**MPS – Praktikum Termin 1 – Sebastian Zill(769544)**

1. **Installieren Sie einen GNU-C-Compiler auf Ihrem PC, beispielsweise *CodeBlocks*.**
   1. **Testen Sie die Entwicklungsumgebung durch Erstellen eines Konsolenprojekts (*Console application*).**
   2. ***Achtung: bei CodeBlocks muss die Konsole explizit von Ihnen geschlossen werden, bevor neu kompiliert und ausgeführt werden kann!!***
   3. **Nehmen Sie einige einfache Änderungen und Erweiterungen an Ihrem Programm vor und testen Sie.**

Notiz: Probleme mit Standard-Codeblocks Installation gehabt:  
– Mingw zusätzlich laden & installieren (in C:/Mingw packen - autodetect)

1. **Betrachten Sie das beigefügte Projekt und erweitern Sie**
   1. **Was machen die Precompiler-Statements \_\_DATE\_\_, \_\_TIME\_\_, \_\_LINE\_\_?**

\_\_DATE\_\_: Gibt das aktuelle Kompilierdatum aus Systemzeit im Format MM DD YYYY anstatt des Platzhalters %s aus

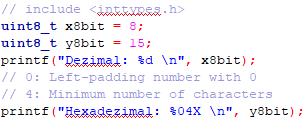
\_\_TIME\_\_: Gibt die aktuelle Kompilierungsuhrzeit aus Systemuhrzeit im Format HH:MM:SS anstatt des Platzhalters %s aus

\_\_LINE\_\_: Gibt die Codezeile anstatt des Platzhalters %d an der Stelle aus

****

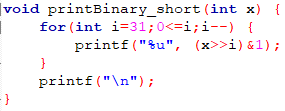
****

* 1. **Geben Sie eine 8bit-Integer Zahl (uint8\_t) als Dezimal- und Hexadezimal-Wert aus.**



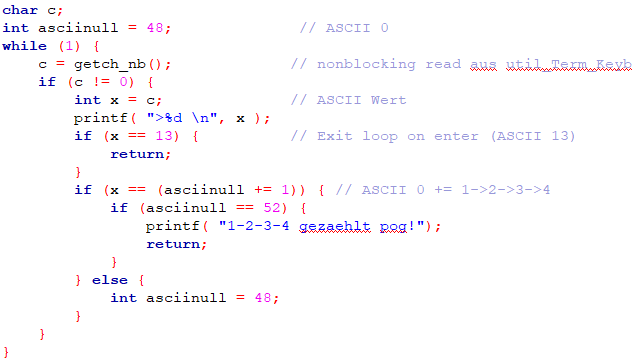


* 1. **Wie kann der Wert binär ausgegeben werden? (eigener Code)**
  2. **Ist Ihre Binärdarstellung little- oder big-endian?**Hier: Mit Funktion 127 aufgerufen

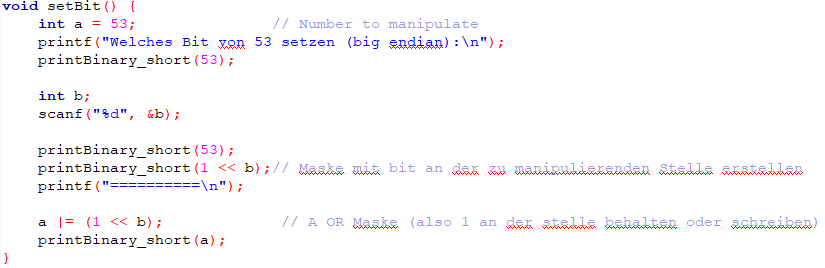




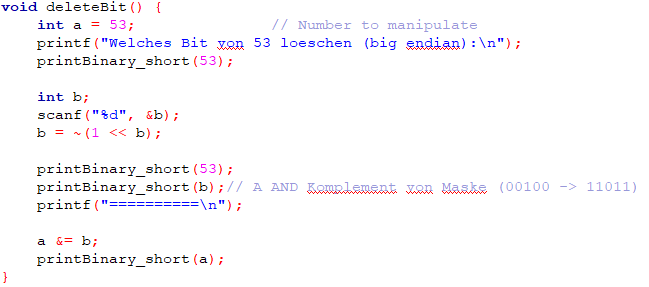
1. **Characters lesen vom Terminal, d.h. auf dem PC von der Tastatur**
   1. **Um das Verhalten der Funktion getc auf einem ARM-Prozessor zu simulieren, ist die Funktion getch\_nb()*(nicht blockierend)* vorgeschlagen. (Ohne Eingabe wird der Wert ‚0‘ zurückgegeben oder bei Tastendruck der ASCII-Wert)**
   2. **Testen Sie das Verhalten der Funktion.**
   3. **Führen Sie bestimme Aktionen aus bei einer Reihe von Eingaben, z.B. 0, 1, 2, 3.**



1. **Setzen und Löschen einzelner Bits.**
   1. **Wie können mit C-Code einzeln Bits gesetzt werden?**

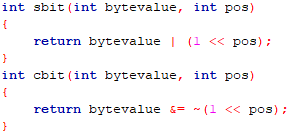
****

* 1. **Wie können mit C-Code einzeln Bits gelöscht werden?**

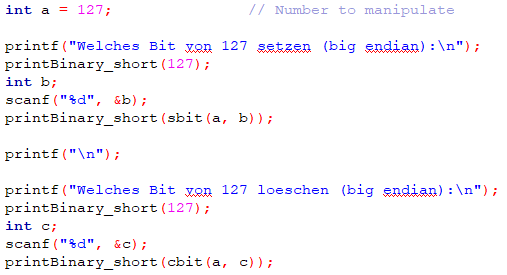
****

* 1. **Schreiben Sie zwei wiederverwendbare Funktionen uint8\_t sbit(bytevalue, pos) und uint8\_t cbit( bytevalue, pos).**
  2. **Testen Sie mit 2.3.**

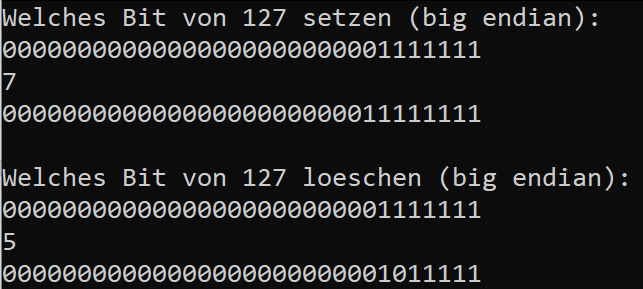
util\_Term\_Keyb.c:

****

main.c:

****

output:

****

* 1. ***Was ist ein Macro?***

Ein Makro ist eine Quelltextersetzung, die vor dem Kompiliervorgang durchgeführt wird. Sie wird häufig zur Definierung von Konstanten verwendet.

1. ***Kommentieren Sie – gegebenenfalls nach dem Praktikum zu Hause – Ihren Code. Archivieren Sie Ihr Projekt zu Ihrem späteren Gebrauch.***
2. ***Schreiben Sie ein kurzes Protokoll und fassen Sie Ihre Erkenntnisse zusammen und fügen Sie die jeweiligen Codeabschnitte hinzu. Laden Sie Ihren Code+) als \*.zip und Ihr Protokoll als \*.pdf in Moodle hoch bis maximal 1 Woche nach dem Termin.***