲ بیت و ۳ بیت هم، تشخیص خطا دارد اما در مورد خطای ۴ بیت و بالاتر تضمینی برای تشخیص خطا وجود ندارد.

مانند روش قبل دو نوع زوج و فرد دارد.



Parity Check (row – column)

تصحیح (و تشخیص) خطای یک بیت:



برای ۲ بیت تشخیص خطا (سه حالت):

هر دو بیت تغییر یافته در یک سطر.

هر دو بیت تغییر یافته در یک ستون.

دو بیت تغییر یافته در سطر و ستون‌های متفاوت.

Parity Check (row – column)

برای ۲ بیت تشخیص خطا (سه حالت):

هر دو بیت تغییر یافته در یک سطر.

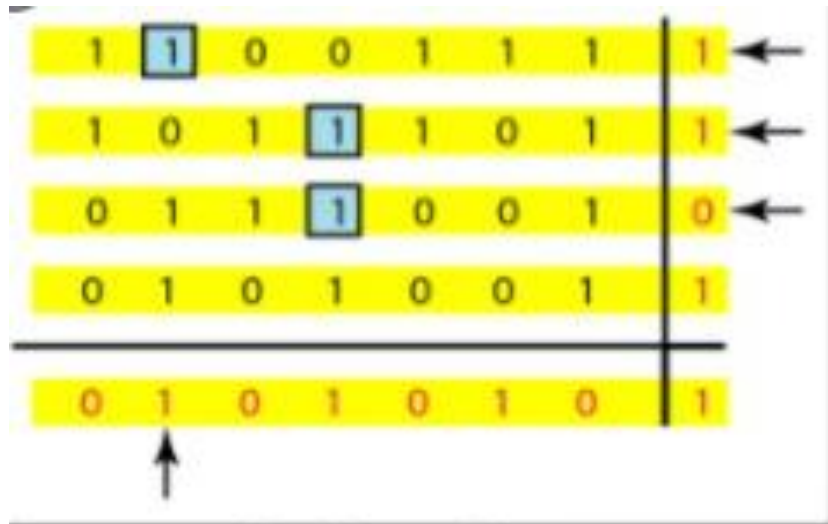
هر دو بیت تغییر یافته در یک ستون.

دو بیت تغییر یافته در سطر و ستون های متفاوت.

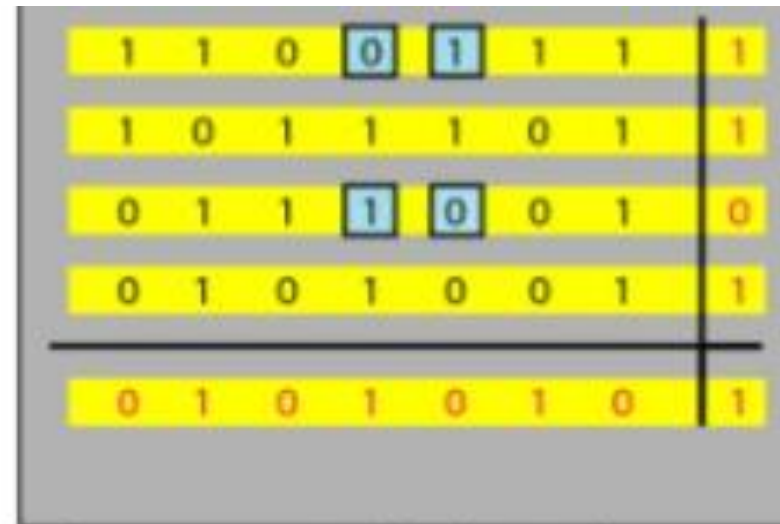
1	1	0	0	1	1	1	1
1	0	1	1	1	0	1	1
0	1	1	1	0	0	1	0
0	1	0	1	0	0	1	1
0	1	0	1	0	1	0	1

Parity Check (row – column)

برای ۳ بیت هم، تشخیص خطا دارد اما در مورد خطای ۴ بیت و بالاتر **تضمینی** برای تشخیص خطا وجود ندارد.



