

# Echt Zeit

Nr. 7, Februar 2018

Mitteilungen  
des GI/GMA/ITG-Fachausschusses  
Echtzeitsysteme



GESELLSCHAFT FÜR INFORMATIK E.V.



VDI/VDE-Gesellschaft  
Mess- und Automatisierungstechnik

**ITG** INFORMATIONSTECHNISCHE  
GESELLSCHAFT IM VDE

## Impressum

**Herausgeber** GI/GMA/ITG-Fachausschuss Echtzeitsysteme  
<http://www.real-time.de>

**Sprecher** Prof. Dr.-Ing. habil. Herwig Unger  
FernUniversität in Hagen  
Lehrstuhl für Kommunikationsnetze  
58084 Hagen  
[herwig.unger@fernuni-hagen.de](mailto:herwig.unger@fernuni-hagen.de)

**Stellvertreter** Prof. Dr. Dieter Zöbel  
Universität Koblenz-Landau  
Institut für Softwaretechnik  
56016 Koblenz  
[zoebel@uni-koblenz.de](mailto:zoebel@uni-koblenz.de)

**Redaktion** Prof. Dr.-Ing. habil. Herwig Unger  
Dipl.-Ing. Jutta Düring  
FernUniversität in Hagen  
58084 Hagen  
[echtzeit@fernuni-hagen.de](mailto:echtzeit@fernuni-hagen.de)

**ISSN** 2199-9244

Redaktionell abgeschlossen am 14. Februar 2018

### Einreichung von Beiträgen:

Alle Leserinnen und Leser sind aufgerufen, das Mitteilungsblatt auch zukünftig durch Beiträge mit zu gestalten, um den Informations- und Meinungs austausch zwischen allen an den Fragen der Echtzeitprogrammierung Interessierten zu fördern.

### In dieser Ausgabe:

- 1 Neuwahl des Sprechers des Fachausschusses Echtzeitsysteme
- 2 Echtzeit 2018 – Echtzeit und Sicherheit: Call for Papers
- 3 Einreichung von Vorschlägen zum Graduiertenwettbewerb 2018
- 4 11th GI Conference on Autonomous Systems 2018
- 5 Veranstaltungen mit Unterstützung des Fachausschusses
- 6 PEARL in der Ausbildung:
- 7 Entstehung der Programmiersprache PEARL

# 1 Neuwahl des Sprechers des Fachausschusses Echtzeitsysteme

Jutta Düring, Fachausschuss Echtzeitsysteme

Der bisherige Sprecher Prof. Dr. Dr. Wolfgang A. Halang wurde zum 1. April 2017 pensioniert; er trat deshalb als Sprecher des Fachausschusses und Mitglied der Fachausschussleitung zurück.

Am 16. November 2017 wurde in geheimer Wahl Prof. Dr.-Ing. habil. Herwig Unger einstimmig ohne Enthaltungen zum neuen Sprecher des Fachausschusses Echtzeitsysteme gewählt. Prof. Dr. Dieter Zöbel bleibt stellvertretender Sprecher.



Der bisherige Sprecher Prof. Halang (links) und der neue Sprecher Prof. Unger (rechts)

Die Mitglieder der Fachausschussleitung sind:

- Prof. Dr. Reinhard Baran, Hamburg
- Prof. Dr. Marcel Baunach, Graz
- Barbara Beenen, Lüneburg
- Prof. Dr. Juliane Benra, Wilhelmshaven
- Volker Cseke, Wedemark
- Prof. Dr. Roman Gumzej, Maribor
- Prof. Dr. Hans Heinrich Heitmann, Hamburg
- Prof. Dr. Jürgen Jasperneite, Lemgo
- Prof. Dr. Rainer Müller, Furtwangen
- Marcel Schaible, München
- Prof. Dr. Gudrun Schiedermeier, Landshut
- Prof. Dr.-Ing. habil. Herwig Unger, Hagen (Sprecher)
- Prof. Dr. Dieter Zöbel, Koblenz (Stellvertretender Sprecher)

## 2 Echtzeit 2018 – Echtzeit und Sicherheit: Call for Papers

Jutta Düring, Fachausschuss Echtzeitsysteme

Homepage: <http://www.real-time.de/CfP.html>

Die Tagung „Echtzeit“ findet in diesem Jahr am **15. und 16. November 2018** wie gewohnt in Boppard am Rhein statt. Unser diesjähriges Leitthema lautet:

### „Echtzeit und Sicherheit“

Zu folgenden und benachbarten Themen werden Vorträge über Methoden, praktischen Einsatz, Erfahrungen und Ausblicke erbeten. Exponate sind immer willkommen.

