King Fahd University of Petroleum & Minerals
Computer Engineering Dept

COE 241 – Data and Computer Communications

Term 141

Dr. Ashraf S. Hasan Mahmoud

Rm 22-420

Ext. 1724

Email: ashraf@kfupm.edu.sa

Lecture Contents

1. Asynchronous and Synchronous Transmission
2. Error Detection
3. Error Correction (Optional)

Timing Requirement

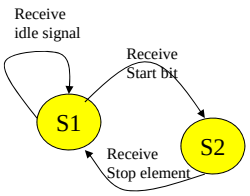
- Reception of digital data requires sampling of received signal at receiver → Sampling time should be known
- Clock drift (example):
  - If a receiver clock drifts by 1% every sample time,
  - Then for Tb = 1μsec, total drift after 50 bit times = 50 X 0.01 = 0.5 μsec
  - Hence, instead of sampling at the middle of the bit time, the receiver will sample at the edge of the bit (I.e. receiver is out-of-synch with transmitter clock)
- For correct reception, receiver clock/carrier should be synchronized with transmitter

Asynchronous Transmission

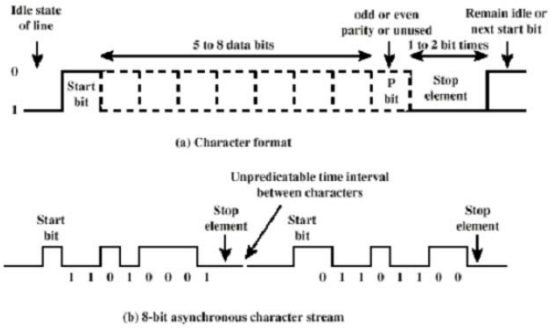
- Exploits: Rx-er can remain for short period in synch with Tx-er
- Used for short stream of bits – data transmitted one character (5 ~ 8 bits) at a time
- Synchronization is needed to be maintained for the length of short transmission
- Character is delimited (start & end) by known signal elements: start bit – stop element
- Rx-er re-synchs with the arrival of new character

Asynchronous Transmission

- Simple / Cheap
- Efficiency: transmit 1 start bit + 8 bit of data +2 stop bits ➔ Efficiency = 8/11 = 72% (or overhead = 3/11 = 28%)
- Good for data with large gaps (e.g. keyboard, etc)



S1: receiver in idle state
S2: receive is receiving character



Example: Problem 6-5

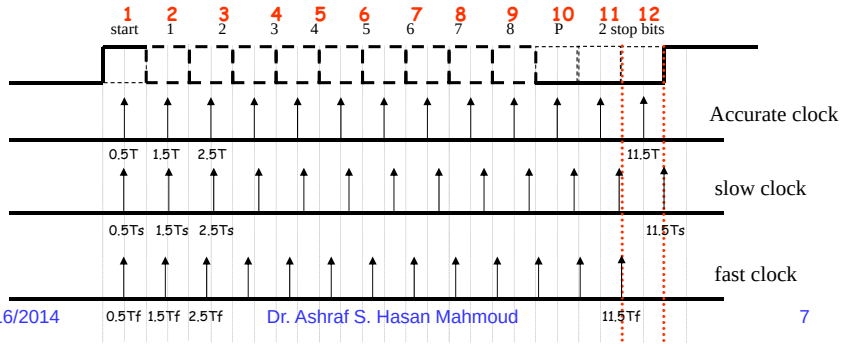
6-5: An asynchronous transmission scheme uses 8 bits, an even parity, and a stop element of length 2 bits. What percentage of clock inaccuracy can be tolerated at the receiver with respect to the framing error? Assume that the bit samples are taken at the middle of the clock period. Also assume that at the beginning of the start bit the clock and incoming bits are in phase.

Example: Problem 6-5- solution

An accurate clock will start will start in phase (middle of first bit) and end end in phase (middle of last bit)

However, a slow clock (time between two consecutive samples increases) will start in phase but will sample the last bit away from the middle of the actual bit duration – for this not to make a mistake it should sample <u>at most</u> at end of the last bit duration

For a fast clock (time between two consecutive samples decreases) will start in phase but will sample the last bit before the middle of the actual bit duration – for this clock not to make a mistake it should sample <u>at least</u> at beginning of the last bit duration



Example: Problem 6-5 - solution (2)

Let the bit duration be T. Then a frame is 12T long.