Three errors affect four parities



e. Four errors cannot be detected

CRC (Cyclic Redundancy Check)

Divisor: عدد ثابت بین فرستنده و گیرنده وجود دارد.

مراحل:

1. فرستنده داده ارسالی را به صورت یک عدد بزرگ فرض کرده و به داده به تعداد n بیت صفر اضافه می کند ($n+1$ تعداد بیت Divisor).
2. فرستنده عدد حاصل را بر Divisor تقسیم می کند.
3. فرستنده باقیمانده را به عنوان CRC به داده اضافه و داده حاصل را ارسال می کند.
4. گیرنده کل داده دریافتی را به صورت یک عدد بزرگ فرض کرده و آن را بر Divisor تقسیم کرده که باقیمانده باید صفر شود، اگر صفر نشد خطا رخ داده است.

روش دیگر CRC: Polynomial

مشابه همان روش قبلی است ولی اعداد بوسیله جملات **چند جمله‌ای** نشان داده می‌شود.
مانند:

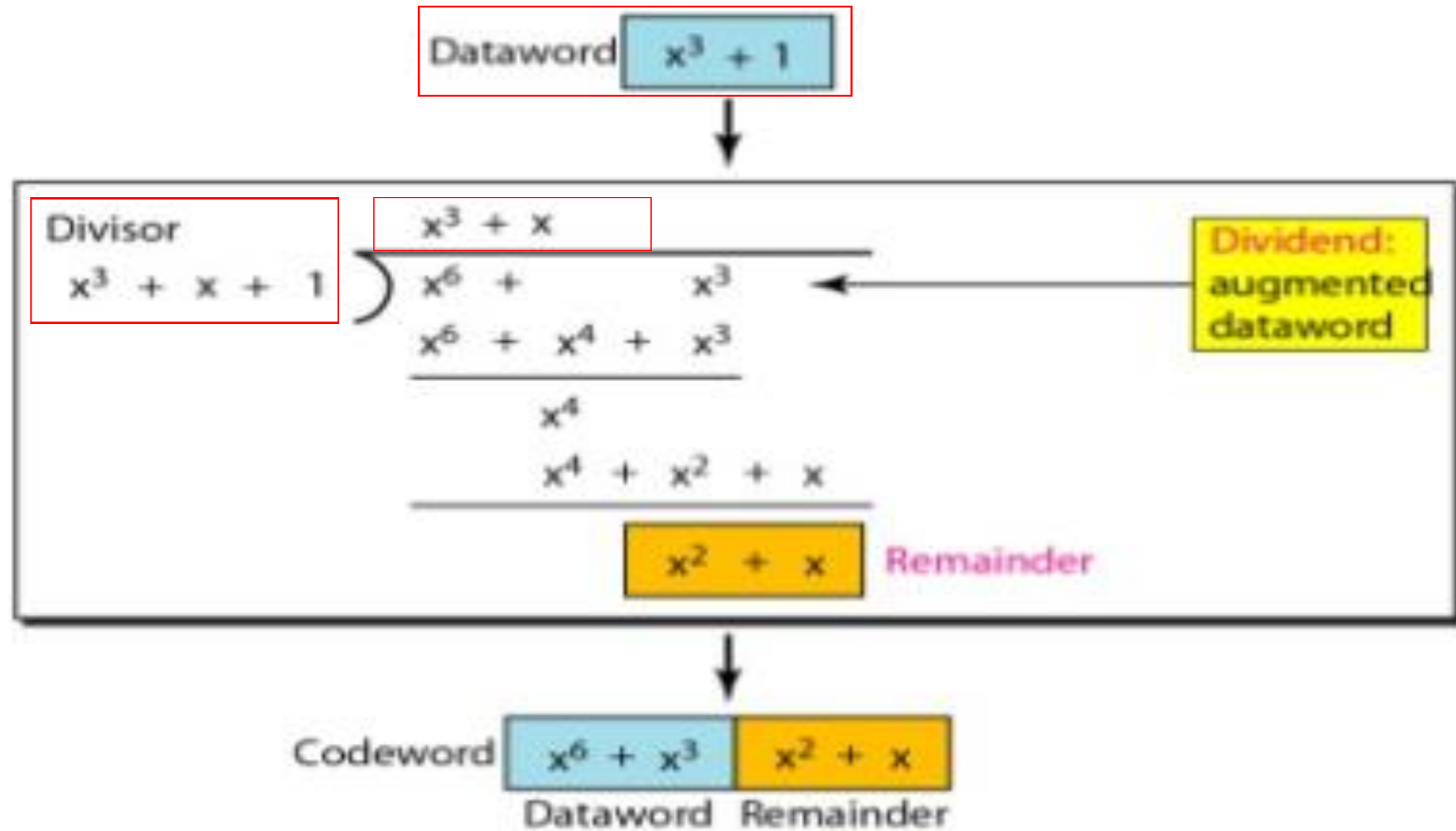
1 0 1 0 0 1 1 1 Divisor	→	$x^7 + x^5 + x^2 + x^1 + 1$ Polynomial
----------------------------	---	---