- Echtzeitfähigkeit
- Funktionale Sicherheit
- Datensicherheit und Datenschutz
- Sichere Datenkommunikation
- Industrie 4.0 und Internet der Dinge
- Zuverlässigkeit
- Ambient Assisted Living
- Autonomes Fahren
- Smart Grids
- Aktuelle Anwendungen
- Ausbildung

Stichtag für die Vortragsanmeldung ist **Mittwoch, der 22. April 2018**.

## 3 Einreichung von Vorschlägen zum Graduiertenwettbewerb 2018

Jutta Düring, Fachausschuss Echtzeitsysteme

Homepage: <http://www.real-time.de/preise/grad-bedingung.html>

Der Graduiertenwettbewerb des Fachausschusses wird bereits seit 2007 durchgeführt. Bis zu drei herausragende Diplom-, Bachelor- und Masterarbeiten im Bereich Echtzeitsysteme werden mit je 500,-€ prämiert. Erstmals in diesem Jahr können auch **Doktorarbeiten** vorgeschlagen werden.

Die Abschlussarbeiten sollten zum Zeitpunkt der Einreichung abgeschlossen und bewertet, aber nicht älter als ein Jahr sein. Elektronisch vorzulegen sind:

- die eingereichte Abschlussarbeit,
- eine Zusammenfassung der Arbeit,
- Angaben zur Person der Verfasserin / des Verfassers (CV)
- sowie eine **Stellungnahme der Betreuerin / des Betreuers**.

Das Ende der Vorschlagsfrist ist **Sonntag, der 15. April 2018**.

## 4 11th GI Conference on Autonomous Systems 2018

Jutta Düring, Fachausschuss Echtzeitsysteme



Der GI-Arbeitskreis Echtzeitkommunikation beschäftigt sich mit Echtzeitfragen des Internets und sozialer Netze. Die von ihm veranstaltete Konferenz „Autonome Systeme“ findet vom

**28. Oktober – 02. November 2018**

in Cala Millor auf Mallorca statt. Der Tagungsband erscheint in den „Fortschritt-Berichten“ des VDI-Verlages.

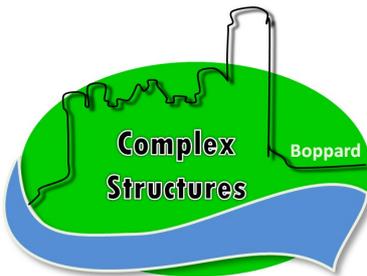
Homepage: <http://www.confautsys.org>

## 5 Veranstaltungen mit Unterstützung des Fachausschusses

Jutta Düring, Fachausschuss Echtzeitsysteme

### 3rd GI Workshop on Complex Structures

**25. – 28. März 2018, Hotel Ebertor in Boppard am Rhein**



This workshop is intended as a forum for presenting methods to master the growing amount of information by new — not only mathematical — methods and analyses, models and simulations as well as by algorithms to recognise, consider, predict or influence structural changes. Since methods seems to be similar in different areas of application and research, it is the goal of the workshop to bring practitioners and scientists from different areas together, discuss the same or (apparently) completely different problems and find solutions for their problems by interdisciplinary discussions.

Homepage: <http://www.fernuni-hagen.de/kn/complex-structures/>

Der Workshop ist offen für eine Teilnahme von Hochschullehrern und Doktoranden.

### 9. Jahreskolloquium „Kommunikation in der Automation“

**21. November 2018, Lemgo**



Das 9. Jahreskolloquium „Kommunikation in der Automation – KomMA“ der beiden Institute, dem Institut für industrielle Informationstechnik (inIT) der Hochschule OWL und dem Institut für Automation und Kommunikation (ifak) e.V. in Magdeburg findet in diesem Jahr in Lemgo statt. Das Kolloquium

ist ein Forum für Wissenschaft und Industrie im deutschsprachigen Raum für alle technisch/wissenschaftlichen Fragestellungen rund um die industrielle Kommunikation.