Let a clock period be T'. The last bit (bit 12) is sampled at 11.5T'.

For a fast running clock, the condition to satisfy is

$$11.5T' > 11T \Rightarrow \frac{T}{T'} < \frac{11.5}{11} = 1.045 \Rightarrow f_{clock} < 1.045 f_{bit}$$

For a slow running clock, the condition to satisfy is

$$11.5T' < 12T \Rightarrow \frac{T}{T'} > \frac{11.5}{12} = 0.958 \Rightarrow f_{clock} > 0.958 f_{bit}$$

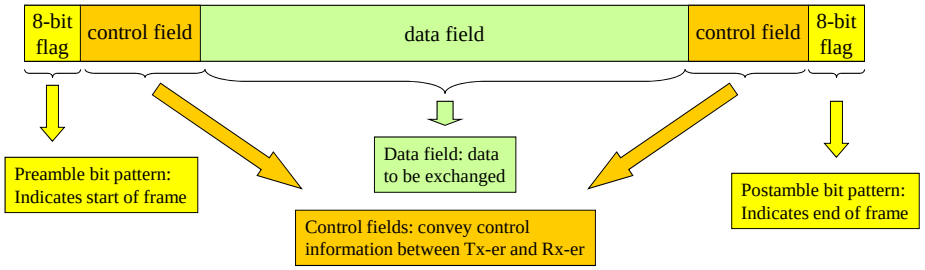
Therefore, the overall condition: 0.958 f_{bit} < f_{clock} <1.045 f_{bit}

Synchronous Transmission

- What if there is a STEADY STREAM of bits between Tx-er and Rx-er
  - Still use the start/stop bits ➔ low efficiency
  - Use synchronous transmission
- Synchronous Techniques:
  - Provide SEPARATE clock signal
    - Expensive and only good for short distances
  - Depend on data encoding to extract clock info
    - E.g. Manchester encoding

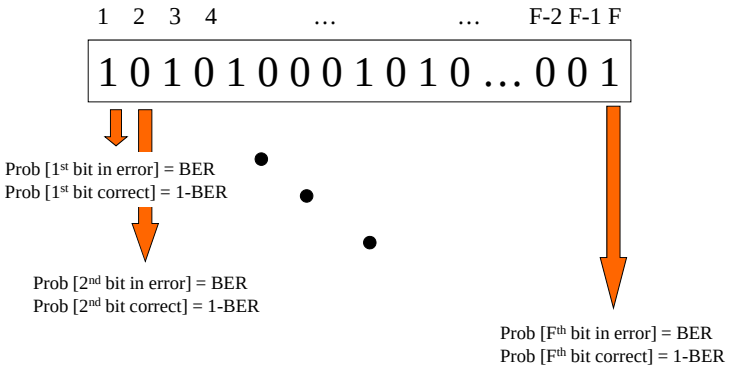
Synchronous Frame Format

- Typical Frame Structure



- For large data blocks, synchronous transmission is far more efficient than asynchronous:
  - E.g. HDLC frame (to be discussed in Chapter 7): 48 bits are used for control, preamble, and postamble – if 1000 bits are used for data ➔ efficiency = 99.4% (or overhead = 0.6%)

