

### 3. THE DATA LINK LAYER

#### 3.1. Data link layer design issues.

- 3.1.1. Services provided to the network layer
    - Unacknowledged connectionless service
    - Acknowledged connectionless service
    - Acknowledged connection-oriented service
  - 3.1.2. Framing
    - Character count
    - Starting and ending characters
      - character stuffing (kunnen uitvoeren)
    - Starting and ending flags
      - bit stuffing (kunnen uitvoeren)
    - Physical layer coding violations
  - 3.1.3. Error control
  - 3.1.4. Flow control
- Weten wat de beide begrippen inhouden

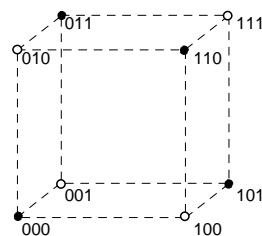
#### 3.2. Error detection and correction

- 3.2.1. Error-correcting codes
  - Hamming distance (kunnen bepalen van gegeven codewoorden)

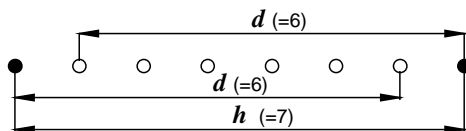
##### Voorbeeld Hammingafstand

Driebits code met Hammingafstand 2

- Gebruikte code
- Niet gebruikte code



##### Hammingafstand nodig voor detectie

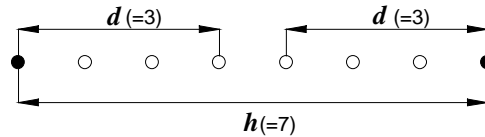


Ontdekken mogelijk indien:  $h \geq d + 1$

of  $d \leq h - 1$

$h$  = Hammingafstand [bits]  
 $d$  = aantal foute bits

##### Hammingafstand nodig voor correctie



Herstel mogelijk indien:  $h \geq 2d + 1$

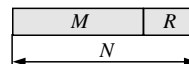
$h$  = Hammingafstand [bits]  
 $d$  = aantal foute bits

of  $d \leq \left\lfloor \frac{h-1}{2} \right\rfloor$

- het aantal nodige redundante bits voor correctie van een fout van één bit kunnen bepalen afhankelijk van de lengte van de data.

##### E én bit kunnen hers tellen

Hoeveel bits nodig?



$M$  = databits  
 $R$  = bits voor foutcontrole

- De databits hebben  $2^M$  verschillende bitpatronen
- Per bitpatroon N patronen door een fout van één bit

$$2^M + 2^M \cdot N \leq 2^N$$

Oorspronkelijk aantal bitpatronen      Aantal foute bitpatronen      Totaal aantal bitpatronen

Met  $N = M + R$  volgt  $M + R + 1 \leq 2^R$  ( $R$  bepalen door proberen)

##### Aantal extra bits nodig

(Herstel van één fout bit)

M	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
+															
R	2	3	3	3	4	4	4	4	4	4	4	5	5	5	5
=															
N	3	5	6	7	9	10	11	12	13	14	15	17	18	19	20

$M$  = netto aantal bits van het bericht (databits)  
 $R$  = aantal benodigde controlebits  
 $N$  = totaal aantal bits (machten van 2 ontbreken!)

### Hammingcodering

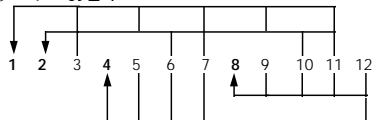
- Plaats pariteits bits op plaatsnummers die machten van 2 zijn (1, 2, 4, 8, 16...)  
**P** **P** 3 **P** 5 6 7 **P** 9 10 11 12
- Plaats nrs. van de databits ontbinden in som van machten van 2 (Vb. 11=1 + 2+ 8)
- Een pariteits bit telt die databits waarbij zijn plaats nr. in de ontbinding voorkomt  
 1 telt 3, 5, 7, 9, 11, ...  
 2 telt 3, 6, 7, 10, 11, ...  
 4 telt 5, 6, 7, 12, ...  
 8 telt 9,10,11,12, ...

