

CLD-Übung 4: Timeout der Ereignisstruktur

Ziel

Entwickeln eines einfachen Zustandsautomaten mit einer als Timer fungierenden Ereignisstruktur auf Grundlage des vorgegebenen Frontpanels (Abbildung 1). Die Anwendung soll einen Zählwert anzeigen, der durch Anklicken der Schaltflächen **Inkrementieren** und **Dekrementieren** festgelegt wird. Wenn über eine die **Zielzeit (s)** hinausgehende Zeitspanne keine Aktivität verzeichnet wird, soll die LED **Zeit ist verstrichen** leuchten.

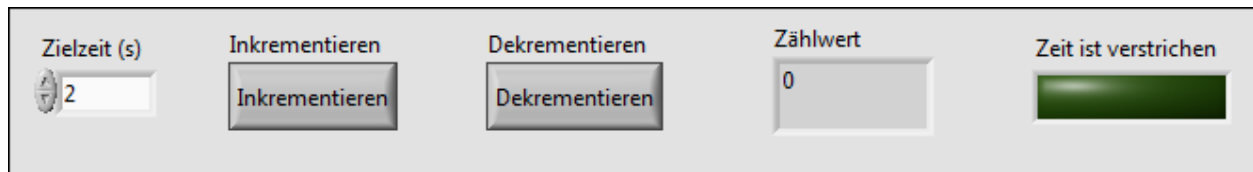


Abbildung 1. Frontpanel der Anwendung

Allgemeine Arbeitsweise

Die Anwendung stellt einen Zählwert bereit, der durch Anklicken der Schaltflächen **Inkrementieren** und **Dekrementieren** eingestellt wird. Wenn auf dem Frontpanel länger als bis zur eingestellten **Zielzeit (s)** keine Aktivität festgestellt wird, soll das Anzeigeelement **Zeit ist verstrichen** auf EIN wechseln und der **Zählwert** auf null zurückgesetzt werden.

Terminologie der Anwendung

Zählwert

Ein mit den Schaltflächen **Inkrementieren** und **Dekrementieren** einstellbarer Wert. Der **Zählwert** ist ein vorzeichenloser Long-Integer. Wenn der Zählwert auf einen Wert kleiner als null eingestellt wird, schaltet das Anzeigeelement automatisch auf den höchsten Integer-Wert um und es wird abwärts gezählt.

Inkrementieren und Dekrementieren

Diese Bedienelemente sollen den Zählwert erhöhen oder verringern.

Zielzeit

Gibt den Timer-Wert (die gewünschte Zeit) in Sekunden an. Wenn die eingestellte Zeitspanne ohne Aktivität auf dem Frontpanel verstreicht, wird der Zählwert auf null zurückgesetzt.

LED „Zeit ist verstrichen“

Diese LED soll leuchten (auf EIN wechseln), wenn bis zum Erreichen der Zielzeit keine Frontpanel-Aktivität festgestellt wird. Die Anzeige bleibt eingeschaltet, bis Frontpanel-Aktivität erkannt wird.

Initialisierung

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

- **Zielzeit:** 2 Sekunden
- **Zählwert:** Auf null setzen
- **Zeit ist verstrichen:** AUS

Arbeitsweise

VI-Ausführung

Das VI beginnt nach dem Start mit der Zeitmessung. Wenn länger als bis zur **Zielzeit** keine Aktion auftritt, soll das VI den **Zählwert** auf null setzen und die LED **Zeit ist abgelaufen** einschalten.

Betätigen der Schaltflächen „Inkrementieren“ und „Dekrementieren“

Das VI stellt basierend auf der Einstellung der Schaltflächen **Inkrementieren** und **Dekrementieren** einen veränderlichen Zählwert bereit.

Festlegen der Zielzeit

Wenn die **Zielzeit** geändert wird, soll der Timer sofort mit dem neuen Wert arbeiten.

Fragen

Ist die Nutzung eines Timeout-Cases in einer Ereignisstruktur eine gute Methode zum Messen der verstrichenen Zeit?