Homepage: <http://www.jk-komma.de>

## 6 Aktualisierung der Übersicht „PEARL in der Ausbildung“

Jutta Düring, Fachausschuss Echtzeitsysteme

Homepage: <http://www.real-time.de/archiv-irt/ausbildung.html>

Am Institut für Regelungstechnik der Universität Hannover unter Prof. Wilfried Gerth wurde eine Übersicht der angebotenen PEARL-Anteile in der Ausbildung an Universitäten, Fachhochschulen und Berufsakademien geführt. Nach der Emeritierung von Prof. Gerth und Umstrukturierung des Instituts wurde die Übersicht nicht mehr weitergeführt, weshalb diese den Stand von 2008 hat und nach 10 Jahren dringend einer Aktualisierung bedarf. Professoren haben die Hochschule gewechselt oder sind emeritiert, Studiengänge wurden eingestellt, Fachhochschulen haben sich umbenannt etc..

Institution	Fachbereich / Studienrichtung	Dozentin / Dozent	Name der Veranstaltung	PEARL-Anteile
Slovakische Technische Universität Bratislava	Elektrotechnik und Informatik/ Automatisierungssysteme	B. Hruz	Projektierung der automatisierten Systeme	Echtzeit-Anweisungen; Übungen: Steuerung eines simulierten Robotersystems
Universität Bremen	Elektrotechnik / Automatisierungstechnik	G. Thiele	Realzeitsoftware-Entwurf I,II	PEARL-orientiert
Universität -GH- Duisburg	Elektrotechnik	U. Maier	Prozessoren und Steuerungen / Prozess-automatisierung	Echtzeit - Datenverarbeitung mit PEARL - Beispielen
TU Dresden	Elektrotechnik / Automatisierungstechnik	P. Rieger	Prozessleittechnik	Sprachübersicht, Praktikum
Universität Erlangen-Nürnberg	Informatik	P. Holleczeck, H.-J. Schneider	Programmierung von Prozessrechnerproblemen mit PEARL	Sprachumfang, Komponenten einer Programmierumgebung für PEARL
Universität -GH- Essen	Maschinentechnik/ Automatisierungstechnik -Verfahrenstechnik	R. Tracht	Steuerungstechnik	Prozessrechentechnik PEARL - orientiert

Abbildung 1: Auszug aus Übersicht von 2008

### Bitte um Aktualisierung

Ich rufe alle Leserinnen und Leser auf, Lehrveranstaltungen und Praktika mit Echtzeit- und/oder PEARL-Anteilen zu melden; wenn möglich mit URL oder Ansprechpartner/in.

E-Mail: [echtzeit@fernuni-hagen.de](mailto:echtzeit@fernuni-hagen.de)

# 7 Entstehung der Programmiersprache PEARL

Rainer Müller, Hochschule Furtwangen, [mueller@hs-furtwangen.de](mailto:mueller@hs-furtwangen.de)

## 7.1 Sprachvorschlag, Pilot- und Industriimplementierungen

PEARL ist aus einer Reihe von Entwicklungen und Strömungen entstanden, die bis ca. 1967 rück verfolgbar sind [1]. So gab es beispielsweise PAS1 — eine Programmiersprache für Prozessautomatisierung von der Fa. Brown, Boveri & Cie und auch frühe Ideenskizzen für eine Sprache zur Steuerung von kerntechnischen Anlagen [3]. 1968 startet dann das Bundesministerium für Bildung und Wissenschaft (BMBW) ein Projekt unter dem Titel „NDV13“ mit dem Ziel, eine Experimentorientierte Sprache (EXOS) zu erarbeiten.

Diese verschiedenen Ansätze kamen Anfang 1970 zusammen und führten zu einem Arbeitstitel „PEARL“ [2].

In der ersten Hälfte der 1970-er Jahre ging die Sprachentwicklung von PEARL stark voran, so dass 1973 die erste Definition veröffentlicht werden konnte [4]. Hier waren Mitarbeiter der Firmen Werum, BBC, Siemens, UNICOMP, AEG-Telefunken, ESG, der Forschungsinstitute Kernforschungsanlage Jülich und Hahn-Meitner-Institut Berlin sowie der Universitäten Heidelberg, Erlangen-Nürnberg und Karlsruhe beteiligt.