a464 - Datacommunicatie en netwerken

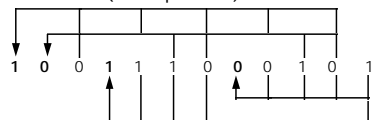
10

### Hammingcodering (voorbeeld)

$M = 8 \Rightarrow R \geq 4$



Databericht 0 1 1 0 0 1 0 1 (even pariteit):



a464 - Datacommunicatie en netwerken

11

- Hamming code  
 Bij een gegeven databericht het gecodeerde bericht kunnen construeren

#### 3.2.2. Error detecting codes

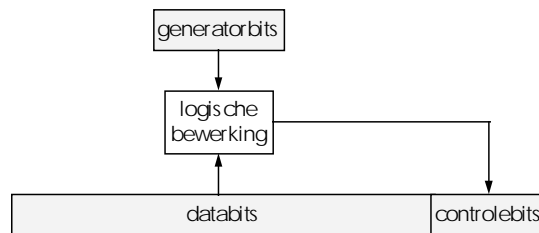
Fout corrigerende codes zijn aantrekkelijk als de kans op fouten groot is zoals bij (wel of niet mobiele) radioverbindingen, ze worden daar veel toegepast.

- Afweging kunnen maken voor een bepaald geval tussen foutdetectie en foutcorrectie.
- parity bit ( even, odd)
- CRC
  - Indien een bericht en een generatorveelterm gegeven zijn moet een verstuurd bericht, dus met CRC-bits, kunnen worden berekend.
  - Je moet kunnen aantonen dat een foutenpatroon dat overeenkomt met dat van de generatorveelterm, of een combinatie daarvan, niet gedetecteerd kan worden.

-

### Cyclic Redundancy Check (CRC)

Uit een afgesproken bitrij, de generator, en de databits worden de controlebits berekend



a464 - Datacommunicatie en netwerken

13

### Bitreeks voorges teld door een veelterm

$$a_{k-1}x^{k-1} + a_{k-2}x^{k-2} \dots a_2x^2 + a_1x + a_0$$

Machten van  $x$  geven het plaats nummer aan; de bijbehoren coëfficiënten  $a_i$  zijn 1 of 0.

Vbld:

1	1	0	0	0	1
$x^5$	$+ x^4$				$+ x^0$

1	1	0	1	0	0	1	0
$x^7$	$+ x^6$		$+ x^4$			$+ x$	

a464 - Datacommunicatie en netwerken

14

### Nullen toevoegen aan bitrij

Een bitrij voorges teld door polynoom  $M(x)$ .

Die bitrij met  $r$  nullen erachter is dan  $x^r M(x)$

Voorbeeld:

$$x^5 + x^4 + x + 1 \triangleq 110011$$

$$x^4(x^5 + x^4 + x + 1) =$$

$$x^9 + x^8 + x^5 + x^4 \triangleq 1100110000$$

a464 - Datacommunicatie en netwerken

15

### Algoritme CRC

- Afgesproken generatorveelterm  $G(x)$  met graad  $r$
- Bericht  $M(x)$
- Maak  $x^r M(x)$
- Deel dit door  $G(x)$
- Tel de rest van de deling,  $R(x)$ , op bij  $x^r M(x) \Rightarrow$
- Verstuur  $x^r M(x) + R(x)$  (Dit is deelbaar door  $G(x)$ )
- Bij de ontvanger wordt  $x^r M(x) + R(x)$  gedeeld door  $G(x)$
- Is de rest 0 dan is het bericht goed
- Is de rest *niet* 0 dan is het bericht beschadigd

a464 - Datacommunicatie en netwerken

1

Algoritme CRC (2)

$$\frac{x^r M(x)}{G(x)} = D(x) + \frac{R(x)}{G(x)}$$

$$x^r M(x) = D(x).G(x) + R(x)$$

$$x^r M(x) \pm R(x) = D(x).G(x) + \cancel{R(x)} \pm R(x)$$

$\xrightarrow{\text{versturen; is deelbaar door } G(x)}$        $\xrightarrow{\text{is 0 bij mod 2 optelling}}$

