# Consolenprozess (Process C)

Funktion „***process c“***

Zunächst wird der Stackpointer von „process c“ kopiert.

Es startet eine Endlosschleife. In dieser wartet eine weitere Schleife auf Eingaben auf PORT1. Ist eine Eingabe erfolgt setzt sich das RI0-Empfangsflag und die Schleife endet. Die Eingabe wird anschließend in Register 7 (R7) kopiert. Danach wird die Funktion zur weiteren Eingabebehandlung („handleSerial0Input“) aufgerufen. Das RI0-Empfangsbit wird zurückgesetzt und die Endlosschleife von „process c“ beginnt von vorne.

Zusätzlich wird in der Endlosschleife zyklisch der Watchdogtimer zurückgesetzt. Andernfalls würde der Watchdog nach einer bestimmten Anzahl von Durchläufen annehmen, das Programm befände sich in einer Verklemmung und alle Register zurücksetzen.

Funktion „***handleSerial0Input***“

Gemäß der Aufgabenstellung ist der weitere Programmverlauf abhängig von der Tastatureingabe

|  |  |
| --- | --- |
| Taste | Aktion |
| A | Prozess A starten |
| B | Prozess A beenden |
| C | Prozess B starten |
| Sonst | Keine Aktion |

Hierzu wird der bedingte Aufruf CJNE verwendet. CJNE springt an das angegebene Label wenn die Bedingung nicht erfüllt wird. War die Eingabe weder a,b oder c, so wird nur der Inhalt von Register 7 (R7) gelöscht und zurück in die Endlosschleife von „***process c***“ gesprungen. Erfolgt eine gültige Eingabe wird der jeweilige Datenzeiger kopiert. Durch das Bit „isNew“ und „isDel“ wird angegeben ob der Prozess gestartet oder beendet wird.

# Prozess A

Funktion von Prozess A ist es im 1-Sekunden Takt den Buchstaben ‚a‘ auf PORT2 zu schreiben. Prozess A beendet sich nicht selbstständig sondern wird durch Benutzereingaben beendet.

Funktion „processA“

Zunächst wird das Register von „process a“ kopiert. Anschließend wird die Register 5 (R5) auf 246 gesetzt. Eserfolgt der Aufruf der „mainloop“ diese ruft abwechselnd die Funktion „printAToUART“ und

„waitRoutine“ auf.

Funktionsabschnitt „printAToUART“

„a“ wird auf PORT2 geschrieben. Danach wird durch zyklisches Abfragen von S0CON auf Beendigung des Sendevorgangs geprüft. Ist diese erfolgt wird das TI1-Empfangsbit zurückgesetzt und der Funktionsabschnitt endet.

Funktionsabschnitt „waitRoutine“

Das 5.Bit von Timercontrol wird gesetzt und Register 1 (R1) zurückgesetzt. Die darauffolgende Schleife wiederholt sich nur wenn das 5.Bit von Timercontrol nicht gesetzt ist. Nach zurücksetzen des Watchdogs und von Timercontrol wird der Wert von R5 dekrementiert. Erreicht dieser 0 endet die „waitRoutine“

# Prozess B

Aufgabe von Prozess B ist die Zeichenfolge „54321“ auf PORT2 zu schreiben und sich anschließend selbstständig zu beendent.

Der Prozess beginnt mit Kopieren des Registers für Prozess B. Es wird der Dezimal 53 (ASCII-Wert für 5) in Register 1 (R1) kopiert. In der folgenden Schleife wird nun der Wert von Register 1 auf PORT2 kopiert. Die innere schleife wartet wieder auf Beendigung des Sendevorgangs, ist dieser vollendet wird das TI1-Empfangsbit zurückgesetzt. Der Wert von Register 1 wird dekrementiert und die Schleife wiederholt solange bis Register 1 den Wert 48 erreicht. Dadurch lässt sich durch den Einsatz von nur einer Schleife die Ausgabe „54321“ generieren (ASCII-Werte : 53 = „5“, 52 = „4“, 51 = „3“, 50 = „2“, 49 = „1“, 48 = „0“). Anschließend beendet sich der Prozess durch setzen des #isDel-Bits selbst.