## 7.2 Pilot- und Industriimplementierungsphase

Ab ca. 1970 wurden Rechner zumindest so weit erschwinglich, dass aufwändige Steuerungsaufgaben immer stärker rechnergesteuert realisiert werden konnten. Nach der Einigung über die Sprachdefinition wurden Sprachsysteme für verschiedene Plattformen entwickelt. Die nachstehende unvollständige Liste soll einen Eindruck über die Breite der beteiligten Firmen sowie der unterstützten Rechnerhardware geben:

**BBC für DP1000** wird in [6] als erstes verfügbares PEARL-System erwähnt.

**ASME PEARL** Die Arbeitsgemeinschaft „Stuttgart – München – Erlangen“, mit Vertretern der Universitäten Stuttgart und Erlangen-Nürnberg sowie der Firmen ESG und GPP in München, hat ein rechnerunabhängiges Kompiliersystem für PEARL entwickelt. Es lief auf der Rechnern der Projektpartner (AEG 6050 und 6080, sowie Siemens 306 und 404/3) [6].

**Krupp-Atlas** Die Firma Krupp Atlas Elektronik bietet für ihr System 1300 eine Programmierumgebung für PEARL an [7].

**Siemens Systeme 300** Exemplarisch sei hier ein Vortrag bezüglich PEARL300 auf Siemens Prozeßrechnern über den Aufbau des PEARL Übersetzungssystems herausgegriffen [8] sowie ein Marketingartikel über PEARL auf Siemens R20 und R30 Systemen erwähnt [9].

**Dornier MUDAs 432 und AEG 80-20** [32]

**Werum-PEARL** stand für ein portables PEARL-System, welches auf diversen Systemen verfügbar war: COBRA 530, DEC LSI 11, DEC VAX 11, NOVADATA, PCS QU 68000, Siemens 7.xxx (BS2000), SISCO MB8000 und Zilog Z8000 [28].

**RTOS-UH** wurde bereitgestellt für MC68000 (siehe Motorola MC68000) und PowerPC-Systeme. Es umfasst ein integriertes Entwicklungssystem mit Editor, Compiler und Ladebinder [12].

**Z80** Über Crosscompilierung auf VAX 780 und Cyber 173/174 werden PEARL-Programme für den Z80-Prozessor in Assemblerquellcode vorübersetzt. Die weiteren Übersetzungsstufen erfolgen mit den Werkzeugen am Z80-System unter CP/M. Zur Laufzeit wird ein spezielles PEARL-Betriebssystem (PBS) eingesetzt [33].

**Motorola MC68000** Für diesen Prozessor gab es zwei PEARL-Systeme. Einerseits das von der Universität Erlangen [31], andererseits das am Institut für Regelungstechnik der Universität Hannover entwickelte RTOS-UH [11, 12, 13, 14, 15], welches auch auf ATARI ST portiert wurde [16]. Die Zeitschrift c't entwickelte sogar eine extra Platine auf Basis dieses Prozessors als Zielplattform für dieses PEARL-System [20].

## 7.3 Anwendung, Weiterentwicklung und Standardisierung

### 7.3.1 Anwendungsbeispiele

In [10] wurde berichtet, dass im Jahr 1979 etwa 140 verschiedene PEARL-Anwendungen im Einsatz waren — ein großer Teil davon im industriellen Umfeld. Die Anwendungsgebiete Metallverarbeitung, Energieverteilung, Energieerzeugung, Chemie, Wasserversorgung, Versandhandel sowie andere Dienstleistungen (Fernsehen, Verkehr, Raumfahrt) wurden genannt.

Die Tagungsbände der Fachtagung von PEARL-Verein und GI-Fachgruppe 4.4.3 geben einen Eindruck über das Anwendungsspektrum von PEARL. Einige Schlagworte mit PEARL-Einsatz aus den Tagungsbänden der Jahre:

**1983 [21]** Steuerung von Schienenverkehr, Datenhaltung in Betriebsleitsystem, Waffeneinsatzsystem ARES, hohe Systemverfügbarkeit in Rechnerverbundsystemen, Bewegungsprogrammierung von Robotern

**1984 [22]** Steuerung von Produktionsleitständen, On-Line Simulation, Fördertechnik, 3D-Visualisierung, Test auf PEARL-Sprachebene, Steuerung von kernphysikalischen Experimenten, Filetransfer