Error Detection



Prob [k bits in error in frame] = $\binom{F}{k}(BER)^k(1-BER)^{F-k}$

Error Detection – cont'd

- Hence, for a frame of F bits,

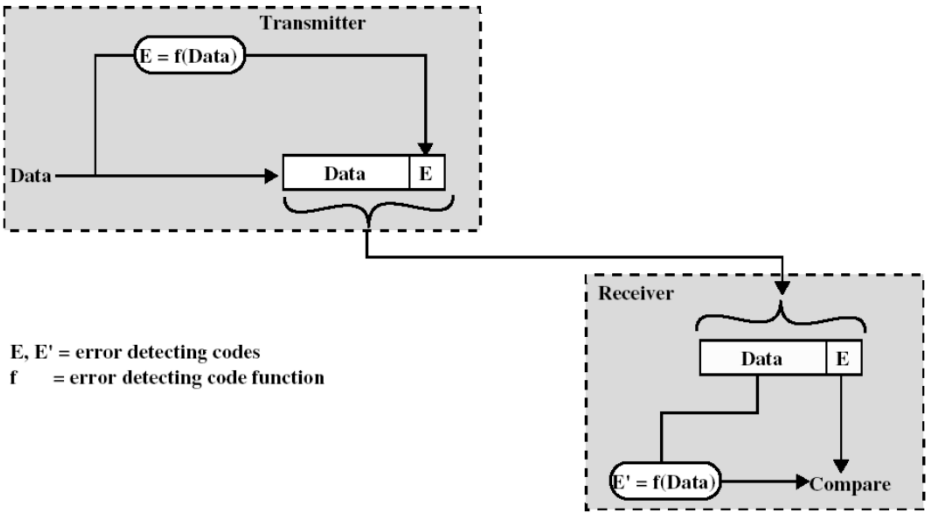
Prob [frame is correct] = Prob [0 bits in error]
= $(1\text{-BER})^F$

Prob [frame is erroneous] = Prob[1 OR MORE bits in error]
= 1 – Prob[0 bits in error]
= $1 - (1\text{-BER})^F$

Or

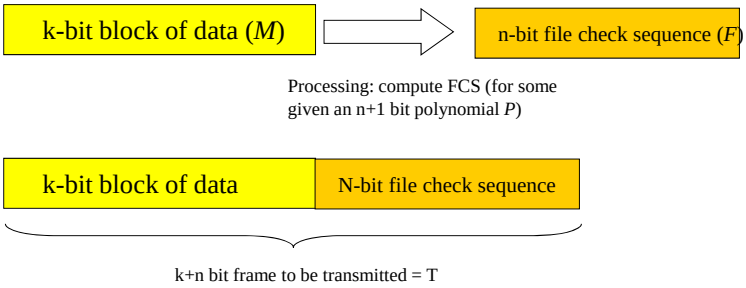
Prob [frame is erroneous] = Prob [1 bit in error] +
Prob[2 bits in error] + … +
Prob[F bits in error]
= 1 – Prob[0 bits in error]
= $1 - (1\text{-BER})^F$

Error Detection (2)



E, E' = error detecting codes
f = error detecting code function

Cyclic Redundancy Check (CRC)



- Modulo 2 arithmetic (like XOR) is used to generate the FCS:
  - 0 ± 0 = 0; 1 ± 0 = 1; 0 ± 1 = 1; 1 ± 1 = 0
  - 1 X 0 = 0; 0 X 1 = 0; 1 X 1 = 1

CRC – Mapping Binary Bits into Polynomials

- Consider the following k-bit word or frame and its polynomial equivalent:

$$b_{k-1}\ b_{k-2}\ \ldots\ b_2\ b_1\ b_0 \rightarrow b_{k-1}x^{k-1} + b_{k-2}x^{k-2} + \ldots + b_1x^1 + b_0$$

where b_i $(k-1 \leq i \leq 0)$ is either 1 or 0

- Example1: an 8 bit word M = 11011001 is represented as $M(x) = x^7+x^6+x^4+x^3+1$

CRC – Mapping Binary Bits into Polynomials - cont'd

- Example2: What is $x^4M(x)$ equal to?

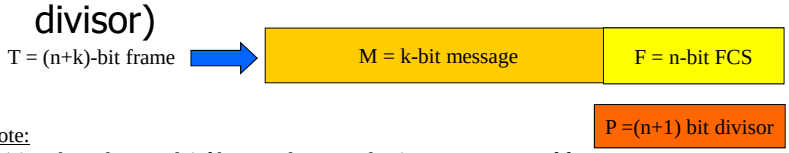
  $x^4M(x) = x^4(x^7+x^6+x^4+x^3+1) = x^{11}+x^{10}+x^8+x^7+x^4$, the equivalent bit pattern is 110110010000 (i.e. four zeros added to the left of the original M pattern)

- Example3: What is $x^4M(x) + (x^3+x+1)$?

