

## Test der PEARL-Anwender-Programmbibliothek an verschiedenen Prozeßrechnern

Dipl.-Inform Th. Roehrich, Stuttgart

### Zusammenfassung.

Im Rahmen dieses Beitrages wird ueber Portabilitaetserfahrungen berichtet, die im Zusammenhang mit der Erprobung der PEARL-Anwender-Programmbibliothek gewonnen wurden. Dabei wird speziell dargestellt, welche Syntaxanpassungen bei der Implementierung des Basis-Moduls Analogdatenerfassung auf verschiedenen Prozessrechnern notwendig sind.

Es zeigte sich, dass trotz weitestgehender Einhaltung von Basic-PEARL und unveraenderter Aufgabenstellung bei der Implementierung auf verschiedenen Zielrechnern einige syntaktische Aenderungen im PEARL-Problemteil notwendig waren.

### Schluesselworte.

PEARL, Programmbibliothek, Portabilitaet, Analogwerterfassung.

### Abstract.

The purpose of this part is a report about our experience in the portability in connection to testing of PEARL User Library. The main emphasis is to show which corrections of the syntax have to be made for the implementation of the basis-module Analog-Signal-Input in different processcomputers. It turned out that in spite of sticking strictly to Basic PEARL and the same problem, some corrections of the syntax in the PEARL-problempart had to be made to implement the modules in different target computers.

### Keywords.

PEARL, User Library, Portability, Analog-Signal-Input

### 1. EINFUEHRUNG

Wie im vorangehenden Beitrag /1/ berichtet, wurde an der Abteilung Stromerzeugung und Automatisierungstechnik (IVD) der Universitaet Stuttgart eine Pearl-Anwender-Programmbibliothek (PAP) auf der Grundlage von Basic-PEARL entwickelt. Die Zielsetzung war dabei, moeglichst universell einsetzbare, getestete ( Basic- ) PEARL Programmbausteine auf Prozedur-, Task- und Modulebene fuer den Anwender zur Verfuegung zu stellen.

Durch die Programmierung der PEARL-Anwender-Programmbibliothek in Basic-PEARL mit nur geringen Ergaenzungen aus Full-PEARL ist die Portabilitaet der erstellten Programmbibliothek weitgehend gewaehrleistet. Dennoch erforderliche Syntaxaenderungen bei der Uebertragung von PAP-Prozeduren, -Tasks oder -Modulen auf andere PEARL-Systeme soll im folgenden anhand der Uebertragung des PAP-Moduls "Analogdatenerfassung" auf verschiedene Prozessrechner gezeigt werden.

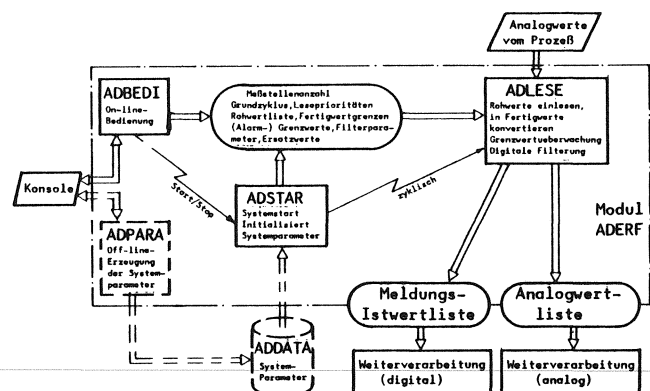


Bild 1: Aufbau des PAP Moduls Analogdatenerfassung

## 2. DER ANALOGDATENERFASSUNGSMODUL

In B I L D 1 ist eine Uebersicht ueber den Aufbau des Analogdatenerfassungs-Moduls ADERF gegeben. Die Tasks sind dabei als Rechtecke, die globalen Daten des Moduls als langgezogene Kreise dargestellt. Der Zugriff der Tasks zu den globalen Daten wird durch gerichtete Pfeile symbolisiert. Der Kontrollfluss wird durch zusaetzliche 'Blitz'-Pfeile von aktivierenden zu aktivierten Tasks veranschaulicht.