**1985 [23]** Steuerung von kernphysikalischen Experimenten, Wasserstraßenüberwachung, Waffensteuerungen, Simulation von Blockheizkraftwerken und Antriebssystemen, Datenbanken, Rechnerkommunikation, ausfallsichere Datenerfassung

**1986 [24]** Expertensysteme, Fördertechnik, Lagertechnik, Bildverarbeitung

**1987 [25]** Rechnerkopplung,

**1988 [26]** Erfassung und Verwaltung von Störmeldungen, Prozeßleitsysteme, Funknetzüberwachung, Netzleittechnik

**1991 [27]** Paketsortierung, Realisierung eines Steuerpults, digitale Regelung

Weitere Veröffentlichungen aus dem Industrieumfeld sind in diesem Bereich recht selten, da hier oft Geheimhaltungsinteressen entgegenstehen. Ein Beispiel für eine verteilte PEARL-Anwendung ist in [42] von 1986 gegeben. 18 Tasks mit ca. 4.500 Quellcodezeilen wurden auf vier gekoppelte Z80-Rechner verteilt, so dass Telefonabrechnungsdaten redundant erfasst werden konnten.

Im Bereich der Experimentalphysik gab es auch Ansätze, langsame unter anderem thermisch bedingte Drift des Verhaltens von Elektronik auf Zeit- und Amplitudenskala zu erkennen. Auf diese Weise wurden daraus resultierende Meßfehler bei wochenlangen Messungen frühzeitig erkannt und gemeldet [43, 44]. Notwendige Korrekturen konnten so schon während der Datenerfassung durchgeführt und Fehlmessungen vermieden werden.

### 7.3.2 Weiterentwicklung

Parallel zu den Anwendungen wurden in dieser Zeitspanne weiterführende Fragestellungen untersucht. So ging es in [29] um Synchronisation und Kommunikation in verteilten Automatisierungsprogrammen. Der *Guard* basierte Vorschlag wurde später in die Norm Mehrrechnerpearl (DIN 66253 Teil 3) übernommen.

Eine andere Fragestellung betrifft den Test von verteilten Programmen in der Prozeßautomatisierung [30]. Während bei zeitunkritischen Systemen der Test einfach durch Anhalten der CPU erfolgen kann, so ist dieser Ansatz offensichtlich bei nebenläufigen verteilten Programmen nicht möglich.

Zu Beginn der 1980-er Jahre wurde PEARL auf die aufkommenden Mikroprozessoren implementiert [11, 12, 13, 14, 16, 15, 18, 19, 20]. In den Universitäten erfolgte dies unter Förderung der DFG [39].

### 7.3.3 Standardisierung

1977 wurde die Sprachdefinition für *Full PEARL* fertig gestellt. Diese ersetzte die Sprachdefinition aus dem Jahr 1971 [5]. 1978 startet die Normung unter dem DIN-Arbeitskreis DIN/FNI AK 5.8. Die Sprachentwicklung wurde fortgeführt. Dies ist auch aus den entsprechenden Normen beim DIN nachzuvollziehen:

- DIN65252, Teil 1: Basic PEARL. Beuth Verlag, Berlin, 1979 und 1981
- DIN66253, Teil 2: Programmiersprache PEARL — Full PEARL. Beuth Verlag, Berlin, 1982.
- DIN 66253 Teil 3: Programmiersprache PEARL — Mehrrechner-PEARL. Beuth Verlag, Berlin, 1989.
- DIN 66253 Teil 2: Programmiersprache PEARL — PEARL90. Beuth Verlag, Berlin, 1998

## 7.4 Aktueller Stand

Ende der 1980-er Jahre begann ein langsamer Rückzug der beteiligten Firmen, wodurch sich die Weiterentwicklung auf die Hochschulen verlagerte. Die bedarfsorientierte Entwicklung von PEARL kam zum Erliegen.

Die Informatik hatte damals ihren Fokus auf Gebiete verlegt, die abseits der Echtzeitwelt waren. Die Zeit der globalen Vernetzung war angebrochen, die Industrie hungerte nach C-Programmierern — JAVA entstand. Die Weichen waren auf Objektorientierung und Vernetzung gestellt.