  $x^4M(x) + (x^3+x+1) = x^{11}+x^{10}+x^8+x^7+x^4+ x^3+x+1$, the equivalent bit pattern is 110110011011 (i.e. pattern 1011 = x^3+x+1 added to the left of the original M pattern)

CRC Calculation

- T = (k+n)-bit frame to be tx-ed, n < k
- M = k-bit message, the first k bits of frame T
- F = n-bit FCS, the last n bits of frame T
- P = pattern of n+1 bits (a predetermined divisor)



Note:
-T(x) is the polynomial (of k+n-1st degree or less) representation of frame T
-M(x) is the polynomial (of k-1st degree or less) representation of message M
- F(x) is the polynomial (of n-1st degree or less) representation of FCS
- P(x) is the polynomial (of nth degree or less) representation of the divisor P
-$T(x) = X^n M(x) + F(x)$ – refer to example 3 on previous slide

CRC Calculation (2)

- Design: frame T such that it divides the pattern P with no remainder?
- Solution: Since the first component of T, M, is the data part, it is required to find F (or the FCS) such that T divides P with no remainder

Using the polynomial equivalent:
$T(x) = X^nM(x) + F(x)$
One can show that $F(x)$ = remainder of $x^nM(x) / P(x)$
i.e if $x^nM(x) / P(x)$ is equal to $Q(x) + R(x)/P(x)$, then $F(X)$ is set to be equal to $R(X)$.

Note that:

$$\frac{\text{Polynomial of degree k+n}}{\text{Polynomial of degree n}} = \text{polynomial of degree k + remainder polynomial of degree n-1}$$

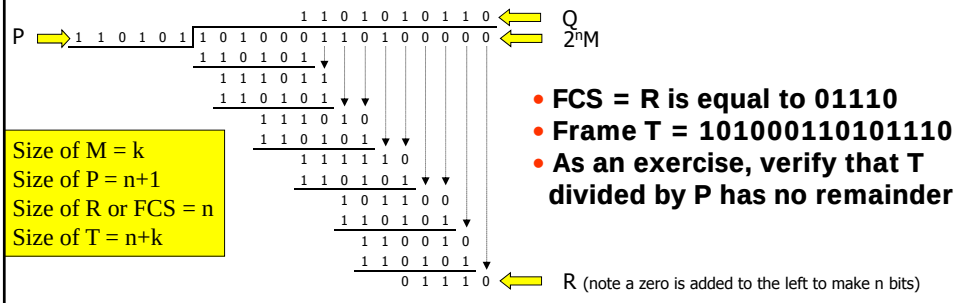
CRC Calculation - Procedure

1. **Shift pattern M n bits to the lift**
2. **Divide the new pattern 2^nM by the pattern P**
3. **The remainder of the division R (n bits) is set to be the FCS**
4. **The desired frame T is 2^nM plus the FCS bits**

Note:
2^nM is the pattern resulting from shifting the pattern M n bits to the left. In other words, the polynomial equivalent of the pattern 2^nM is $x^nM(x)$

CRC Calculation – Example

- **Message M = 1010001101 (10 bits) ➔ k = 10**
  Pattern P = 110101 (6 bits – note 0^{th} and n^{th} bits are 1s)
  ➔n + 1 = 6 ➔ n = 5
  Find the frame T to be transmitted?
- **Solution:**



Size of M = k
Size of P = n+1
Size of R or FCS = n
Size of T = n+k

- **FCS = R is equal to 01110**
- **Frame T = 101000110101110**
- **As an exercise, verify that T divided by P has no remainder**

CRC Calculation – The previous example BUT using Polynomials

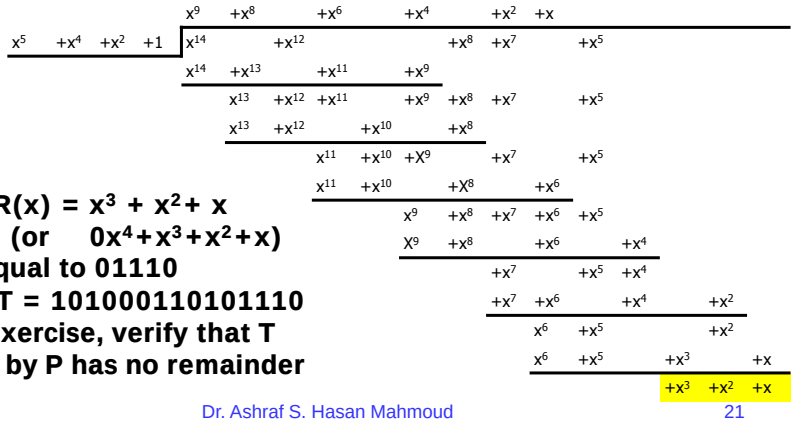
- Message M = 1010001101 (10 bits) ➔ k = 10
- M(x) = $x^9 + x^7 + x^3 + x^2 + 1$ ➔ $x^5M(x) = x^{14} + x^{12} + x^8 + x^7 + x^5$

Pattern P = 110101 (6 bits – note 0th and nth bits are 1s) ➔ n + 1 = 6 ➔ n = 5

- P(x) = $x^5 + x^4 + x^2 + 1$

Find the frame T to be transmitted?

