



5. Übung zur Vorlesung Rechnernetze

Abgabe: 17.12.2004

Aufgabe 1: HDLC (5 Punkte, 1 + 2 + 2)

- Welche Rahmenarten gibt es in HDLC und wofür werden sie verwendet?
- Untersuchen Sie die beiden folgenden HDLC-Pakete! Identifizieren alle einzelnen Bestandteile des HDLC-Rahmens, und geben Sie für jeden Teil seine genaue Bedeutung an! Um welche Art von Rahmen handelt es sich jeweils? (Hinweis: Gehen Sie davon aus, dass das Adressfeld 1 Byte lang ist und die CRC-Checksumme mittels eines Generatorpolynoms vom Grad 16 ermittelt wurde).
 - 7E 83 71 74 82 7E
 - 7E 83 00 A1 B3 82 4C 9D 6F E4 11 01 02 45 6B 71 80 7E
- HDLC verwendet ein Schiebefensterprotokoll zur Flusskontrolle mit der maximalen Fenstergröße 7. Nehmen Sie an, Stationen A und B seien über einen Kanal mit einer Kapazität von 100 Mbps verbunden. Die beiden Stationen seien 10 km voneinander entfernt. Die Ausbreitungsgeschwindigkeit des Signals betrage 200.000 km/s. Es wird von einer Rahmengröße von 1000 Bit inklusive Header ausgegangen. Jedes Paket wird vom Empfänger bestätigt. Gehen Sie ferner davon aus, dass die Länge der Bestätigungen zu vernachlässigen ist. Reicht die maximale Fenstergröße 7 bereits aus, um eine Effizienz von 100% zu erreichen, wenn keine Übertragungsfehler auftreten? Ab welcher Fenstergröße wird eine Effizienz von 100% erreicht, wenn keine Übertragungsfehler auftreten? *Hinweis: Verwenden sie die Formeln zur Berechnung der Effizienz aus der vergangenen Übungsstunde (siehe <http://www-vs.informatik.uni-ulm.de/teach/ws04/rn1/Fehlerkontrolle.pdf>). Auf den Folien wird der senderseitige Timeout auf $2 * \text{Signallaufzeit}$ gesetzt. In der Praxis wird meist ein längerer Timeout verwendet.*

Aufgabe 2: Programmieraufgabe File-Transfer (5 Punkte)

Wir wollen unser Programm vom letzten Übungsblatt erneut etwas verbessern. Laden Sie sich zunächst den Java-Code aus folgendem Archiv herunter: http://www-vs.informatik.uni-ulm.de/teach/ws04/rn1/OSI_Aufgabe3.zip. (Achtung: Wir haben die Implementierung der Layer4 Klasse etwas verändert, sodass nun auch Bitfehler bei der Übertragung simuliert werden).

Erweitern Sie das Rahmenformat der Layer7-Pakete vom letzten Übungsblatt um eine 16-Bit lange CRC Checksumme am Ende des Pakets. Die CRC Checksumme soll mittels des folgenden Generator-Polynoms erzeugt werden: $x^{16} + x^{12} + x + 1$. Die Pakete sollen weiterhin eine Maximallänge von insgesamt 100 Byte haben. Abzüglich des 2-Byte langen Headers und der 2 Byte langen Checksumme kann ein Paket also maximal 96 Datenbytes enthalten. Entdeckt der Empfänger einen Übertragungsfehler bei einem Daten-, Start_of_file-, oder End_of_file-Paket, so soll er sofort ein explizites Negative_Acknowledge (Pakettyp 5) mit der Sequenznummer des fehlerhaften Paketes an den Sender schicken, welcher das fehlerhafte Paket daraufhin sofort wiederholt anstatt den Timeout für das Acknowledge abzuwarten. Wird ein Übertragungsfehler bei einem Acknowledge oder Negative_Acknowledge entdeckt, so soll das fehlerhafte Paket einfach ignoriert werden, (was zum Ablauf des Timeouts führt). Die Klasse Layer7 soll jedes Mal, wenn sie einen Übertragungsfehler entdeckt hat, eine entsprechende Meldung auf dem Bildschirm ausgeben, mit der Sequenznummer und dem Typ des fehlerhaften Pakets. Auf Senderseite soll außerdem bei jedem Erhalt eines Negative_Acknowledge eine Meldung mit der Sequenznummer des fehlerhaften Pakets auf dem Bildschirm ausgegeben werden.

Geben Sie bitte den kompletten Quelltext ausgedruckt oder per Email ab! Schicken Sie den gesamten Quelltext gezippt an: schorr@informatik.uni-ulm.de mit dem Betreff „RN1 – Übung 5“.