### 7.4.1 Wissenschaftliche Einbettung

Der PEARL-Verein wurde 1991 aufgelöst und die aktive Arbeitsgruppe in die Gesellschaft für Informatik in der Fachgruppe FG 4.4.2 Echtzeitprogrammierung eingebettet [45]. Mit der Zusammenlegung mit der Fachgruppe FG 4.4.1 Echtzeitsysteme erfolgte im November 2003 eine Bündelung der Echtzeitaktivitäten der beiden Fachgruppen unter dem Namen „Echtzeitsysteme und PEARL“ und unter der kürzeren Form „Real-Time“, um deren Zielsetzung als Kompetenzzentrum für Echtzeitfragen auszudrücken.

### 7.4.2 Embedded Praxis

An der Universität Hannover wurde im Fachbereich ein zweibeiniger Roboter mit PEARL RTOS-UH entwickelt [41]. Mehrere PowerPC-Systeme kommunizierten über mehrere CAN-Busse miteinander und führten eine Gleichgewichtsregelung und Bahnplanung aus.

Absolventen aus diesem Arbeitsbereich gründeten mittelständische Unternehmen, wie z. B. die Firma esd electronic und das Ingenieurbüro für Echtzeitprogrammierung (IEP). Die Weiterentwicklung und Wartung von RTOS-UH wird seit 2009 von der Firma IEP übernommen. Der Erfolg von PEARL läßt sich an den Zahl der RTOS-UH Systeme ablesen. So wurden im Jahr 2002 ca. 6.000 RTOS-UH Kopien ausgeliefert und insgesamt ca. 40.000 bis Juni 2003 [40].

Das Internet der Dinge (IoT) erfordert sichere, eingebettete Systeme mit der Möglichkeit der Vernetzung. In der Makerscene werden die inzwischen gut und günstig verfügbaren leistungsfähigen Einplatinensysteme wie den Raspberry Pi oder auch Arduino eingesetzt. Dort wird in den Referenzbeispielen viel mit der Skriptsprache Phyton gearbeitet — oder mit den nur eingeschränkt echtzeitfähigen Arduino Sketches. Diese Systeme mit ihrer Leistungsfähigkeit sowie der einfachen Möglichkeit, Sensoren und Aktoren anzusprechen, machen die Sprache PEARL wieder sehr interessant.

### 7.4.3 Open-Source

Seit 2012 wird in einem Arbeitskreis der GI-Fachgruppe Echtzeitsysteme unter dem Titel „Open-PEARL“ ein quelloffenes Übersetzungssystem für PEARL90 entwickelt<sup>1</sup>. Als primäre Zielsysteme werden hier Linuxsysteme und Microcontroller verfolgt. Die Übersetzung erfolgt unter einem Linuxsystem zunächst in C++ Quellcode, welcher dann von der gcc-Toolchain unter Nutzung

<sup>1</sup>Die Quellen sind verfügbar unter <https://sourceforge.net/projects/openpearl/>

der POSIX-Schnittstelle zu einer nativen Linux-Multithreadinganwendung wird. Bei der Variante mit dem Microcontroller wird der C++-Quellcode crosscompiliert. Für das Scheduling kommt dort FreeRTOS [34] zum Einsatz.

Dies führte zur Idee, ein PEARL-System auf die zwischenzeitlich sehr weit verbreitete Linux-Plattform zu portieren, um die Klarheit der Sprache für die Systemkonzeption und die Realisierung nutzen zu können.

Mit der Implementierung von OpenPEARL wird ein quelloffenes PEARL-System erstellt, welches einerseits auf einem normalen Linux aufsetzt und andererseits auch mit Hilfe des externen Schedulers FreeRTOS direkt auf verbreiteten Mikrocontrollern (z. B. ARM-Cortex-M3) betrieben werden kann.

Damit gibt es ein portables PEARL-System auf dem aktuellen Stand der Technik.

## 7.5 PEARL-Archiv

In der Entwicklungszeit von ca. 40 Jahren unter dem Dach des PEARL-Vereins und später der GI-Fachgruppe entstand eine umfangreiche Sammlung von Dokumenten. Soweit die Nutzungsrechte es zulassen, werden diese Dokumente derzeit in den Dokumentenbereich der GI-Fachgruppe als Archiv veröffentlicht.