Welche Vor- und Nachteile hat die Timeout-Konstante im Timeout-Case einer Ereignisstruktur?

Zusatzaufgabe

Entwickeln eines Timers unter Zuhilfenahme des Ereignisstruktur-Timeouts. Die Anwendung dient außerdem als Bit-Zähler und enthält Anzeigeelemente für die vier Bits. Da vier Bits vorliegen, müssen alle Zeitschritte ein Sechzehntel der **Zielzeit** betragen.

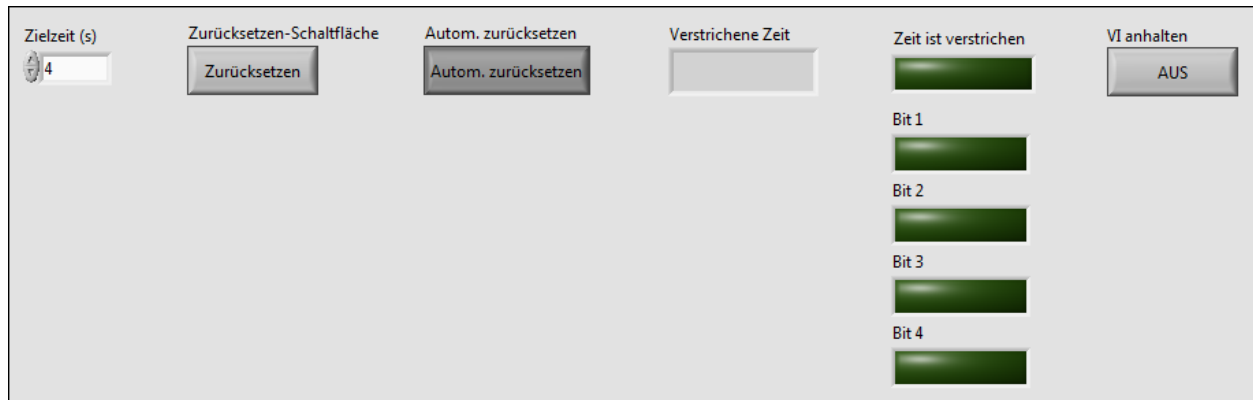


Abbildung 2. Ereignisstruktur-Timer und Bit-Zähler

Allgemeine Arbeitsweise

Die Anwendung misst die verstrichene Zeit und zeigt diese in Millisekunden an. Die Zeit wird mit Hilfe des Timeout-Werts des Ereignis-Cases gemessen. Die Zielzeit muss in 16 Schritte aufgeteilt werden, so dass die vier Bit-Anzeigen jeden Schritt als Binärzahl angeben können.

Terminologie der Anwendung

Zielzeit

Gibt den Timer-Wert (die gewünschte Zeit) in Sekunden an. Wenn dieser Wert geändert wird, soll die neue **Zielzeit** sofort in Kraft treten.

Zeit ist verstrichen

Soll beim Erreichen des Timer-Werts auf EIN wechseln. Solange die Zeit noch nicht verstrichen ist, soll die Anzeige auf AUS verbleiben.

Bits 1 - 4

Diese Anzeigeelemente stehen für die vier Bits und müssen ein- und ausgeschaltet werden. **Bit 1** steht für den niedrigsten und **Bit 4** für den höchsten Stellenwert. Die **Zielzeit** muss durch 16 dividiert werden, so dass sich 16 Binärzustände ergeben.

Wenn die **Zielzeit (s)** beispielsweise 4 Sekunden beträgt, müssen sich die Anzeigeelemente jede Viertelsekunde ändern. In Abbildung 3 beträgt die verstrichene Zeit 2,5 Sekunden, was 10 von 16 Viertelsekunden – ausgedrückt durch die Binärzahl 1010 – entspricht.