مشخصات:

- خطاهای یک بیتی را تشخیص می‌دهد.
- خطاهای دو بیتی را در صورتی تشخیص می‌دهد که CRC حداقل ۳ بیتی باشد.
- خطاهای فرد را به شرطی تشخیص می‌دهد که Divisor دارای جمله $x+1$ باشد.
- خطای انبوهی را در صورتی تشخیص می‌دهد که تعداد خطای رخ داده، مساوی و یا کمتر از طول FCS (یا CRC) باشد، یعنی جمله $x^{12} + x^{11} + x^3 + x + 1$ می‌تواند تا ۱۳ بیت خطای انبوه را تشخیص دهد.

روش دیگر CRC: Polynomial

مثال: فرستنده (محاسبات بدون در نظر گرفتن بیت‌های دوبریک میانی).



روش دیگر CRC: Polynomial

دلیل جواب دادن روش (در گیرنده باقیمانده صفر):

1 - $\frac{X^n M(x)}{P(x)} = Q(x) + \frac{R(x)}{P(x)}$ داده اولیه

2 - $T(x) = X^n M(x) + R(x)$ داده ارسالی

3 - $\frac{T(x)}{P(x)} = \frac{X^n M(x) + R(x)}{P(x)}$ معادله کمکی

4 - $\frac{T(x)}{P(x)} = Q(x) + \frac{R(x)}{P(x)} + \frac{R(x)}{P(x)} = Q(x)$ در گیرنده (باقیمانده صفر)

Checksum

در این روش:

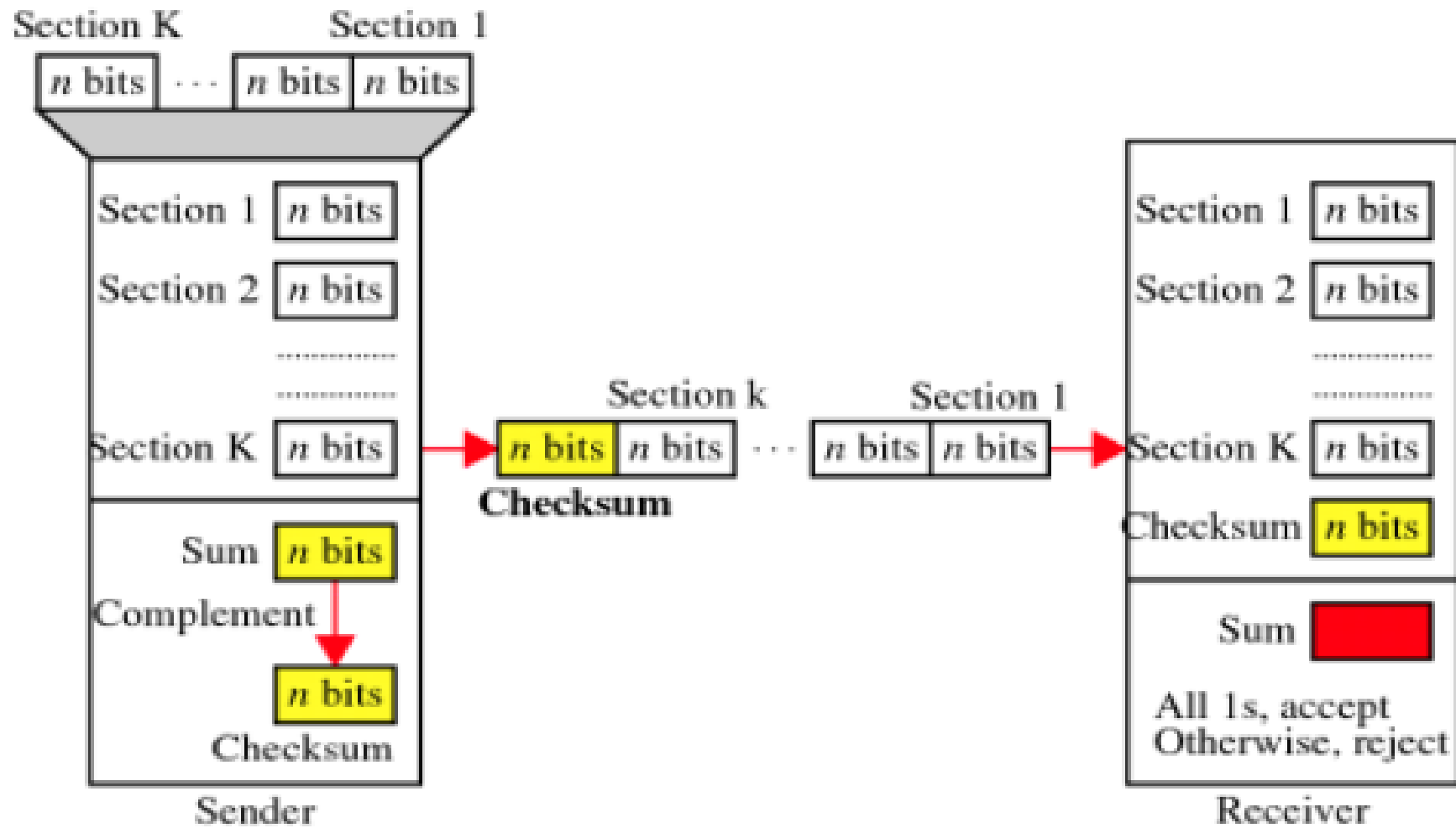
فرستنده:

- داده به بخش‌هایی به طول مساوی تقسیم کرده
- بخش‌ها را با هم جمع کرده که در جمع بیت دوبریک در نظر گرفته نمی‌شود
- عدد حاصل Checksum را مکمل کرده و به انتهای داده اضافه می‌شود
- در نهایت کل داده ارسال می‌گردد.

گیرنده:

- بخش را از هم جدا کرده و با هم جمع می‌شود
- نتیجه با جمع Checksum می‌شود
- تمام بیت‌های حاصل باید یک شود.

Checksum



Checksum

101101	011101	011111
S1	S2	S3

مثال: داده اولیه.

101101 +
011101 +
011111

111001

Complement

000110

101101011101011111000110 داده ارسالی در فرستنده:

عملیات گیرنده:

101101 +
011101 +
011111 +
000110

111111

داده، صحیح منتقل شده.