## Literatur

- [1] P. Elzer: Anmerkungen zur (Früh-)Geschichte von PEARL. In: GI-Fachgruppe 'EP' (Hrsg.): PEARL-News 2/2005, S. 3–8, ISSN 1437-5966
- [2] J. Brandes, S. Eichentopf, P. Elzer, L. Frevert, V. Haase, H. Mittendorf, G. Müller, P. Rieder; PEARL — The Concept of a Process- and Experiment-oriented Programming Language; elektronische datenverarbeitung 10(1969) S. 429–442
- [3] B. Eichenauer, et al.; PEARL, eine prozeß- und experimentorientierte Programmiersprache, Angewandte Informatik, Jahrgang 15, Heft 9/1973, Vieweg Verlag, Braunschweig
- [4] K.H. Timmesfeld, B. Schürlein, P. Rieder, K. Pfeiffer, G. Müller, K. Kreuter, P. Holleczeck, V. Haase, L. Frevert, P. Elzer, S. Eichentopf, B. Eichenauer, J. Brandes; PEARL — Ein Vorschlag für eine Prozeß- und Experimentautomatisierungssprache; Gesellschaft für Kernforschung mbH, Karlsruhe, PDV-Bericht KfK-PDV1, April 1973
- [5] Projekt Prozeßlenkung mit DV-Anlagen; PEARL — Language Description; Forschungsbericht KfK-PDV 130; Karlsruhe (1977)
- [6] P. Elzer, P. Holleczeck; ASME-PEARL-Compiler wird der Öffentlichkeit vorgestellt; Regelungstechnik 12(1975), S. 433
- [7] Krupp Atlas erweitert ERP1000: PEARL und Pascal für Forschung und Lehre; In: Computerwoche vom 28.4.1978
- [8] H. Schoknecht; PEARL300: Kompiliersystem und Verwendung für Echtzeitaufgaben; In: U. Voges (Hrsg.); Tagungsbericht der 9. Jahrestagung des Siemens Prozeßrechner Anwenderkreis I; KfK-Bericht 2642 (1978) S. 161–175

- [9] Siemens System 300: Neue Produkte und Systeme für die Prozeßautomatisierung. In: Computerwoche 27.4.1979
- [10] T. Martin: Einführung; In: T. Martin (Hrsg.); Industrielle Erfahrungen mit der Programmiersprache PEARL; KfK-PDB 171 (1979)
- [11] J. Assenbauer: c't 68000-Software - Das Konzept. In: c't 2(1985), Heise Verlag (1985) S. 79
- [12] W. Gerth, R. Hausdörfer: c't 68000-Software - Multitasking und Echtzeit. In: c't 3(1985), Heise Verlag (1985) S. 36–39
- [13] W. Gerth: c't 68000-Software - PEARL-Compiler, Assembler, Editor. In: c't 4(1985), Heise Verlag (1985) S. 62–65
- [14] B. Reißerweber: c't 68000-Software - Programmieren mit PEARL. In: c't 5(1985), Heise Verlag (1985) S. 36–41
- [15] U. Bartels, J. Hertzner: c't 68000-Software - Prozessornah. In: c't 7(1985), Heise Verlag (1985) S. 124–128
- [16] K. Koerth: Echtzeit-Multitasking mit RTOS/PEARL - Mehrere Programme 'gleichzeitig' jetzt auch auf dem Atari ST. In: c't 6(1986), Heise Verlag (1986) S. 60–62
- [17] K. Koerth: Echtzeit-Multitasking mit RTOS/PEARL - Teil 2: Umgang mit dem Betriebssystem. In: c't 7(1986), Heise Verlag (1986) S. 62–67
- [18] C-M. Weitz: Echtzeit-Multitasking mit RTOS/PEARL - Teil 3: Programmieren in PEARL unter RTOS. In: c't 8(1986), Heise Verlag (1986) S. 94–97
- [19] C-M. Weitz: Echtzeit-Multitasking mit RTOS/PEARL - Teil 4: Programmieren in PEARL unter RTOS. In: c't 9(1986), Heise Verlag (1986) S. 53–58
- [20] H-J. Himmeröder, W. Mayer-Gürr, R.M. Toschke: c't-KAT-Ce: Ein 68000-Einplatinensystem, Teil 1: Die Platine. In: c't 11(1986), Heise Verlag (1986) S. 86–90
- [21] PEARL-Verein (Hrsg): PEARL-Tagung '83, PEARL Verein e.V., Stuttgart (1983)
- [22] PEARL-Verein (Hrsg): PEARL-Tagung '84, PEARL Verein e.V., Stuttgart (1984)
- [23] PEARL-Verein (Hrsg): PEARL-Tagung '85, PEARL Verein e.V., Stuttgart (1985)
- [24] PEARL-Verein (Hrsg): PEARL-Tagung '86, PEARL Verein e.V., Stuttgart (1986)
- [25] PEARL-Verein (Hrsg): PEARL-Tagung '87, PEARL Verein e.V., Stuttgart (1987)
- [26] PEARL-Verein (Hrsg): PEARL-Tagung '88, Springer Verlag, Berlin Heidelberg (1988)
- [27] W.A. Halang (Hrsg.): PEARL 91 - Workshop über Realzeitsysteme, PEARL Verein e.V., Stuttgart (1991)
- [28] H. Windauer: Aktuelle Implementierung des PEARL Systems von Werum. In: H.J. Schneider (Hrsg): PEARL-Tagung '83, PEARL Verein e.V., Stuttgart (1983) S. 93–96
- [29] A. Fleischmann, et al.: Synchronisation und Kommunikations verteilter Automatisierungsprogramme. In: Angewandte Informatik 2, Heft 7 (1983)
- [30] C. Andres, et al.: Testen von verteilten Programmen zur Prozeßautomatisierung. In: Angewandte Informatik, Heft 2 (1985)

