

Bereitstellung der Makros für die Echtzeitstruktogramme im Internet

Kurze Erläuterungen

Die hier bereitgestellte Software ist die probeweise Implementierung eines Konzeptes zur Strukturierung von Echtzeitsoftware, die im Rahmen einer Studienarbeit an der TU Clausthal durchgeführt wurde [1]. Eine detaillierte Erläuterung des Konzeptes findet sich in [2, 3]. Eine neuere Beschreibung wird in absehbarer Zeit verfügbar sein [4].

Diese Software eignet sich für zwei **Programmiersprachen**:

- 1) **"Echtzeit-C"** auf der Basis von Siemens SIMATIC für M7-300/400 Systeme.
- 2) **"PEARL90"** auf der Basis von DIN 66253-2. Ein Windows Emulator für ein komplettes PEARL System kann unter <http://www.irt.uni-hannover.de/rtos/freeware/winston-rtos.html> gratis heruntergeladen werden.

Weiterhin wird folgende **Hilfssoftware** benötigt:

Die Makro-Programmiersprache **m4**. Sie ist in UNIX Systemen im Lieferumfang enthalten und für viele weitere Systeme frei zu beziehen.

Für die beiden genannten Programmiersprachen wurden Makros erstellt, deren Verwendung identisch ist. Die folgende Beschreibungselemente wurden realisiert:

| | | PEARL | C |
|-------|---|-------|---|
| 1 | Behandlung von Programmablaufanomalien (Exceptions) | | |
| 1.1 | Ankopplung an eine Exception (TRY) | ✓ | ✓ |
| 1.2 | Auslösen einer Exception (THROW) | ✓ | ✓ |
| 1.3 | Start eines Exception-Handlers | ✓ | ✓ |
| 1.4 | Deklaration einer Exception mit Fehlermeldung bei falscher Verwendung | ✓ | ✓ |
| 1.5 | Reaktionen auf Exceptions | | |
| 1.5.1 | "CONTINUE" | ✓ | ✓ |
| 1.5.2 | "WAIT" | ✓ | ✓ |
| 1.5.3 | "REPEAT" | ✓ | ✓ |
| 1.5.4 | "ABORT" | ✓ | ✓ |
| 2 | Strukturierter Parallelismus | | |
| 2.1 | Deklaration eines parallelen Prozesses (PARALLEL) | ✓ | ✓ |
| 2.2 | Spezifikation der benötigten Ressourcen (CLAIM) | ✓ | ✓ |
| 2.3 | Steuerung des parallelen Ablaufs von Prozessen | | |
| 2.3.1 | "PARALLEL" | ✓ | ✓ |
| 2.3.2 | "PAREND" | ✓ | ✓ |
| 3 | Verwaltung von Ressourcen | | |
| 3.1 | Anforderung von Ressourcen ("DEMAND") | ✓ | ✓ |
| 3.2 | Freigabe von Ressourcen ("FREE") | ✓ | ✓ |
| 4 | Behandlung von Meldungen | | |
| 4.1 | Erzeugen eines Signals ("EMIT") | ✓ | ✓ |
| 4.2 | Warten auf Signal | ✓ | ✓ |

Literatur

1. Trenkel S., Gollub M.: Implementation von Methoden zur zuverlässigen Programmierung von Echtzeitsystemen (Implementation of Methods for Reliable Programming of Real-Time Systems); Student Project in the Department of Mathematics and Computer Science of the Technical University of Clausthal (Dezember 2004)
2. Elzer, P.: Strukturierte Beschreibung von Prozeßsystemen; Dissertation, Arbeitsberichte des Instituts für Mathematische Maschinen und Datenverarbeitung der Universität Erlangen-Nürnberg, Band 12, Nr.1 (Februar 1979)
3. Elzer, P.: Eine Technik zur Konstruktion sicherer und zuverlässiger Echtzeitsysteme; in: Hollecsek, P., Vogel-Heuser, B. (Eds.): Echtzeitaspekte bei der Koordinierung autonomer Systeme; Informatik aktuell, Springer Berlin Heidelberg New York, S. 77-86 (2005)
4. Elzer, P., Trenkel S., Gollub M.: A simple and cost-effective Method for the Construction of Reliable Real-Time Programs; Zeitschriftenartikel in Vorbereitung