Rechnernutzung in der Physik

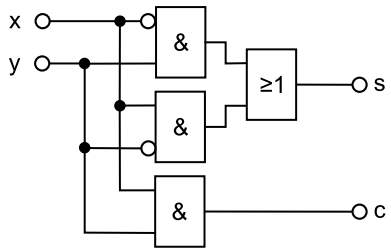
Institut für Experimentelle Teilchenphysik
Institut für Theoretische Teilchenphysik
Interfakultatives Institut für Anwendungen der Informatik

Dr. Th. Kuhr, Prof. Dr. M. Steinhauser, Prof. Dr. U. Husemann
Mildenberger / Hoff / Hermann / Heck
WS2012/13 – Blatt 02
`http://comp.physik.kit.edu` Prog: Di, 30.10.2012 / Ausarb: Fr, 02.11.2012

Das Passwort für den Online-Status von Testaten und Auswertungen auf der Kurswebseite ist `qubit` (klein geschrieben).

Aufgabe 4: Addition von Binärzahlen **Programmtestat und Ausarbeitung**[1]

Ein sogenannter „Halbaddierer“ addiert zwei binäre Signale (z.B. dargestellt durch 0V oder 5V Spannung). Zur Darstellung des Ergebnisses hat der Halbaddierer zwei Ausgänge, einen für die niederwertigste Stelle s der Summe und einen zur Darstellung des Übertrags c (*carry*). Mit Logikbausteinen kann das z.B. folgendermaßen realisiert werden:



Wenn man Binärzahlen mit mehr als einer Stelle addieren möchte, ist bei der Addition jeder Stelle ein eventueller Übertrag einer niederwertigeren Stelle zu berücksichtigen. Es sind also drei binäre Eingangssignale zu addieren. Diese Schaltung heißt „Volladdierer“. Nach wie vor genügen zwei Ausgänge, die die gleichen Bedeutungen wie beim Halbaddierer haben. Überlegen Sie sich bitte, warum diese ausreichen.
Entwickeln Sie einen Volladdierer, indem Sie nur auf elementare Logikfunktionen zurückgreifen.

Programmtestat: Schreiben Sie zwei C++-Funktionen `adder_sum` und `adder_carry`, die jeweils drei boolsche Argumente haben. Die Argumente stehen für die zu addierenden Werte. Rückgabenwert der Funktion soll jeweils ein `bool` sein. Die Funktionen sollen lediglich logische Rechenoperationen (also `and`, `or`, `not` bzw. die Kurzformen `&&`, `||`, `!`) verwenden.
Lassen Sie im Hauptprogramm den Benutzer drei `bool`-Werte (am einfachsten durch 0 oder 1 dargestellt) eingeben und geben Sie die Funktionswerte beider Funktionen aus.

Ausarbeitung: Zeichnen Sie und erklären Sie kurz das Schaltbild Ihres Volladdierers. Notieren Sie eine vollständige Wahrheitstabelle (Wertetafel). Diese enthält für alle Eingangswertekombinationen der drei Eingänge die Signale an den Ausgängen s und c.
Entwerfen Sie eine Schaltung, um mit drei Volladdierern zwei dreistellige Binärzahlen (jede Zahl hat drei Leitungen) zu addieren und erklären Sie kurz ihre Schaltung. Sie können dabei ein eigenes Schaltsymbol für einen Volladdierer einführen, um nicht alle Logikgatter einzeln zeichnen zu müssen.

---

[1] Es werden im ersten Block der Vorlesung nur zwei schriftliche Ausarbeitungen angeboten, diese sind beide erforderlich.

Aufgabe 5: Zahlendarstellung im Rechner **Programmtestat**

Schreiben Sie ein C++-Programm, welches für eine eingegebene Integer-Zahl folgendes ausgibt: Die Zahl selbst, die Speicheradresse der Zahl, die Länge in Bytes und den Inhalt der einzelnen Speicheradressen in binärer Form. Diese Ausgaben sollen in einer Funktion vorgenommen werden, die als Argument die eingegebene Zahl erhält (z.B. als Referenz `const int &`).
Testen Sie Ihr Programm mit kleinen positiven ganzen Zahlen und mit Zweierpotenzen. Welche Bits haben welche Wertigkeit? Wie werden negative ganze Zahlen dargestellt?

Anleitung: Sie benötigen nacheinander den Zugriff auf die einzelnen Bytes der `int`-Zahl. Mit dem Operator `sizeof(...)` können Sie die Anzahl der Bytes ermitteln, aus dem der Datentyp `int` besteht.
`char` umfasst per Definition in C++ genau ein Byte. Besorgen Sie sich also per erzwungener Typenumwandlung `(char *)& zahl` einen Zeiger auf das erste Byte Ihrer `int`-Zahl.
Geben Sie nun ab dieser Speicheradresse die Bytes der Ursprungszahl binär aus.

Zusatz (freiwillig): (i) Verwandeln Sie Ihre Darstellungsroutine in eine `template`-Funktion, so dass sie für beliebige Datentypen funktioniert. (ii) Untersuchen Sie `unsigned int`, `bool` oder Gleitkommazahlen mit Ihrem Programm.

Hinweis:
Mit einem ssh-Klienten und dem Zielrechner

`fphctssh.physik.uni-karlsruhe.de`

können Sie sich von einem beliebigen Rechner aus auf einem Poolrechner einloggen.