- [31] P. Holleczeck, R. Müller; Ein PEARL-Programmiersystem für 68k-PCs; Personal-Realtime-Computing '86; Markt&Technik; 1986; ISBN 3-89090-408-4
- [32] F. Graf (1981) PEARL für Microcomputer. In: Baumann R. (eds) Fachtagung Prozessrechner 1981. Informatik-Fachbericht, vol. 39. Springer, Berlin, Heidelberg
- [33] P. Holleczeck, L. Frevert; Z80-PEARL für Verteilte Prozeßdatenverarbeitung; Regelungstechnische Praxis; 9(1983) S.375
- [34] R. Barry; FreeRTOS; [www.freertos.org](http://www.freertos.org) (31.3.2017)
- [35] P. Holleczeck; Erfahrung mit portabler Systemsoftware bei PEARL-Implementierungen für die Prozeßrechner SIEMENS 306 und 310; In: Portable Software / hrsg. von J. Schneider. Stuttgart, Teubner Verlag, 1980, (Berichte des German Chapter of the ACM; Bd 4) ISBN 3-51902423-3
- [36] RSX11M PEARL Users Guide, 1979, Digital Equipment Corporation, Maynard, Massachusetts
- [37] PEARL Manual, July 1979, Digital Equipment Corporation, Maynard, Massachusetts
- [38] C. Andres, et al.; Control and Accounting of user Access to a PDN from Private X.25 Data Networks; Proceedings of IEEE INFOCOM '88; New Orleans; IEEE; (1988)314-319
- [39] Gefördert im Rahmen eines DFG-Schwerpunktprogramms unter FKZ Ho 824-2
- [40] PEARL-User-Group AK1 GI-FG 4.4.2 — Treffen am 12.06.2003; In: B. Vogel-Heuser (Hrsg.), PEARL-News 1(2003)6-7
- [41] M. Seebode, W. Gerth: Echtzeitsystem für einen zweibeinigen Roboter mit adaptiver Bahnplanung; In: P. Holleczeck, B. Vogel-Heuser (Hrsg.): Mobilität und Echtzeit, PEARL 2007, S. 88–97. Springer, Berlin, 2007
- [42] E. Heilmair, P. Holleczeck, M. Trauter: Verteilte Programme zur ausfallsicheren Datenerfassung mit redundanten Mikrorechnern; Automatisierungstechnische Praxis atp, 28. Jahrgang, Heft 7/1986, S. 336–340
- [43] R. Besold, et al.: An Automatic Fault Monitoring System using a Microcomputer; Nucl. Instr. and Meth. A250(1986)534-538
- [44] R. Baran, et al.: An Intelligent Multichannel Analyser for Stability Supervision of Pulse Height Spectra; Nucl. Instr. and Meth: A258(1987)91-94
- [45] GI-Fachgruppe ist Ersatz für Pearl-Verein; Computerwoche; 27.12.1991