- Solution:



- FCS = R(x) = $x^3 + x^2 + x$
  (or $0x^4+x^3+x^2+x$)
- ➔ R is equal to 01110
- Frame T = 101000110101110
- As an exercise, verify that T divided by P has no remainder

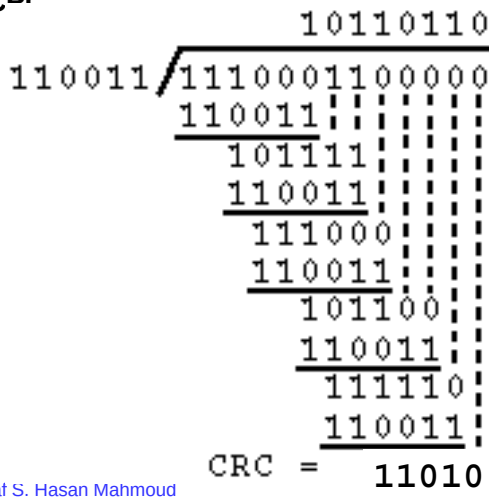
CRC Calculation – The previous example BUT using Polynomials – cont'd

- Message M = 1010001101 (10 bits)
- ➔ M(x) = $x^9 + x^7 + x^3 + x^2 + 1$
- ➔ ➔ $x^5M(x) = x^{14} + x^{12} + x^8 + x^7 + x^5$
- Pattern P = 110101
- ➔ $P(x) = x^5 + x^4 + x^2 + 1$
- $R(x) = x^3 + x^2 + x$
- $Q(x) = x^9+x^8+x^6+x^4+x^2+x$
- $T(X) = x^5M(x) + R(x)$
  $= x^{14} + x^{12} + x^8 + x^7 + x^5 + x^3 + x^2 + x$, or
  T = 1010001101**01110**

- **Exercise**: Verify that $Q(x)\,P(x) + R(x) = x^5\,M(x)$

Example: Problem 6-12

6-12: for P=110011 and M = 11100011, find the CRC



CRC – Receiver Procedure

- **Tx-er transmits frame T**
- **Channel introduces error pattern E**
- **Rx-er receives frame $T_r = T \oplus E$** (note that if E = 000..000, then Tr is equal to T, i.e. error free transmission)
- **T_r is divided by P, Remainder of division is R**
- **if R is ZERO, Rx-er assumes no errors in frame; else Rx-er assumes erroneous frame**
- **If an error occurs and T_r is still divisible by P ➔ UNDETECTABLE error** (this means the E is also divisible by P)

Some Properties

- All single-bit errors, if P(x) has more than one nonzero term
- All double-bit errors, as long as P(x) has a factor with three terms
- Any odd number of errors, as long as P(x) contains a factor (X+1)
- Any burst error for which the length of the burst is less than or equal to n-k
- A fraction of error bursts of length n-k+1
- A fraction of error bursts of length greater than n-k+1

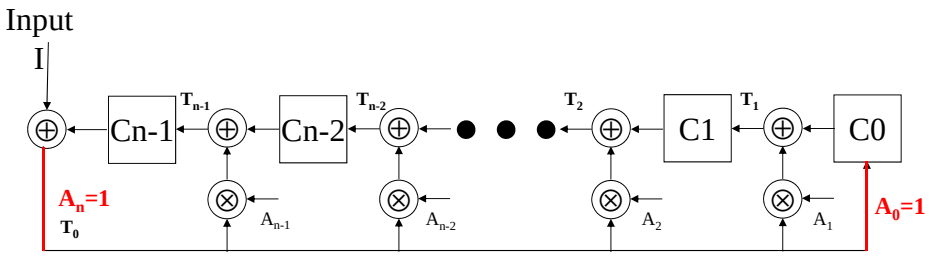
Some Popular CRC Polynomials

- CRC-12: X12+X11+X3+X2+X+1
- CRC-16: X16+X15+X2+1
- CRC-CCITT: X16+X12+X5+1
- CRC-32: X32+X26+X23+X22+X16+X12+X11+X10+X8+X7+X5+ X4+X2+X+1