Die in B I L D 1 gestrichelt dargestellte Task 'ADPARA' sowie die Datei 'AD-DATA' dienen zur alternativen variablen Initialisierung des Moduls. Bei dieser Variante werden - im Gegensatz zur festen Initialisierung, bei der die notwendigen Parameter fest programmiert sind - die notwendigen Initialisierungen beim Start von der Datei ADDATA eingelesen. Diese Datei kann im vorbereitenden oder im Hintergrundbetrieb mit Hilfe der Task ADPARA erstellt oder veraendert werden. Eine genauere Beschreibung des Moduls Analogdatenerfassung ist in /4/ gegeben.

## 3. PORTABILITAETSERFAHRUNGEN

In diesem Abschnitt wird gezeigt, welche Aenderungen im Problemteil des PAP-Moduls Analogdatenerfassung bei der Implementierung auf verschiedene Prozessrechner, notwendig waren.

Bei der Uebertragung wurde von der Version des Moduls ADERF mit fester Initialisierung ( B I L D 1 ) ausgegangen, die auf dem Prozessrechner Siemens 330 ( mit Siemens PEARL-System ) entwickelt wurde. Dieses Programm wurde von verschiedenen Mitarbeitern nacheinander auf die Prozessrechner EPR 1300 der Fa. Krupp Atlas Elektronik, PDP11/34 der Fa. DEC und AEG 80-20 uebertragen. Da bei den Rechnern EPR 1300 und PDP11/34 keine Prozessperipherie zur Verfuegung stand, wurde fuer diese Rechner der Modul dahingehend abgeaendert, dass

die Messwerte aus einer Pheripherspeicherdatei gelesen werden. Zusaetzlich wurde bei allen Versionen eine Ausgabetask ergaenzt, die es ermoeoglicht, die erfassten Werte als Protokoll auf einem Drucker auszugeben.

Bei der Implementierung des Moduls auf den verschiedenen Prozessrechnern erwiesen sich folgende Aenderungen im Problemteil als notwendig:

### a) Modulname

Bei Siemens 330: 'MODULE;'  
 bei AEG 80-20: 'MODULE;'  
 bei EPR1300: 'MODULE name;'  
 bei PDP11: 'MODULE(name);'

### b) Name der bei Programmstart vom Betriebssystem zu startenden PEARL-Task

Bei den Prozessrechners Siemens 330 und AEG 80-20 ist keine Namenskonvention fuer PEARL-Tasks vorhanden. Bei diesen Rechnern wird jede PEARL-Task in eine vom Betriebssystem verwaltete Task ueberfuehrt. Beim Programmstart wird dann explizit die Starttask gestartet.

Bei den Versionen fuer die Rechner EPR1300 und PDP11 war es notwendig, die Starttask ADBEDI in MAIN umzubenennen. Die Verwendung dieses Standardnamens fuer die Starttask ist notwendig, da bei diesen PEARL-Implementierungen das gesamte PEARL-Programm in eine vom Betriebssystem verwaltete Einheit umgesetzt wird und beim Programmstart bekannt sein muss, welche interne PEARL-Task vom Betriebssystem gestartet werden soll.

### c) Einlesen und Verarbeitung der Bedienzeile in Task ADBEDI

Die on-line Bedienung des Moduls ADERF ist so konzipiert, dass Task ADBEDI eine Bedienzeile vom Bediengerat einliest, analysiert und dann den gewuenschten Befehl ausfuehrt. Danach wird die Befehlsausfuehrung quittiert und Task ADBEDI er-

wartet erneut Bedieneingaben. Eingelesen werden maximal 60 Einzelcharacter, die mit Hilfe einer Prozedur analysiert und zu Schlüsselworten zusammengebaut werden.

Bei der Version fuer den Siemens 330 Rechner konnte die Eingabe wie konstruiert mit Hilfe des Statements:

```
GET ZEILE(1:60) FROM TERMINAL BY (60)A(1);
```

vorgenommen werden. Zudem wird dabei das Bediengerat nicht durch die auf Eingabe wartende Task ADBEDI blockiert, sondern vom gleichen Bediengerat aus koennen gleichzeitig andere Tasks bedient werden.

Bei der Uebertragung auf den Prozessrechner AEG 80-20 musste auf die Eingabe von Einzelcharacter verzichtet werden, da bei der Implementierung der character-verarbeitenden Prozedur LEX zur Analyse des Bedienstrings und Aufbau der Schlüsselworte ein Fehler bei der Characterverarbeitung des verwendeten PEARL-Systems festgestellt wurde. Bei dieser Version wurden deshalb die einzugebenden Schlüsselworte als CHAR(6) Variablen, getrennt durch Blank eingelesen, so dass die Analyse des Bedienstrings und der Aufbau der Schlüsselworte komplett entfallen konnte.

Bei der Version fuer den Prozessrechner EPRL300 musste die Bedienzeile als eine CHAR(60) Variable ohne Formatangabe eingelesen werden und danach in ein Feld (60) CHAR(1) umgespeichert werden. Dies war notwendig, da sonst bei weniger als 60 eingegebenen Zeichen ein Fehler auftrat. Zudem musste eine Reaktion auf ein I/O Error signal im Programm vorgesehen werden, das auftritt, falls der leere Bedienstring, d.h. nur (CR) eingegeben wird. Desweiteren ist bei dieser PEARL Implementierung das Bediengerat bei der auf Eingabe wartende Task ADBEDI fuer andere Bedienungen blockiert.

Bei der Version fuer den Rechner PDP11 musste ebenso wie beim EPRL300 die Bedienzeile in eine CHAR(60) Variable eingelesen und in ein Feld von (60) CHAR(1) umgespeichert werden.

d) Eroeffnung der Datenwege zur Standardperipherie

Bei den Versionen fuer die Prozessrechner Siemens 330 und AEG 80-20 ist das Eroeffnen der Datenwege zur Standardperipherie, d.h. zum Drucker und Terminal nicht erforderlich.

Hingegen ist bei der Version fuer den Prozessrechner EPRL300 ein einmaliges Eroeffnen der Datenkanale zu den Standardperipheriegeraeten Drucker und Terminal erforderlich.

Bei der Version fuer den Rechner PDP 11 ist ein Eroeffnen und Schliessen der Datenwege zum Drucker und Terminal in jeder - diese Geraete benutzende - Task erforderlich.

e) Holen der aktuellen Uhrzeit vom System

Bei der Version fuer den Siemens 330 Rechner wird die im Modul notwendige aktuelle Erfassungszeit mit Hilfe der vom System zur Verfuegung gestellten Funktion NOW ermittelt. Vor Verwendung der Funktion NOW muss diese spezifiziert werden.

Bei der Version fuer den AEG 80-20 Rechner wird die aktuelle Uhrzeit ebenfalls mit NOW ermittelt, wobei die Spezifikation der Systemfunktion NOW vor Verwendung entfallen kann.

Bei der Version fuer den Prozessrechner EPRL300 muss die aktuelle Uhrzeit mit der Funktion DAYTIME ermittelt werden. Zusaetzlich kann die Spezifikation dieser Systemfunktion vor deren Verwendung entfallen.

Bei der PDP11 Version erfolgt die Ermittlung der aktuellen Uhrzeit mit der Systemfunktion NOW. NOW muss vor Verwendung spezifiziert werden.

f) Sonstiges

Ausser den Aenderungen a) bis e) musste bei der Version fuer den PDP 11 Rechner zusaetzlich eine Funktion GETCHAR fuer die Character-Selektion '.CHAR(i)' geschrieben werden, da bei der PDP11-PEARL Implementierung bei obiger Character-Selektion 'i' keine Variable,

sondern nur eine Konstante sein darf.

Zudem wurde bei der Version fuer den AEG 80-20 Rechner die Deklaration von Prozeduren die nur in einer Task benutzt werden nicht auf Modul- sondern auf Taskebene vorgenommen. Dies war notwendig, um das Programm an die vorhandene Speicher- aufteilung der Maschine ( Laufbereiche ) anzupassen.

In T A F E L 1 sind die oben beschriebenen notwendigen Aenderungen bzw. Abweichungen der vier Versionen unter Angabe der 'Verwendung' ( Basis-PEARL, Full-PEARL, Zusatzfunktion ) zusammengestellt.

Speicherplatzbedarf

Der Speicherplatzbedarf fuer den lade- faehigen Objektcode ohne Daten des Modul ADERF war

- bei Siemens 330:
  - alle gebundene Tasks 1) 20k Worte
- bei AEG 80-20:
  - alle gebundene Tasks 1) 3) 22k bytes
- bei KAE 1300:
  - fuer kompl. Programm 2) 25k Worte

bei PDP 11/34:

- fuer kompl. Programm 2) 30k Worte
- incl. Errortask

- 1) Tasks koennen z.T in dieselben Lauf- bereiche geladen werden.
- 2) Speicherplatz fuer Daten abgezogen
- 3) Laufzeitroutinen z.T. in System

Umstellungsaufwand

Die Anpassungen des PAP-Moduls ADERF wurde durch, auf dem jeweiligen Zielrechner, erfahrene PEARL- Programmierer durchgefuehrt.

Der Aufwand fuer die Umstellung des Moduls ADERF, d.h. die Anpassung des Systemteils und obige Aenderungen im Problemtteil war dabei folgender:

- fuer KAE 1300 etwa halber Tag
- fuer PDP 11/34 etwa ein Tag
- fuer AEG 80-20 etwa ein Tag

Nicht enthalten sind die Zeiten die be- noetigt wurden um den Quellcode physi- kalisch auf den jeweiligen Zielrechner zu transferieren. Zudem wurde von den Kollegen uebereinstimmend erklart, dass die obigen Zeiten nur notwendig waren, da die Abweichungen im einzelnen erst ermittelt werden mussten. Fuer die da-

Aenderung Version	a) Modulname	b) Task 'MAIN'	c) d) alphanum. E/A mit Standardperipherie	e) holen von Uhrzeit	f) Character- verarbeitung
Siemens 330	MODULE;	-	•kein open •Einleseformat (60)A(1)	•NOW •muß spezifi- ziert werden.	-
AEG 80-20	MODULE;	-	•kein open •Einlesen ganz abgeaendert.	NOW	Fehler ent- deckt in akt. PEARL-Version
EPR 1300	MODULE: 'name'; 1) 2)	erforder- lich	• open erforderlich •I/O Signalreaktion vorsehen •Einleseformat A(60)	DAYTIME	-
PDP11	MODULE: ( 'name' ); 1) 3)	erforder- lich	• open erforderlich •Einleseformat A(60)	•NOW •muß spezifi- ziert werden	•CHAR(i) mit i:= Konstante
Verwendung ( Grund )	1) nicht Basic- PEARL 2) name kann entfallen bei nur einem Modul 3) Full-PEARL	wegen Implemen- tierung	Implementierung der Dations	Systemfunktion	nicht Basic- PEARL

Tabelle 1:  
Notwendige Aenderungen  
im PEARL-Problemtteil

nach ausgefuehrten Anpassungen anderer PAP-Module sind die oben angegebenen Zeiten etwa zu halbieren.

#### 4. SCHLUSSBETRACHTUNG

-----

Wie oben gezeigt, sind bei der Uebertragung von PEARL-Programmen auf andere Zielrechner trotz weitestgehender Einhaltung von Basic-PEARL und identischer Aufgabenstellung Aenderungen im Problemteil notwendig. Der Aenderungsaufwand ist jedoch alles in allem gering und betrifft in der Hauptsache die alphanumerische E/A c) d), die Systemfunktionen fuer die Ermittlung von Uhrzeit und Datum e), die Character-Verarbeitung c) f) und die Namenskonvention fuer die Start-Task a).

#### SCHRIFTTUM

-----

- /1/ Welfonder E., Th. Röhrich:  
 Uebersicht ueber die PEARL-Anwender-  
 Programmbibliothek  
 Vortrag bei PEARL-Tagung '82

- /2/ Welfonder E., Th. Roehrich, H. Sternad:  
 Aufbau und Erprobung einer modular  
 strukturierten portablen Basic-PEARL  
 Programmbibliothek  
 PEARL Rundschau, Bd.1, Nr.3 Nov.1980

- /3/ Welfonder E., Th. Roehrich:  
 Meldungs- und Protokolliersystem der  
 PEARL-Anwender Programmbibliothek  
 PEARL Rundschau, Bd.2, Nr.6 Dez.1980

- /4/ Th. Roehrich, E. Welfonder, M. Alt:  
 Kurzbeschreibung der PEARL-Anwender  
 Programmbibliothek

#### ANSCHRIFT DES AUTORS

-----

Roehrich, Thomas

Abteilung Stromerzeugung und  
 Automatisierungstechnik am  
 Institut fuer Verfahrenstechnik  
 und Dampfkesselwesen  
 Pfaffenwaldring 9  
 7000 Stuttgart 80  
 Tel. 0711/685-6203