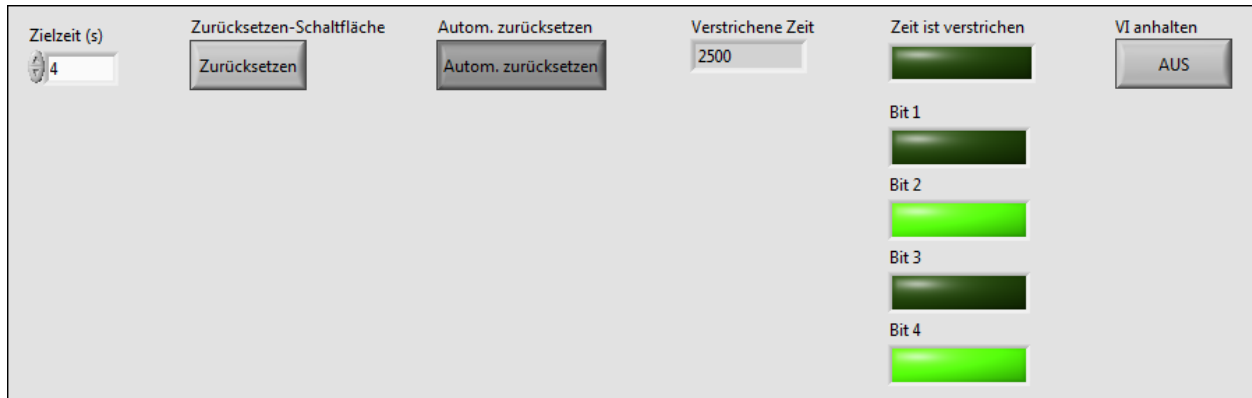


Abbildung 3. Bit-Zähler-Beispiel

Zurücksetzen

Beim Betätigen der Schaltfläche **Zurücksetzen** soll der Timer wieder bei null beginnen und alle Anzeigeelemente müssen auf FALSE zurückgesetzt werden, so dass der Zählvorgang wieder von vorn beginnen kann.

Autom. zurücksetzen

Wenn diese Schaltfläche TRUE ist, soll der Timer beim Erreichen der Zielzeit wieder von vorn beginnen. Wenn dieses Element auf FALSE eingestellt ist, soll die Anwendung nach Ablauf der Zeit anhalten.

Initialisierung

Die Anwendung soll entsprechend Abbildung 2 mit den folgenden Einstellungen der Frontpanel-Bedien- und -Anzeigeelemente starten:

- **Zielzeit:** 4 Sekunden
- **Autom. zurücksetzen:** EIN
- **Verstrichene Zeit:** Null
- **Zeit ist verstrichen:** AUS
- **Bits 1 – 4:** AUS

Arbeitsweise

VI-Ausführung

Das VI soll die Zeitmessung beginnen, **Verstrichene Zeit** soll aufwärts zählen und die LED **Zeit ist verstrichen** soll sich in AUS-Stellung befinden. Die **Bit**-LEDs sollen einen Bitzähler simulieren, indem sie dem Zählwert entsprechend ein- und ausgeschaltet werden.

Beim Erreichen der **Zielzeit (s)** soll die LED **Zeit ist verstrichen** auf EIN wechseln.

Wenn sich **Autom. Zurücksetzen** in EIN-Stellung befindet:

- wechselt die LED **Zeit ist verstrichen** auf EIN
- muss der Timer auf null zurückgesetzt werden und erneut hochzählen
- wechselt die LED **Zeit ist verstrichen** auf AUS

Wenn sich **Autom. Zurücksetzen** in AUS-Stellung befindet:

- soll die Anwendung anhalten

Festlegen der Zielzeit

Wenn die **Zielzeit** (s) geändert wird, soll der Timer sofort mit dem neuen Wert arbeiten.

Autom. Zurücksetzen festlegen

Die Anwendung muss bei Betätigung der Schaltfläche **Autom. zurücksetzen** sofort reagieren.

- Wenn sich **Zeit ist verstrichen** im AUS-Zustand befindet, soll die Anwendung unabhängig von der Stellung von **Autom. zurücksetzen** ihre Ausführung fortsetzen.

VI anhalten

Die Anwendung soll innerhalb eines Zeitschritts reagieren.

Fragen

Was ist der größte Nachteil dieser Art von Zeitmessung?