- CRC-12 – used for transmission of streams of 6-bit characters and generates a 12-bit FCS
- CEC-16 and CRC-CCITT – used for transmission of 8-bit characters in USA and Europe – result in 16-bit FCS
- CRC-32 – used in IEEE802 LAN standards

CRC – Shift Register Implementation – General Divisor

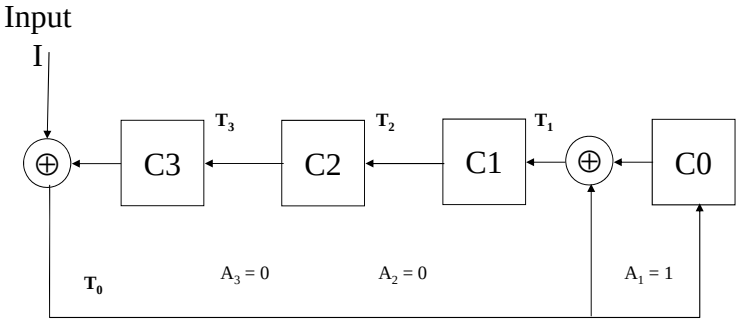
- **Divisor of degree n ➔ no of delay elements = n**
- **Note role of coefficients A0, A1, A2, …, An-1, and An.**



General CRC architecture to implement divisor

$P(X) = X^n + A_{n-1}X^{n-1} + \ldots + A_1X^1 + 1$

CRC – Shift Register Implementation – Example

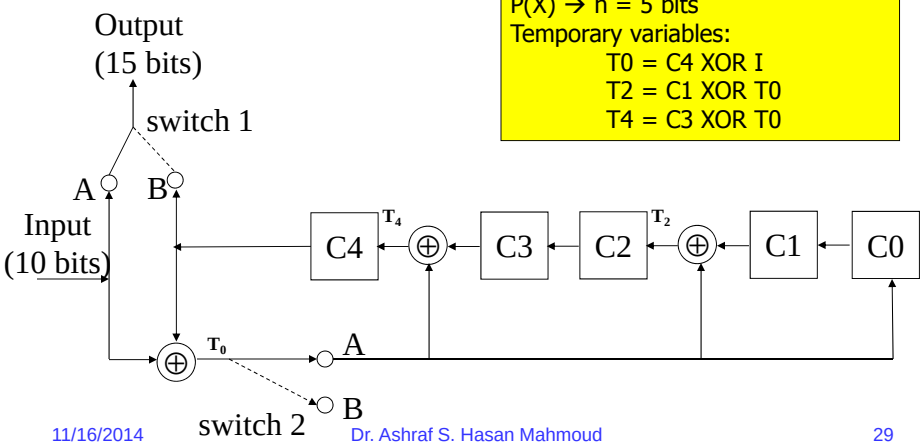


General CRC architecture to implement divisor

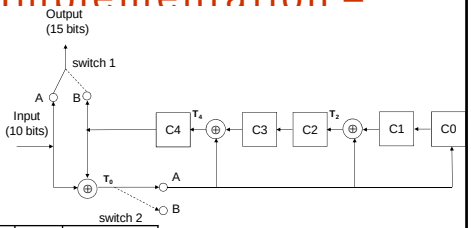
$P(X) = X^4 + X^1 + 1$ (Note that A1 = A4 = 1 and A2 = A3 = 0)

CRC – Shift Register Implementation – Example 6.6

Shift register circuit for dividing by
$P = X^5+X^4+X^2+1$
I = [1010001101] → k = 10 bits
P(X) → n = 5 bits
Temporary variables:
T0 = C4 XOR I
T2 = C1 XOR T0
T4 = C3 XOR T0