a464 - Datacommunicatie en netwerken

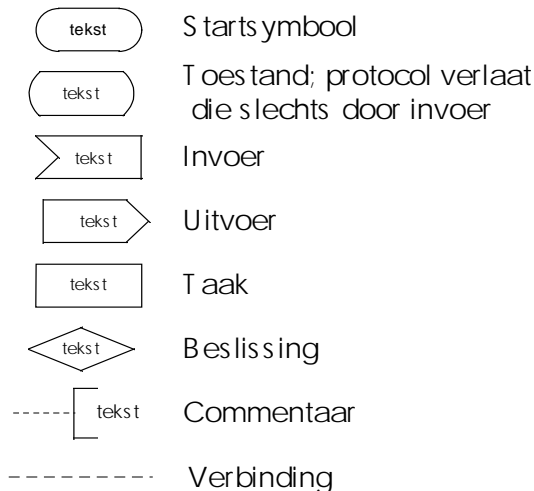
17

3.3. Elementary data link protocols

In het boek worden de verschillende protocollen geïllustreerd aan de hand van programma's. In plaats daarvan zijn hier SDL-diagrammen van de betreffende protocollen gegeven. De programma's behoren dus niet tot de tentamenstof.

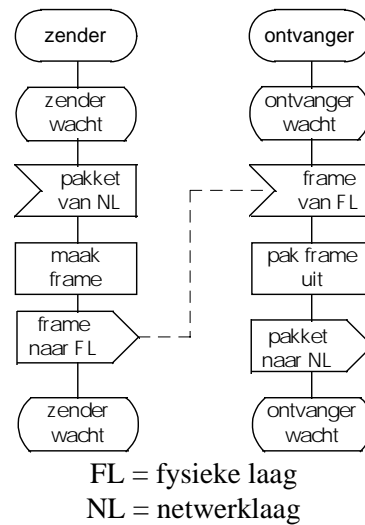
SDL, (Specification and Description Language) is een taal om protocollen te beschrijven. Ze is genormaliseerd door de ITU-T. De taal is gebaseerd op eindige automaten. Er bestaat een programmavorm en een grafische vorm. De grafische vorm is in feite een aangekleed toestandsdiagram (transitiediagram).

Hieronder worden de bij dit college gebruikte symbolen van de grafische vorm gegeven.



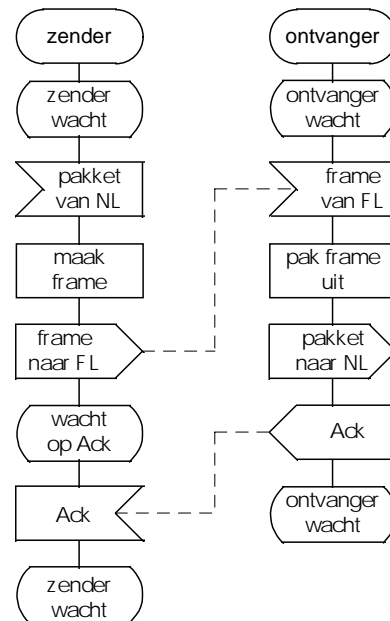
In een protocolbeschrijving van SDL komt vaak een toestand meer dan één keer voor. In dat geval is het dezelfde toestand. Op deze manier wordt het diagram eenvoudiger te tekenen.

3.3.1. An unrestricted simplex protocol Protocol 1, "Utopia".



Het protocol is niet geschikt voor een kanaal met fouten.  
Geen stroombeheersing mogelijk

3.3.2. A simplex stop-and-wait protocol Protocol 2, "stop-and-wait"



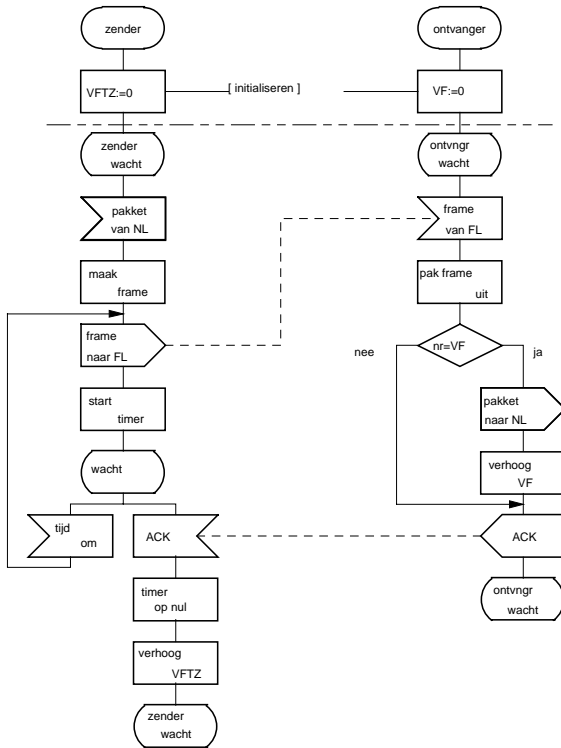
Het protocol is niet geschikt voor een kanaal met fouten.

