Grundstruktur

**Beispielprogramm**

Das Assemblerprogramm hat folgende Grundstruktur:

.syntax unified

Main:

Initialisierung

endlos

Abbildung 1: Darstellung als PAP (Programm-Ablauf-Plan)

**.include** "../src/regs.asm"

**.global** main

**main:**

bl startup

ldr R1,=GPIOC

**schleife:**

mov R0,#0

strb R0,[R1,ODR]

mov r0,#1000

bl wait\_ms

mov R0,#0xFF

strb R0,[R1,ODR]

mov r0,#1000

bl wait\_ms

b schleife

.end

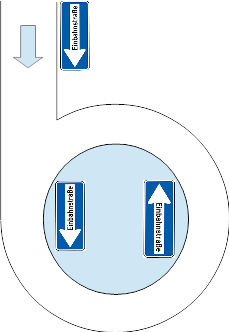
**1. Fragen**

* Was bewirkt die Anweisung **b schleife**?
* Welche Anweisungen werden nach dem Einschalten nur einmal ausgeführt?
* Welche Anweisungen werden endlos wiederholt? Warum?
* Wie lautet die Sprunganweisung?
* Warum ist die Endlosschleife unverzichtbar?
* Warum wird die Ausführung mancher Anweisungen nur einmal benötigt

**2. Schwungübung Schleifenrennbahn**

Ziel: Nachvollziehen, wie der Mikrocontroller in die Endlosschleife saust und wie er sich dabei fühlt.

START



Anleitung: Starten Sie mit einem pinken Farbstift bei START. Drehen Sie mindestens 20 Runden in der Endlosschleife. Achten Sie darauf, die Umrandung nicht zu berühren. Jede Berührung gibt 1 Punkt Abzug:

Ergänzen Sie links neben der Rennbahn:

*main:*

*schleife:*

*b schleife*

**2. Legende**

Füllen Sie die Legenden mit

.syntax unified

**.include** "../src/regs.asm"

**.global** main

**main:**

bl startup

ldr R1,=GPIOC

**schleife:**

mov R0,#0

strb R0,[R1,ODR]

mov r0,#1000

bl wait\_ms

mov R0,#0xFF

strb R0,[R1,ODR]

mov r0,#1000

bl wait\_ms

b schleife

.end

folgenden Begriffen aus:

* Initialisierung
* Endlosschleife
* Programmstart
* Sprungmarke