EI Info3 2010-11	*INFORMATIK* Stunde vom 20.09.2010	$(11)_2$

Wir haben in dieser Stunde über Notengebung, Stoffinhalte usw. gesprochen. Außerdem haben wir begonnen, zu klären, wie eigentlich Daten im PC codiert werden.

Tafelbild
Ein echtes Tafelbild gibt es diesmal noch nicht, hier aber die wichtigsten Inhalte der Doppelstunde:

Angenommen, wir wollten eine Nachricht übermitteln, ohne direkt zu sprechen. Ich möchte dir beispielsweise deine Note mitteilen. Dich interessiert aber erst einmal, ob du unter 5 Punkten stehst oder nicht, da du Info sowieso nicht abrechnen möchtest. Dann wäre eine einfache Möglichkeit, eine Taschenlampe zu nehmen und vorher folgendes zu vereinbaren:

- Taschenlampe, grünes Licht: 5 Punkte oder besser („Daumen hoch")
- Taschenlampe, rotes Licht: unter 5 Punkten („Daumen runter")

Der PC bietet eine ähnliche Möglichkeit. Über Leitungen kann man ein entsprechendes „binäres" („Null oder Eins"; „An oder Aus") Signal schicken. Es kann an einer festen Stelle im Rechner eine Spannung anliegen oder eben nicht. Damit ist es prinzipiell möglich, in einem Rechner Daten zu speichern. Allerdings nur 0-1-Daten (**Bits**) und noch keine komplizierten Texte usw.

Um dieses Problem zu umgehen, werden Pakete von diesen 0-1-Daten gesendet. In unserem anschaulichen Beispiel willst du vielleicht doch die genaue Note wissen. Wir einigen uns auf eine Folge von 4 Blinkern. Wenn wir alle Notenpunkte binär entwickeln, dann finden wir:

0 Notenpunkte = $(0000)_2$ bzw. 1 Notenpunkt = $(0001)_2$ bzw. 2 Notenpunkte = $(0010)_2$ bzw.
3 Notenpunkte = $(0011)_2$ bzw. 4 Notenpunkte = $(0100)_2$ bzw. 5 Notenpunkte = $(0101)_2$ bzw.
6 Notenpunkte = $(0110)_2$ bzw. 7 Notenpunkte = $(0111)_2$ bzw. 8 Notenpunkte = $(1000)_2$ bzw.
9 Notenpunkte = $(1001)_2$ bzw. 10 Notenpunkte = $(1010)_2$ bzw. 11 Notenpunkte = $(1011)_2$ bzw.
12 Notenpunkte = $(1100)_2$ bzw. 13 Notenpunkte = $(1101)_2$ bzw.
14 Notenpunkte = $(1110)_2$ bzw. 15 Notenpunkte = $(1111)_2$!

Grünes Licht ist eine 1, rotes Licht die Null. Wir blinken also für 11 Notenpunkte grün, blau, grün, grün. Übrigens könnte man auch einfach umgekehrt blinken, aber wir legen uns halt von links nach rechts fest.

Im PC geht's genauso, er bekommt jetzt Datenpakete zu 8 Bits geschickt, was einem **Byte** entspricht. Damit lassen sich bereits 256 Dinge kodieren. Man hat sich auf die Zeichen geeinigt, mit denen man möglichst einfach Daten kodieren kann. Da man so auch Buchstaben kodiert, lassen sich durch lange Ketten von Bytes ganze Texte kodieren.

Nur binär zu arbeiten, macht sehr lange Zeichenketten nötig. Man wollte das vermeiden und hat sich auch auf eine Darstellung im 16ner-System geeinigt. Wichtig ist, dass 16 eine Zweierpotenz ist! Diese Entwicklung ist eigentlich eine 2er-Entwicklung… nur kürzer. Da man jetzt für jeder Stelle (1er, 16er, 16^2er usw.) je 16 Zeichen braucht, hat man neben 0...9 auch noch A-F hinzugenommen. Sie ersetzen 10-15. In Farbkodierungen findet man oft #FF0066. Das ist eine

Hexadezimalcodierung (in „Schlau“ für 16). Der Trick ist hier, dass (wie gesehen!) mit vier Bits eine Zahl von 0-15 dargestellt werden kann. Also fassen wir ein Byte als zwei Blöcke á 4 Bits auf. Dann können wir schon mit nur zwei Symbolen (je von 0 bis F) die 256 Zeichen beschreiben. Ist nicht nötig, erleichtert aber den Alltag.

Die modernen PCs haben oft Speicher von bspw. 160 GB (mein Netbook). Ein Byte sind 8 Bit, grob 10. Giga ist Mega mal 1000, Mega ist eine Million. Giga ist also eine Milliarde. somit haben wir etwa 1600 Milliarden Bits zur Verfügung. Als Vergleich: Wir könnten eine Nachricht mit der Taschenlampe kodieren, für die wir 1600 Milliarden mal blinken müssen. Blinken wir einmal in der Sekunde, kannst du dir vorstellen, wie lange das dauert und die Batterien wären auch bald leer. Daher sind Computer so „mächtig“ und damit sehr nützlich.

Der nächste Schritt ist dann das Verarbeiten der Daten. Wie kann eine gespeicherte Nachricht elektronisch bearbeitet werden? Bzw. sind Zahlen kodiert, wie kann man sie verrechnen?! Auch hier geht alles unglaublich schnell mithilfe moderner Elektronik und der Darstellung in Bits.