Wel stroombeheersing.

- ga na waarom het protocol niet werkt als het slechts van een timer voor heruitzendingen voorzien wordt
- bovenstaand diagram kunnen tekenen

3.3.3. A simplex protocol for a noisy channel

Protocol 3, "PAR"



VFTZ= nummer van Volgend Te Zenden Frame

VF = nummer van Verwacht Frame

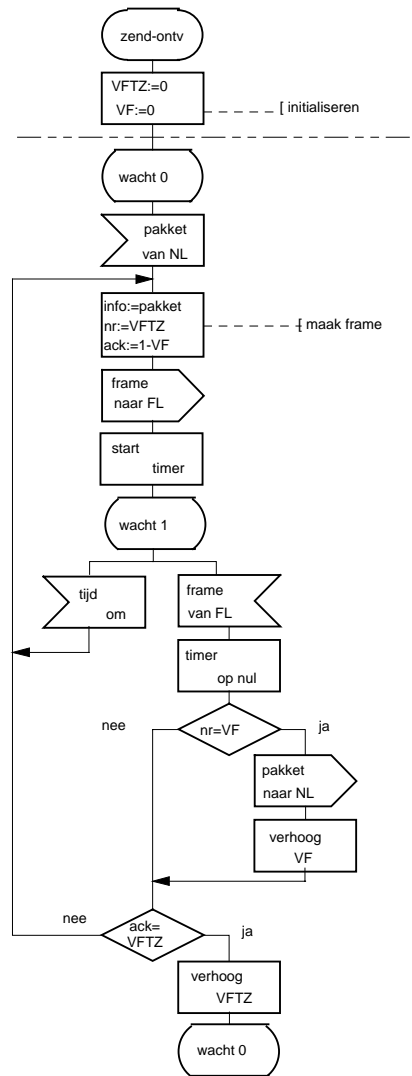
- Bovenstaand diagram kunnen tekenen
- Behoort tot de stop-and-wait-protocollen
- Deze variant wordt ook wel ARQ genoemd
- Het protocol is geschikt voor een kanaal met fouten en maakt stroombeheersing mogelijk.

3.4. Sliding window protocols

- piggybacking
- sliding window
- sending window
- receiving window

3.4.1. A one bit sliding window protocol

Protocol 4 (sliding window)



- bovenstaand diagram kunnen tekenen (en het protocol natuurlijk begrijpen).

3.4.2. A protocol using go back n

- pipelining
- go back n
- selective repeat

3.4.3. A protocol using selective repeat

Het behandelde in deze sub-paragraaf behoort niet tot de tentamenstof. Men moet wel het principe van selective repeat kennen.

### 3.5. Protocol specification and verification

Behoort niet tot de tentamenstof.

### 3.6. Example data link protocols

#### 3.6.1. HDLC - High-level Data Link Control

- frame-indeling van HDLC (fig. 3-24) uit het hoofd kennen.
- functies kennen van
  - Information frame:  
Is voor verbindingsgericht transport van de data.
  - Supervisory frame:  
Is voor begeleidende berichten bij het datatransport, het dataveld is leeg.
  - Unnumbered frame:  
Is voor het voor beheer van de verbinding en voor verbindingsloos datatransport (dus zonder frame-nummers)
- Indeling van de besturingsvelden (fig. 3-25) behoort niet tot de stof.

#### 3.6.2. The data link layer in the Internet

Tot aan blz. 240 2 regels van onder

- PPP
  - LCP
  - NCP
  - verschil tussen PPP en HDLC-frame weten

### 3.7. Summary

Doorlezen