CRC – Shift Register Implementation – Example 6.6 – cont'd



	C4	C3	C2	C1	C0	T4	T2	T0	I
Initial	0	0	0	0	0	1	1	1	1
step 1	1	0	1	0	1	1	1	1	0
step 2	1	1	1	1	1	1	1	0	1
step 3	1	1	1	1	0	0	0	1	0
step 4	0	1	0	0	1	1	0	0	0
step 5	1	0	0	1	0	1	0	1	0
step 6	1	0	0	0	1	0	0	0	1
step 7	0	0	0	1	0	1	0	1	1
step 8	1	0	0	0	1	1	1	1	0
step 9	1	0	1	1	1	0	1	0	1
step 10	0	1	1	1	0				

MSB

message to
be sent (k = 10 bits)

R

Cyclic Redundancy Check (CRC)

- Animation for CRC Calculation

Example: Problem 6-13

A CRC is constructed to generate a 4-bit FCS for an 11-bit message. The generator polynomial is X^4+X^3+1

a) Draw the shift register circuit that would perform this task

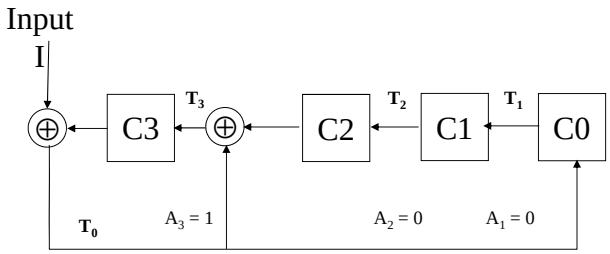
b) Encode the data bit sequence 10011011100 (leftmost bit is the LSB) using the generator polynomial and give the code word

c) Now assume that bit 7 (counting from the LSB) in the code word is in error and show that the detection algorithm detects the error

Example: Problem 6-13 - solution

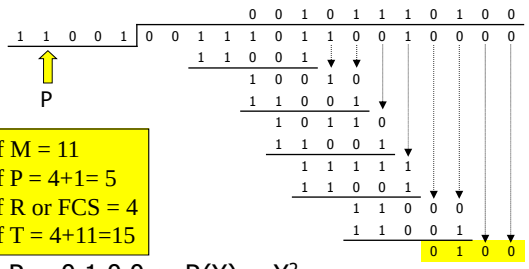
a)

$$P(X) = X^4 + X^3 + 1$$



Example: Problem 6-13 - solution

b)



Size of M = 11
Size of P = 4+1= 5
Size of R or FCS = 4
Size of T = 4+11=15

$M(X) = X^8 + X^7 + X^6 + X^4 + X^3 + 1$
$P(X) = X^4 + X^3 + 1$

➔ R = 0 1 0 0 or $R(X) = X^2$

Transmitted Frame T = 001110110010100

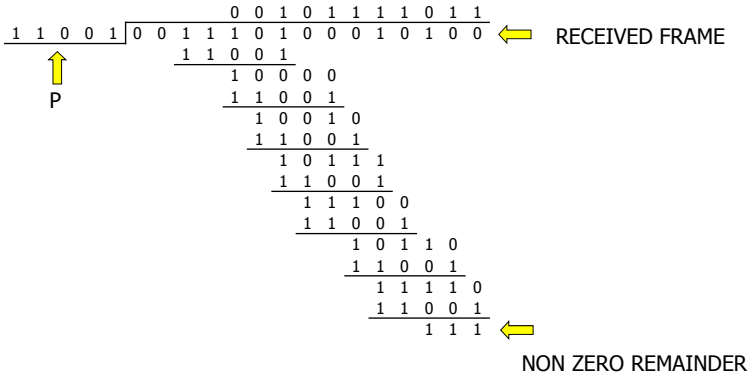
$T(X) = X^4M(X) + R(X) = X^{12}+X^{11}+X^{10}+X^8+X^7+X^4+X^2$

Notes:

1. $X^4M(X)/P(X) = Q(X) + R(X)/P(X)$, where $Q(X) = X^8 + X^6 + X^5 + X^4 + X^2$ (as seen from the long division process)
2. One can verify that P(X) Q(X) + R(X) is indeed equal to $X^4M(X)$ {note that for the addition of polynomial terms modulo-2 applies; i.e. $X^9+X^9 = 0$}

Example: Problem 6-13 - solution

c) Received frame (LSB from the left) = 0 0 1 0 1 0 0 0 1 0 1 1 1 0 0
dividing by P yields a nonzero remainder ➔ error is detected
Remainder = 0111



Textbook Problems of INTEREST

- Texbook Problems list: