

PEARL-Ausbildung im Studiengang Elektrotechnik an der Universität (GH) Paderborn

Dr.-Ing. Bernd Reißweber, Paderborn

Zusammenfassung:

Die Ausbildung besteht aus einer zweisemestrigen Vorlesung und Übungen mit einem BBC-PEARL-System. Vorlesungsinhalt und die Art, wie in der Vorlesung vorgegangen wird, werden beschrieben. Von den Übungsaufgaben wird die Führung eines kleinen Prozesses ausführlicher erläutert, und es wird auf die Erfahrungen mit dem BBC-PEARL-System bei der Durchführung der Übungen eingegangen. Den Abschluß bilden Überlegungen, wie man die PEARL-Ausbildung intensiver und nachhaltiger gestalten könnte.

1. Einführung

Für die PEARL-Ausbildung stehen innerhalb der Universität Paderborn zwei Systeme zur Verfügung:

- a) Siemens 330 mit Basis-PEARL im Fachbereich Mathematik-Informatik
- b) BBC DP 1000 mit BBC-PEARL bzw. PAS2 im Fachgebiet Prozeßautomatisierung des Fachbereichs Elektrotechnik.

Auf dem Siemens-System werden die Studenten des sechssemestrigen Studienganges, auf dem BBC-System die Studenten des achtsemestrigen Studienganges ausgebildet. Hier soll über die Ausbildung am BBC-Rechner berichtet werden.

2. Art der Veranstaltung

Die Studenten sollen in die Lage versetzt werden, ein vollständiges PEARL-Quellprogramm schreiben und am Prozeßrechner zur fehlerfreien Ausführung bringen zu können. Dazu werden in einer normalen Vorlesung

die einzelnen Sprachelemente von PEARL und die Regeln zum Schreiben eines PEARL-Programmes behandelt. Zusätzlich werden Übungsaufgaben ausgegeben. Die Studenten müssen das zur Lösung der Aufgabe notwendige Programm entwickeln und in Übungsstunden am Rechner austesten.

Die gesamte Veranstaltung wird als Wahlpflichtfach durchgeführt, d.h. die Studenten nehmen freiwillig daran teil. Bisher waren es pro Semester etwa 15 - 25 Hörer aus dem 5. oder höheren Semestern. Als Pflicht-Programmiersprache haben diese Studenten bereits Assembler bzw. seit letztem Jahr FORTRAN gelernt. Diese Vorbildung wirkt sich stark auf das Verhalten der Studenten in Vorlesung und Übung aus, so daß man insbesondere in der Ausgestaltung der Vorlesung darauf Rücksicht nehmen muß.

3. Aufbau der Vorlesung und Übung

Im Fachgebiet Prozeßautomatisierung beschäftigen wir uns bevorzugt mit dem Entwurf und der Realisierung von Regelungs- und Steuerungsalgorithmen und mit der Simulation dynamischer Systeme. Dazu muß man - was die Algorithmen anbelangt - numerisch Integrieren, Lösungen von Differentialgleichungen bestimmen, Nullstellen suchen und Matrizen manipulieren. Hinzu kommen Echtzeitaufgaben wie paralleler Lauf von verschiedenen Regelalgorithmen, gleichzeitiger Dialog über Bedienschreibmaschine, gleichzeitiges Erzeugen von Protokollen aus dem Drucker. Auf diese Aufgaben konzentrieren sich dann auch die PEARL-Vorlesung und die Übung.

Die Vorlesung läuft über zwei Semester. In PEARL I (3 Wochenstunden im Wintersemester)

wird der algorithmische Teil von PEARL einschließlich Ein- und Ausgabe über Standardperipherie gebracht.

Die Vorlesung beginnt mit einer kurzen Beschreibung des Prozeßrechners und seiner Peripherie und einer Beschreibung der Systemprogramme, die bei der Programmentwicklung und Programmausführung notwendig sind.

Nächster Punkt ist die Formulierung einer Aufgabe aus der numerischen Mathematik. Der Integralsinus

$$Si(x) = \int_0^x \frac{\sin t}{t} dt$$

soll für eine Reihe von x-Werten durch verschiedene Integrationsverfahren und durch Reihenentwicklung numerisch berechnet werden. Durch einige Zusatzforderungen ist dieses Beispiel so angelegt, daß man die wesentlichen Sprachelemente, die PEARL im algorithmischen Teil bietet, damit demonstrieren kann.

Die Lösung dieser Aufgabe wird zunächst in Form von Programmablaufplänen dargestellt. Gleichzeitig werden Kopien des vollständigen PEARL-Quellprogrammes an alle Hörer verteilt.

Dann werden in der Vorlesung nacheinander die Datenelemente, Ausdrücke, Anweisungen, Ein- und Ausgabe über Standardperipherie, Prozeduren und ON-Blöcke behandelt. Dabei wird immer in drei Schritten vorgegangen:

- a) An Hand des Integralsinus-Beispiels wird gezeigt, warum das gerade betrachtete Sprachelement überhaupt notwendig ist und wozu man es benutzen kann.
- b) Die allgemeinen Regeln für dieses Sprachelement werden beschrieben. Dabei werden gut überschaubare Formulierungen angestrebt. Auf die Metasprache wird weitgehend verzichtet, und es werden auch nicht alle Möglichkeiten, die nach Sprachbeschreibung [1] vorhanden sind, behandelt.
- c) Für jede Regel werden mehrere Beispiele hinzugefügt.

Parallel zu den letzten Vorlesungsstunden werden drei Übungsaufgaben gestellt, nämlich Nullstellensuche mit Newton-Verfahren und Regula falsi, Lösung einer Differential-

gleichung 1. Ordnung mit Runge-Kutta-Verfahren, Matrizenoperationen (Dialogprogramm zum Ein- und Ausgeben von Matrizen, Prozeduren für Matrizenmultiplikation und Matrizeninversion).

In PEARL II (2 Wochenstunden im Sommersemester) werden die Echtzeitelemente von PEARL behandelt.

Die Vorlesung beginnt wieder damit, daß eine Aufgabe formuliert und das dazugehörige PEARL-Quellprogramm verteilt wird. Da bei den Studenten im 6. Semester nur wenig Kenntnisse zum Thema Prozeßautomatisierung vorausgesetzt werden können, wird eine einfache Meßwerterfassungsaufgabe konstruiert.

Von einem schnelleren und einem langsameren Teilprozeß sollen in verschiedenen Zeitrastern analoge Meßwerte aufgenommen und je ein Wirkungsgrad daraus berechnet werden. Wenn eine bestimmte Anzahl von Meßwerten vorliegt, soll die relative Häufigkeit des Auftretens eines bestimmten Wertes von den beiden Wirkungsgraden ermittelt und als Druckerbild (Bild 1) ausgegeben werden. Außerdem sollen in einem festen Zeitraster die aktuellen Meßwerte auf dem Drucker ausgegeben werden.

Auch diese Aufgabe ist so angelegt, daß möglichst viele Sprachelemente, die in PEARL II behandelt werden sollen, darin vorkommen. Es werden deshalb in der gleichen Art wie in PEARL I die Themenkomplexe Ein- und Ausgabe über Prozeßperipherie und Multitasking behandelt.

So kann man an dem Meßwerterfassungsbeispiel zeigen, daß eine Echtzeitaufgabe aus mehreren asynchron nebeneinander ablaufenden Teilaufgaben besteht und daß es sinnvoll ist, jede Teilaufgabe durch eine Task mit geeigneter Priorität zu realisieren. Es ergibt sich die Notwendigkeit, Tasks starten, beenden und unterbrechen zu können. Auch die Wichtigkeit von SemaS wird an diesem Beispiel deutlich, z.B. wird der Schnelldrucker von zwei Protokollierungstasks benutzt, und es soll natürlich die Protokollzeile mit den aktuellen Meßwerten nicht mitten im Druckerbild stehen.

In der PEARL II-Übung wird eine Prozeßführungsaufgabe, die im nächsten Abschnitt beschrieben wird, gestellt.

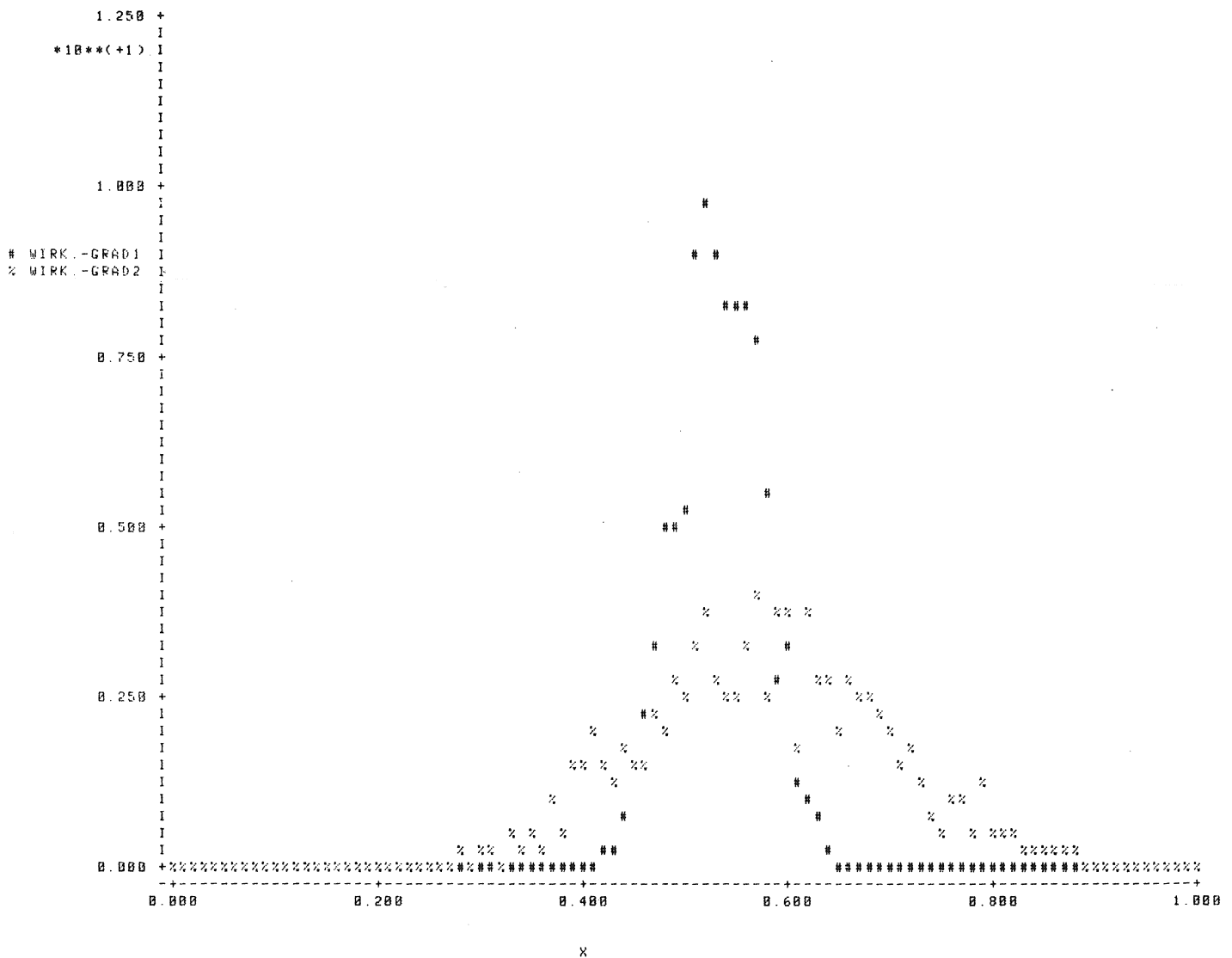


Bild 1. Relative Häufigkeit des Auftretens von bestimmten Werten des Wirkungsgrades 1 und 2 als "Druckerbild" auf einem Schnelldrucker ausgegeben

4. Erfahrungen mit dem BBC-PEARL-System bei den Übungen

In der Vorlesung wird für jede Übungsaufgabe ein Aufgabenblatt, das neben der Aufgabenstellung auch theoretische Grundlagen und Lösungshinweise enthält, verteilt. Die Studenten müssen das vollständige PEARL-Programm entwickeln und in Lochkarten ablochen. An festgesetzten Terminen haben die Studenten einen ganzen Nachmittag die Möglichkeit, ihr Programm zu übersetzen und es zum fehlerfreien Laufen zu bringen.

Dies beginnt zunächst einmal für jedes Programm mit einem, normalerweise jedoch mehreren Compilerläufen, bis der Compiler keine Fehler mehr meldet.

Der Compiler arbeitet absolut zuverlässig, er findet jeden Fehler und macht detaillierte Angaben, die die Fehlerursache sehr stark einkreisen. Zwangsläufig kann ein Fehler aber auch Folgefehler erzeugen. So entstehen in den Übungen gar nicht selten 1, 2 oder 3 Seiten lange Fehlermeldungen. Da hat man natürlich keine Lust, jeder Fehlermeldung nachzuforschen, man wird die größten Fehler beseitigen und durch einen neuen Compilerlauf die Anzahl der Fehlermeldungen zu reduzieren versuchen.

Auch beim Suchen nach logischen Fehlern, die kein Compiler finden kann, wird natürlich oft so verfahren, daß man, sobald der erste Fehler gefunden ist, erst einmal neu kompiliert und schaut, ob das Programm jetzt läuft.

Für den Übungsbetrieb ist es also bei der großen Anzahl von Compilerläufen sehr wichtig, daß der Compiler schnell arbeitet und daß er möglichst wenig Papier verbraucht. Damit sieht

es bei dem BBC-System nicht schön aus. Für die einfache Übungsaufgabe "Nullstellensuche" braucht er etwa 5 Minuten. Außerdem erzeugt er neben den notwendigen Ausdrucken zusätzliche 8 Seiten Papier, die für Studenten, aber auch für Mitarbeiter des Fachgebiets vielleicht 10 Zeilen Information enthalten.

Als Beispiel für eine Übungsaufgabe soll die Prozeßführungsaufgabe aus PEARL II etwas ausführlicher beschrieben werden. Ein kleiner Prozeß, bestehend aus 3 Teilprozessen, wird auf einem Analogrechner realisiert und soll durch den Prozeßrechner geführt werden (Bild 2). Der eine Teilprozeß soll durch einen PI-Regler, der andere durch ein Zweipunktglied mit Hysterese geregelt werden. Beide Regler erhalten als Eingangsgrößen die analogen Regelgrößen x_1 und x_2 und die analogen Sollgrößen w_1 und w_2 . Die Sollgrößen werden als Rechteckschwingung vorgegeben und von einem Funktionsgenerator erzeugt. Aufgabe der Regler ist es, dafür zu sorgen, daß die Regelgröße x_1 möglichst gut mit der Sollgröße w_1 übereinstimmt und daß die Regelgröße x_2 möglichst gut mit der Sollgröße w_2 übereinstimmt (Bild 3). Dazu gibt der PI-Regler die analoge Stellgröße y_1 auf

den Teilprozeß 1 und der Zweipunktregler die digitale Stellgröße y_2 auf den Teilprozeß 2.

Alle Signale, die zwischen Rechner und Prozeß ausgetauscht werden, können auf einem Speicheroszillograph sichtbar gemacht und mit einem Vierkanalschreiber aufgezeichnet werden (Bild 3).

Besonders der PI-Algorithmus kann die ihm gestellte Aufgabe gut oder schlecht erfüllen, je nachdem, wie die Parameter im Algorithmus eingestellt sind. Deshalb sollen diese Parameter während des Betriebs im Dialog über Bedienschreibmaschine geändert werden können. Dazu braucht man, da jeder Regelalgorithmus durch eine Task realisiert wird, die dritte Task. Hinzu kommt eine vierte Task, die eine Meldung auf der Bedienschreibmaschine ausdruckt, wenn der Ausgang von Teilprozeß 3 einen Grenzwert überschreitet und ein Grenzwertmelder einen Interrupt auslöst.

Da das Aufgabenblatt einige Lösungshinweise enthält und da die einzelnen Tasks relativ kurze und nahezu geradlinig durchlaufene Programmstückchen sind, treten beim Austesten dieser Übungsaufgabe nur relativ wenige logische Fehler auf. Die meisten

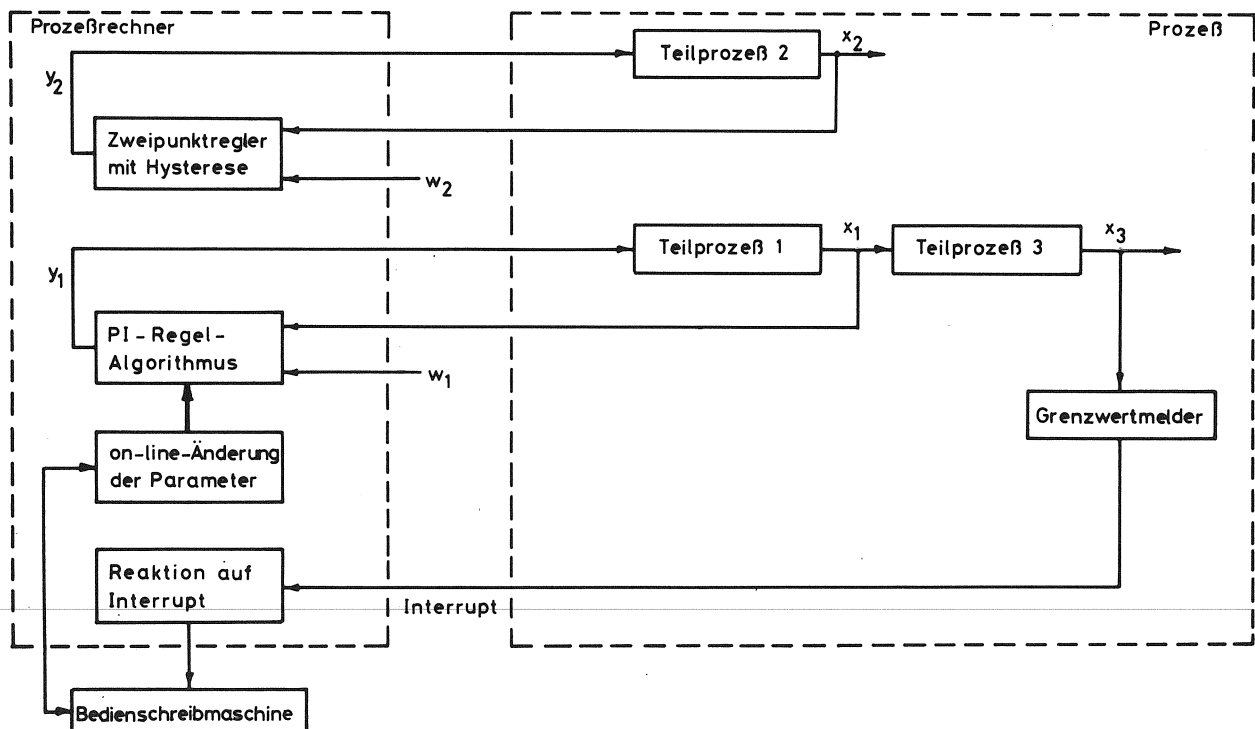


Bild 2. Führung eines einfachen Prozesses

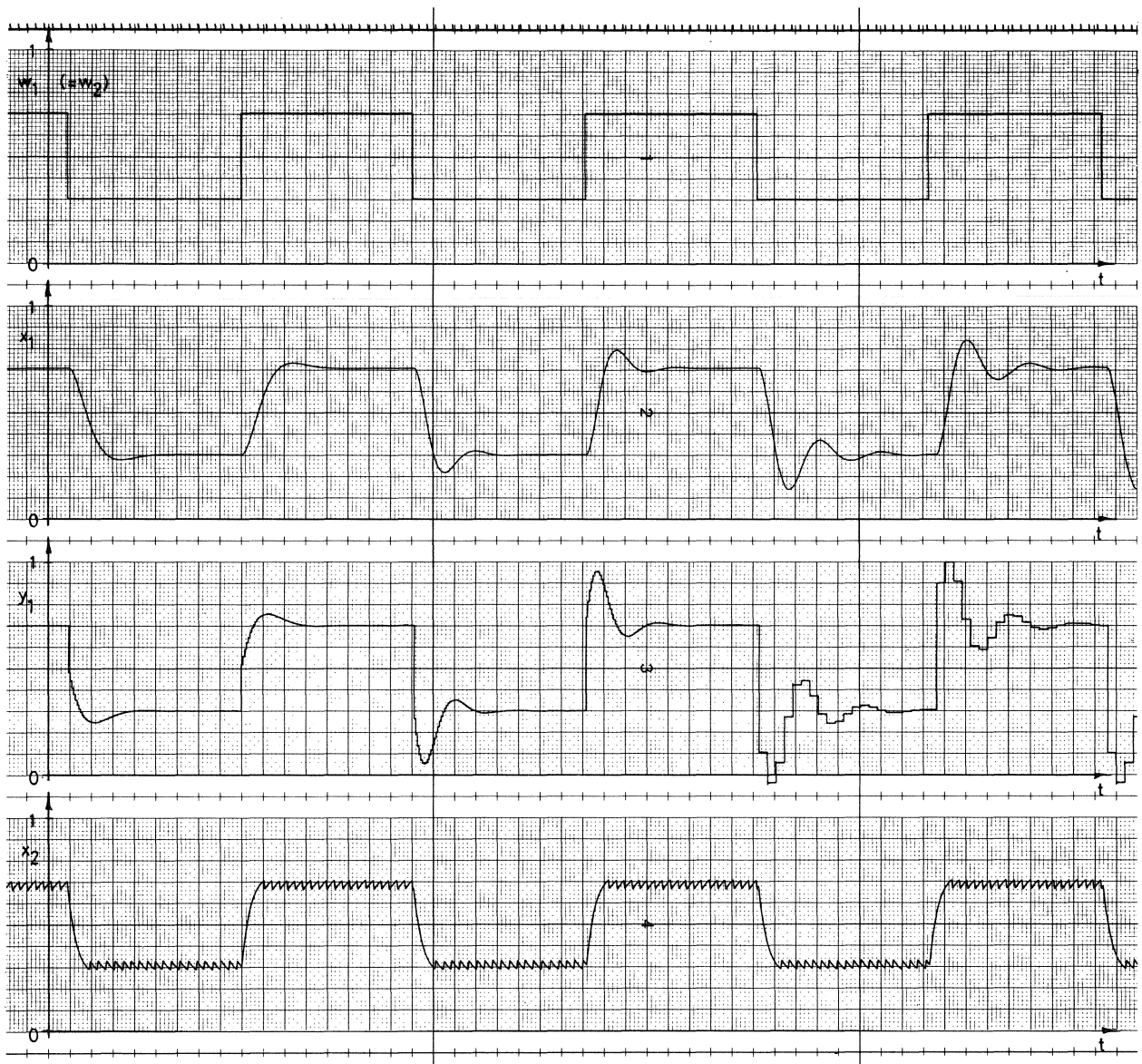


Bild 3. Zeitlicher Verlauf der interessierenden Größen bei on-line-Änderung der Parameter des PI-Algorithmus

Fehler sind von der Art, wie sie bereits in den Übungen zu PEARL I aufgetreten sind, betreffen also z.B. die Ein- und Ausgabe über Standardperipherie. Mit dem Multitasking gibt es so gut wie gar keine Schwierigkeiten. Fehler treten eher bei der Ein- und Ausgabe über Prozeßperipherie auf, da hier PAS2 recht deutlich von der Norm abweicht und einige Umständlichkeiten enthält.

Insgesamt betrachtet kann man sich auf das BBC-PEARL-System bei den Übungen vollständig verlassen. Systembedingte Fehler treten nicht auf. Letztlich kam noch jedes Programm eines Studenten zum fehlerfreien Laufen. In hartnäckigen Fällen hilft da die Sedientask, die ein gutes Testhilfsmittel ist.

5. Fortsetzung der Ausbildung

Von den Studenten wird immer wieder beklagt, daß sie die in der PEARL-Vorlesung und -Übung erworbenen Kenntnisse innerhalb der Hochschule nicht nutzen können, weil kein PEARL-System frei zugänglich ist. Außerdem zeigen die Übungen in PEARL II, daß die Studenten die in PEARL I erlernten Fähigkeiten wohl schon vergessen haben, wenn sie ein halbes Jahr später die PEARL II-Übung absolvieren.

Um die Ausbildung also intensiver und nachhaltiger werden zu lassen, muß den Studenten die Möglichkeit gegeben werden, nach Wunsch auf ein PEARL-System zugreifen zu können.

Unser Prozeßrechner eignet sich dazu nicht, da die Bedienung der zahlreichen verwendeten Systemprogramme so kompliziert ist, daß man nicht jeden Studenten darin einführen kann. Deshalb werden bei uns zur Zeit nur die Studenten, die den Prozeßrechner im Rahmen ihrer Diplom- oder Studienarbeit benutzen, in die Lage versetzt, das System selbständig zu bedienen.

Für die Ausbildung wünschenswert ist deshalb ein PEARL-System, das auf einem Mikroprozessor-Entwicklungssystem mit Floppy-Disk eigenständig lauffähig ist. Der Compiler soll ebenfalls auf dem Mikrorechner laufen, er soll möglichst schnell sein und minimalen Papierverbrauch haben. Das gesamte System soll möglichst einfach zu handhaben sein, so daß ein Student an Hand einer Bedienungsanleitung und nach einer Einführung innerhalb der Übungen das System selbständig bedienen kann.

Schrifttum

[1] BBC-PEARL-Subset, Sprachbeschreibung, Ausgabe 6.0, Juli 1977

Anschrift des Autors:

Reißenweber, Bernd
Universität (GH) Paderborn
Fachbereich Elektrotechnik
Fachgebiet Prozeßautomatisierung

Pohlweg 47 - 49

4790 Paderborn

Telefon: (05251) 60 30 05