

Hans Dieter Hellige

Speicherprogrammierbare Steuerungen (SPS) Entwicklungsmuster im Technologielebenszyklus

Lehren aus der SPS-Technik

- Unerwartete Entwicklung einer Anwender-orientierten Steuerungstechnik: obwohl seit den 80er Jahren totgesagt, hat ist der SPS-Weltmarkt durch den Marktzutritt Osteuropas und Chinas noch immer im Steigen
- Relevanz der einfachen Gestaltung (Strong-and-Simple-Lösung) und der Nutzergruppenspezifischen Anwenderprogrammierung für die Lebensdauer der SPS-Technik
- Wechsel der Markt- und Technologieführerschaft (USA, BRD/Westeuropa, Japan)
- Beispiel für regionale Technik- und Programmierstile
- Beispiel für eine Reihe dynamischer Entwicklungsmuster:

Dynamische Entwicklungsmuster bei den SPS

- Ablösungsprozesse: Brüche, Übergänge
- Systemkonkurrenz, Systemstreite
- Skalierung, Größensteigerung bzw. -verkleinerung
- Up-/Downgrading der Funktionalität
- Modularisierung der Bauformen
- Dezentralisierung, Verteilung, Vernetzung
- Normung

SPS im Rahmen anderer Steuerungstechniken

Abgrenzung von Steuerungs- und Regelungstechnik

Steuerung: offener Wirkungsablauf oder geschlossener Wirkungsweg, bei dem keine Rückkopplung über dieselben Eingangsgrößen erfolgt

Regelung: geschlossener Wirkungsablauf, bei dem die Regelgröße im Wirkungswege des Regelkreises sich fortlaufend selber beeinflusst

SPS ist eine Steuerungstechnik, die aber immer mehr Regelungsaufgaben übernommen hat.

Steuerungsarten

- *Unterscheidung nach Art des logischen oder zeitlichen Ablaufes:*

Verknüpfungs-Steuerung: Ableitung von Steuerungsgrößen (Ausgänge) aus Führungsgrößen des Prozesses (Eingänge) durch logische Verknüpfungen

Ablauf-Steuerung: Sequentielle Auslösung vorgegebener Befehlsfolgen gemäß Zeitbedingungen oder Prozeßzuständen

Zeitplan-Steuerung: Rein zeitabhängige Weiterschaltungsbedingungen

Wegplan-Steuerung, Halteglied-Steuerung

- *Unterscheidung nach Art der Informationsdarstellung der Signalverarbeitung:*

synchron, asynchron, analog, digital, binär

- *Unterscheidung nach der zugrunde liegenden Technologie:*

Mechanische Steuerungen: Zeitschaltuhren

Elektromechanische Steuerungen: Relais, Schaltschützen (seit ca. 1900)

Elektronische Steuerungen: Röhrensteuerungen, Lichtschranken/Senzellen (seit 1930er Jahren), Prozeßrechner (seit Ende 1950er Jahre), Speicherprogrammierbare Steuerungen (seit Ende 1960er Jahre), IPC (seit Mitte der 80er Jahre)

- *Unterscheidung nach Art der technischen Realisierung:*

Relais, Schütz, Zeitschaltuhr/Timer

Verbindungsprogrammierte Steuerung (VPS): *Feste Verdrahtung* (Löten, Klemmen) von Verknüpfungen bzw. Verriegelungen oder *variable Verdrahtung* über Stecktafeln, Matrix-Steuerungen und programmierbare Netzwerke mit Kreuzschienenverteilern

Prozeßrechner (PR): Echtzeitrechner für die ereignisgesteuerte Prozeßführung auf der Basis mathematischer Modelle und arithmetischer Verknüpfungen

Speicherprogrammierte Steuerung (SPS): Austausch-programmierbare Speichersteuerungen mit unverändertem oder veränderbarem Speicher (ROM, PROM, EPROM) oder freiprogrammierbare Speichersteuerungen (RAM, EEPROM, CMOS-RAM) zum Verknüpfen logischer Zustände im Steuerungsablauf

Numerische Steuerung (NC, CNC), Positionier- und Bahnensteuerungen, Steuerungsfunktionen als Nebenaufgabe

Robotersteuerung (RC): Positionier- und Bahnensteuerungen für Montage und Handhabungstechnik

Industrie-PC (IPC): Robuster PC mit Software-Realisierung der Steuerungsfunktionen

Entstehungshintergrund und -anlass der SPS: Lücke zwischen konventionellen Steuerungen und PR

Auslöser neuer Steuerungstechniken: Die Automatisierungswelle um den 2. Weltkrieg zeigt Grenzen der bisherigen Relais- und Schützsteuerungen bei Transferstraßen und verfahrenstechnischen Großanlagen. Die Vermehrung der Produktvarianten und die Verkürzung der Produktlebenszyklen durch schnellere Typenwechsel ist mit herkömmlichen Steuerungstechniken nicht mehr zu bewältigen.

