



Prof. Dr. Bernd Becker
Dipl. Inf. Tobias Schubert
M. Sc. Matthew Lewis

**Mobiles Hardware-Praktikum
Sommersemester 2003**

Versuch 1

Zum Aufbau kombinatorischer und sequentieller Schaltkreise

Aufgabe 1

In der ersten Aufgabe soll die Basis für ein einfaches Rechenwerk, das Addition und Subtraktion beherrscht, gelegt werden: ein 4-Bit Addierer-/Subtrahierer-Schaltkreis. Gehen Sie hierzu wie folgt vor:

- Entwerfen Sie einen Halbaddierer und geben Sie diesen als sogenannte GDF-Datei („Graphic Design File“) in die Software MAX+PLUS II Baseline ein. Kompilieren Sie Ihren Entwurf für das FPGA EPM7128SLC84-15 und generieren Sie ein Symbol, um in den nachfolgenden Schritten auf diesen Baustein zugreifen zu können.
- Realisieren Sie aufbauend auf dem Halbaddierer einen Volladdierer als GDF-Datei. Kompilieren Sie Ihren Schaltkreis und führen Sie eine Simulation durch.
- Nutzen Sie den Volladdierer, um einen 4-Bit Conditional-Sum Addierer zu entwickeln. Definieren Sie innerhalb der GDF-Datei die Ein- und Ausgänge jeweils als Bus, um die Übersichtlichkeit zu erhöhen (siehe auch *MAX+PLUS II Getting Started*, Seite 181ff). Für die Simulation von Bussen ist es notwendig, dass bei der Eingabe das höchstwertigste Bit zuerst aufgezählt wird, bei einer 4-Bit Leitung A bedeutet dies A[3..0] anstelle von A[0..3].
- Erweitern Sie den 4-Bit Addierer zu einem Addierer-/Subtrahierer-Schaltkreis, so wie dies in der Vorlesung *Technische Informatik II* vorgestellt wurde. Kompilieren Sie Ihren Schaltkreis und überprüfen Sie die Korrektheit anhand einer Simulation.

Aufgabe 2

Nutzen Sie Ihren Addierer-/Subtrahierer-Schaltkreis aus Aufgabe 1, um die in Abbildung 1 skizzierte 4-Bit Recheneinheit zu realisieren. In Abhängigkeit von der per Tastendruck gewählten Operation (Addition, Subtraktion oder Clear) sollen die beiden 4-Bit Eingangswerte A und B entweder addiert oder subtrahiert und das Ergebnis angezeigt werden (Im Falle von Clear soll das Ergebnis 0 sein). Der Einfachheit halber können Sie davon ausgehen, dass A stets größer oder gleich B ist.

Beachten Sie, dass die Taster für die Auswahl der gewünschten Rechenoperation nur einen „kurzen“ 1-Impuls beim Drücken der entsprechenden Taste auslösen. Deshalb muss eine Steuerlogik von Ihnen entworfen werden, die einen stetigen *Operationscode* erzeugt, der nach dem Drücken der Taste bis zum nächsten Tastendruck erhalten bleibt. Eine Möglichkeit diesen *Operationscode* mit Hilfe zweier Signale **opsel0** bzw. **opsel1** zu definieren, zeigt die nachfolgende Tabelle:

+	-	Clear	opsel1	opsel0	Funktion
0	0	0	X	X	---
0	0	1	0	0	Ergebnis auf 0 setzen
1	0	0	0	1	Addition
0	1	0	1	0	Subtraktion

Implementieren Sie Ihr Rechenwerk mit der Software MAX+PLUS II als GDF-Datei und führen Sie eine Simulation für das FPGA EPM7128SLC84-15 durch.

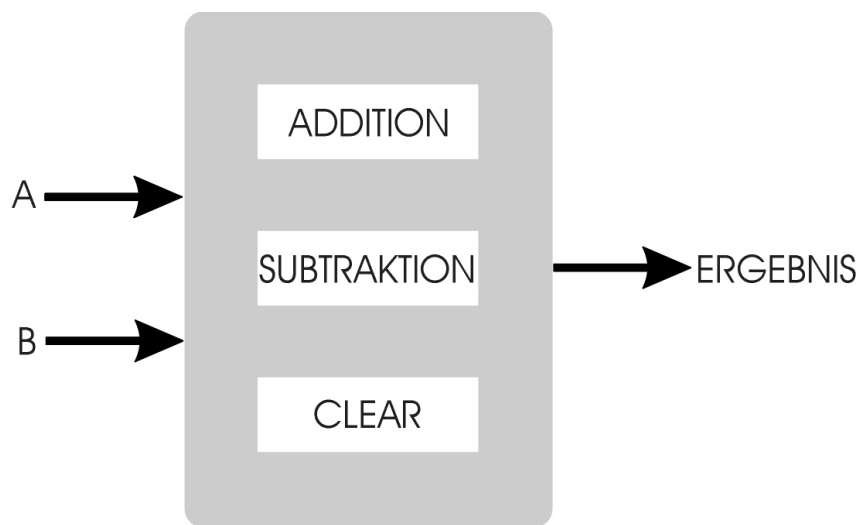


Abbildung 1: Die geplante Recheneinheit

Geben Sie alle von Ihnen zu den zwei Aufgaben erzeugten Schaltkreis-, Symbol-, VHDL- und Simulations-Dateien (Dateiendung GDF, SYM, VHD bzw. SCF) mit entsprechender eindeutiger Namensgebung als eine Winzip Datei über das Übungsportal ab.