

## CLD-Übung 12: Zustandsautomat zum Sequenzieren von Schritten

### Ziel

Entwickeln eines Zustandsautomaten auf Grundlage von Daten, die aus einer Datei geladen werden. Als Grundlage für das VI sollen das vorgegebene Frontpanel (Abbildung 1) und die vorgegebenen Elemente genutzt werden.

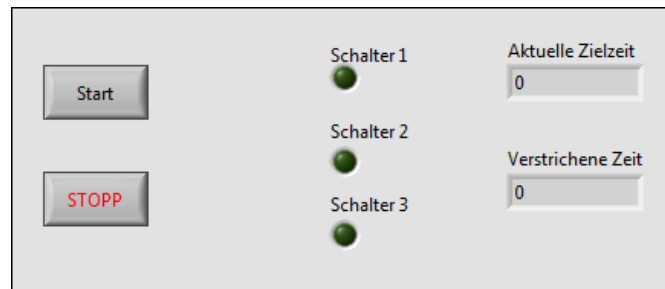


Abbildung 1. Frontpanel der Anwendung

### Allgemeine Arbeitsweise

Die Anwendung führt Schritte in der Reihenfolge aus, die der Datei entnommen wird. Die Schritte legen entweder die Zeitspanne für den Timer fest oder zeigen die booleschen Zustandsdaten und die verstrichene Zeit an. Das Frontpanel zeigt die **Aktuelle Zielzeit**, die **Verstrichene Zeit** und den Zustand des **Schalters** an. Wenn die Zeit verstrichen ist, führt die Anwendung den nächsten Schritt aus.

### Terminologie der Anwendung

#### **Zielzeit**

Gibt den Timer-Wert (die gewünschte Zeit) in Sekunden an. Dieser Wert kann nur in einem SetTime-Schritt festgelegt werden.

#### **Verstrichene Zeit**

Dieses Anzeigeelement soll fortlaufend die jeweils verstrichene Zeit in Sekunden anzeigen.

#### **Datendateiformat**

Jede Zeile enthält fünf Elemente. Der Schrittbefehl ist das erste Element in jeder Zeile. Die Daten sind die letzten vier Elemente jeder Zeile.

Das Datenformat der fünf Elemente jeder Zeile ist:

- Schrittbefehl, Schrittdauer, Schalter 1, Schalter 2, Schalter 3
- **Beispiel:** SetTime, 5, TRUE, FALSE, TRUE

Der erste Schritt muss ein SetTime-Schritt sein. Anderenfalls wird der Timer auf den Standardwert null für die Zielzeit eingestellt.

**Hinweis:** Die Anwendung darf bei den String-Daten nicht zwischen Groß- und Kleinschreibung unterscheiden.

## Schrittbefehle

Bei den Befehlen handelt es sich um Schritte des Zustandsautomaten. Es gibt zwei Arten von Befehlen:

- **SetTime:** Dieser Befehl stellt die Zeit ein *und* startet den Timer. Der SetTime-Schritt verarbeitet nur die Werte für die Schrittdauer.
- **RunState1 und RunState2:** Es sind zwei RunState-Schritte vorhanden. Ihre Funktion besteht darin, die mit ihnen verbundenen booleschen Werte in den **Schalter-LEDs** anzuzeigen. Diese Schritte geben außerdem die verstrichene Zeit und die aktuelle Zielzeit an. Die RunState-Schritte verarbeiten keine Zielzeiten.

## Initialisierung

Die Anwendung soll entsprechend Abbildung 1 mit den folgenden Einstellungen der Frontpanel-Bedien- und -Anzeigeelemente starten:

- **Aktuelle Zielzeit:** 0 Sekunden
- **Schalter-LEDs:** AUS
- **Verstrichene Zeit:** Null

## Allgemeine Arbeitsweise

### VI-Ausführung

Wenn die **Start**-Schaltfläche betätigt wird, lädt das VI die Dateidaten und beginnt anschließend mit dem Ausführen der Schritte. Der erste Schritt muss ein SetTime-Schritt sein, da der Timer sonst automatisch auf null Sekunden eingestellt wird. Welche Handlungen in nachfolgenden Schritten ausgeführt werden, richtet sich nach dem Schrittbefehl.

#### Befehl „SetTime“:

- verwendet die Schrittdauer für den Timer und startet diesen neu
- soll zum nächsten Schritt wechseln

#### Befehl „RunState“:

- verwendet die booleschen Werten zum Einstellen der Schalter-LEDs
- führt den Schritt bis zum Verstreichen der Zeit aus
- startet den Timer nach dem Verstreichen der Zeit neu und wechselt zum nächsten Schritt

Nach dem Ausführen aller Schritte soll die Anwendung im Ruhezustand verharren. Die Anzeigeelemente sollen alle auf null oder AUS eingestellt werden.

### **Fragen**

Für die Beispiellösung: Welche Probleme entstehen beim Hinzufügen der **Stopp**-Schaltfläche zur Ereignisstruktur?