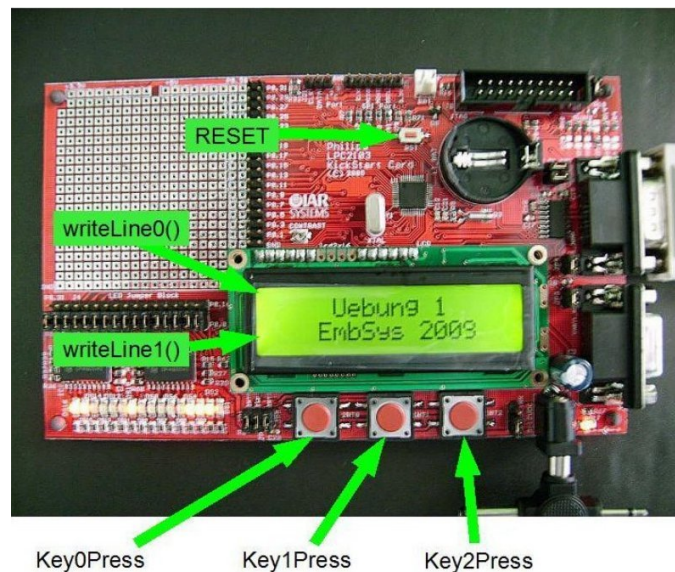


## Übungsblatt 2

(10 Punkte)

### Präsenzaufgaben zur Woche ab Montag, 31. Oktober 2016

Die praktischen Übungen finden im Raum OH16 / U08 statt. Die Bearbeitung der Aufgaben erfolgt in den Übungen.



**Tip:** Simulieren Sie ihr StateChart zuerst und testen Sie es dann erst auf der realen Hardware.

#### Hinweise:

- Für dieses Blatt wird die virtuelle Maschine **CPSF** verwendet.
- Die Archive mit den Templates befinden sich auf dem Labor-Server, wie schon zuvor unter `\\pdc\cpsf`. Die Templates für die zwei Aufgaben sind unterschiedlich!

### 2.1 Wrap-Around-Counter (4 Punkte)

Erstellen Sie einen Wrap-Around-Counter, der von 0 bis 9 hoch- und runterzählen kann. Bei Erreichen der 9 bzw. 0 wird die Zählung am anderen Ende der Zahlenreihe fortgesetzt, also 8,9,0,1,... bzw. 1,0,9,8,... Des Weiteren soll es möglich sein, den Counter zurückzusetzen mittels Tastendruck wieder auf den Wert 0 zurückzusetzen. Benutzen Sie die Tasten zur Steuerung Ihres Counters.

#### Hinweise:

- Benutzen Sie zur Ausgabe der Zahl die Funktion `writeLine0(VS_INT number)`.
- *Action Expressions* sind normale C-Ausdrücke und können daher die üblichen Operatoren wie z.B. Addition (+) oder Subtraktion (−) verwenden. Beachten Sie jedoch, dass der Modulo-Operator (%) für negative Zahlen auch negative Ergebnisse liefert!
- Durch Anlegen einer *Guard Expression* an einer Transition kann bestimmt werden, dass diese nur ausgeführt wird, wenn ihr Event auftritt und gleichzeitig die *Guard Expression* wahr ist. Auch hier werden C-Ausdrücke verwendet, also z.B. `foo == 42`.

## 2.2 Ventilator-Steuerung (6 Punkte)

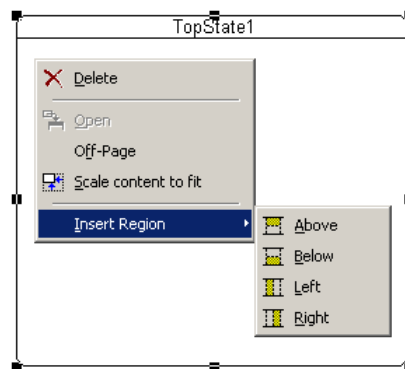
Es soll eine Steuerungssoftware für einen Tisch-Ventilator entwickelt werden. Der Ventilator verfügt über drei Tasten: Eine zum Ein- bzw. Ausschalten, eine zur Geschwindigkeitsregelung und eine um die Drehbewegung des Rotorgehäuses festzulegen. Es gibt drei Geschwindigkeitsstufen: *slow*, *med*, *fast*. Wird bei *fast* der Geschwindigkeitsknopf erneut betätigt, so wird die Geschwindigkeit auf *slow* geregelt. Wird der Ventilator ausgeschaltet, so soll der Zustand der Gehäusebewegung beim Einschalten **wiederhergestellt** werden. Es ist Ihnen überlassen, ob der Ventilator standardmäßig ein- oder ausgeschaltet ist.

Zur Ausgabe auf dem LCD-Display sind folgende Funktionen im Template bereits implementiert:

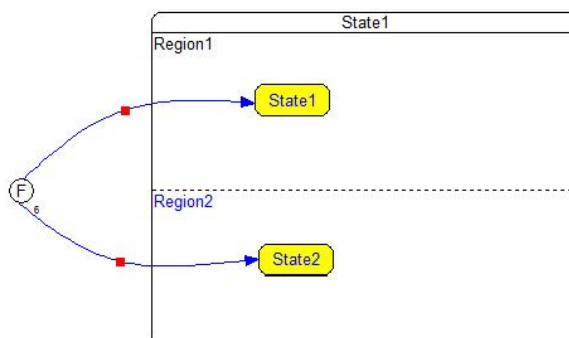
- `writeLine0(-1)` gibt auf dem Display "OFF" aus
- `writeLine0(0)` gibt auf dem Display "Speed: slow" aus.
- `writeLine0(1)` gibt auf dem Display "Speed: med" aus.
- `writeLine0(2)` gibt auf dem Display "Speed: fast" aus.
- `writeLine1(0)` gibt auf dem Display "Moving: yes" aus.
- `writeLine1(1)` gibt auf dem Display "Moving: no" aus.

### Anmerkungen:

Um mit der Software einen AND-Zustand zu erstellen, fügen Sie einen "Composite-State" ein. Klicken Sie im Anschluss mit der rechten Maustaste auf den Zustand und wählen Sie "Insert Region" und dort z.B. "Above".



Um über eine Transition Unterzustände von AND-Zuständen zu aktivieren, reicht es in der Software nicht aus, die Transition auf den AND-Zustand zu ziehen. Die Transition muss in diesem Fall auf einen Fork-Knoten zeigen, von dem im Folgenden jeweils eine neue Transition zu den Unterzuständen zu ziehen ist, siehe folgende Abbildung:



Beim Verlassen des AND-Zustands kann hingegen wie gewohnt eine Transition vom AND- zum Folgezustand gezogen werden.