Alternative Lösungsstrategien: *Entwicklungssprung* (Bruch zwischen alt und neu), *Entwicklungskontinuität* (inkrementelle Verbesserungen ohne Technologiewechsel), *Entwicklungsübergang* (Technologiewechsel mit Übergängen zwischen alt und neu)

Lösungsweg 1 Entwicklungssprung in USA:

Innovationssprung vom Relais- /Schütz zum Prozeßrechner Ende der 50er Jahre aufgrund des Entstehens des Realtime-Computers, der sich im Prinzip für die Prozeßführung in der Verfahrenstechnik und Fertigungstechnik eignet. Der Versuch einer sofortigen PR-Einführung in der Ölindustrie und Kraftwerkstechnik zu komplexen Steuerungszwecken scheitert. Die Probleme zentralistischer PR-Architekturen bei der Bewältigung der Prozeßkomplexität bewirken ein Festhalten an konventionellen Relais- und Schützsteuerungen. Dies führt zu einer Automatisierungslücke, die Steuerungstechnik wird zum Engpaß

Lösungsweg 2 Entwicklungskontinuität in Deutschland, Westeuropa: Vermeidung der Automatisierungs-Lücke durch Entwicklungskontinuität. Wegen des Rückstandes bei der PR-Entwicklung Zuflucht zu *Verbesserungsinnovationen* bei konventionellen Steuerungen, Entwicklung der VPS (Verbindungsprogrammierbaren Steuerungen):

- Einsatz automatischer Verdrahtungsverfahren
- Einsatz gedruckter Leiterplatten
- Physikalische Trennung von Programm und Verdrahtung = matrix-programmierte Steuereinheiten, programmierbare Netzwerke und Kreuzschienenverteiler

Lösungsweg 3 Entwicklungsübergang in USA: Invention und Innovation der SPS 1968/69 zur Schließung der Lücke zwischen traditionellen Steuertechniken und Prozessrechnersteuerung

Erfindung: Richard E. Morley von der Zulieferfirma für General Motors "Bedford Associates" entwickelt den *Programmable Controller (PC)* bzw. *Programmable Logic Controller (PLC)*

Erfinderstory: Frust über ständig wiederkehrende ähnliche Steuerungsaufgaben führt Sylvester 1968 zur Übernahme des Konzeptes der Speicherprogrammierung in die Automatisierungstechnik. Markteinführung der ersten SPS „Modicon 084“ im Jahr 1969, Umgründung der Firma in „Modicon“ (Modular Industrial Control)

Erfindungskonzept

Ein spezialisiertes Mikrocomputersystem zur Steuerung von Prozessen, das die herkömmlichen festverdrahteten oder verbindungsprogrammierten Schaltschränke ablöst und bei dem die steuernden Funktionen in einem Speicher abgelegt werden. Die Hardware ist dadurch problemunabhängig, die speziellen Anwendungsfunktionen werden über Software realisiert. Die Steuerungsaufgabe wird durch eine Folge von Anweisungen auf der Basis Boolescher Verknüpfungen programmiert, die von der SPS in Zyklen sequentiell abgefragt und abgearbeitet werden. Die SPS arbeitet taktgesteuert in serieller Arbeitsweise im Unterschied zur parallelen Arbeitsweise von VPS-Systemen.

Leitbild für SPS-Steuerungstechnik

Robuste prozeßnahe Steuerungstechnik mit Programmierung durch die prozeßerfahrenen Anwender und nicht durch prozeßunkundige DV-Spezialisten.

Erfolgsfaktoren der SPS: komparative Nutzungsvorteile gegenüber VPS und PR

- Vereinfachte Umprogrammierung, Parametrisierung und Einstellung von Maschinen
- Vereinfachung der Projektierung, Realisierung, der Tests und Fehlersuche
- Platzersparnis durch kompaktere Bauweise
- Kopierbarkeit und Wiederverwendbarkeit der Programme
- Erweiterbarkeit der Steuergeräte und -programme
- Übergangsgerechte Lösung, Anschließen an vorhandenes Prozeßwissen

Die SPS haben die erste Dezentralisierungswelle der Automatisierungstechnik ermöglicht. Sie wurden die tragende Säule der Automatisierung im Automobilbau, Maschinenbau und in der Chemischen Verfahrenstechnik. Sie haben die Relaissteuerungen und VPS bis auf eine kleine stabile Nische extrem schneller Anwendungen nahezu völlig verdrängt. Durch die folgende Größensteigerung und Miniaturisierung sind SPS einerseits immer mehr in Prozeßrechner-Aufgaben und andererseits in den Bereich der Klein- und Kleinststeuerungen vorgezogen. Es entwickelte sich nach und nach eine Arbeitsteilung in der Automatisierungshierarchie, in der die SPS aufgrund ihrer Typenvielfalt oft mehrere Ebenen abdeckten. Ihr weiterer Aufstieg wurde seit den 90er Jahren nach oben durch den Industrie-PC und nach unten durch die prozessnahen Feldgeräte und Feldbussysteme begrenzt, trotzdem kam es bis heute nicht zu der seit langem prognostizierten Verdrängung der SPS-Technik.

Ausdifferenzierung der SPS-Programmierkonzepte

US-Ursprungskonzept: Prinzip der Anwenderprogrammierung, sprungvermeidendes Programmierkonzept führt in USA zur Entwicklung einer regelbasierten Sprache, die an die bereits 1937 in Deutschland entwickelten Relaisymbolik anknüpfte und durch ihre Nähe zum traditionellen Stromlaufplan gut vom Anlagen- und Wartpersonal akzeptiert wurde: *Kontaktplan (KOP)*, *Ladder Diagram*

Technologietransfer nach Deutschland / Westeuropa und Anpassung an VPS-Kultur

Import des KOP und Neuentwicklung eines auf VPS aufsetzenden funktionsorientierten Programmierkonzeptes, das mehr auf Elektroniker zugeschnitten war:

Funktionsplan (FUP), *Function Block Diagram*

Die größere DV-Orientierung der europäischen Steuerungstechnik führt zur Erweiterung der Programmiermöglichkeiten durch eine Assembler-artige SPS-Programmiersprache:

Anweisungsliste (AWL), *Instruction List*

Technologietransfer nach den USA

Zusätzlich zum vorherrschenden KOP Aufnahme der FUP- und AWL-Programmierung durch US-Hersteller

Resultat der frühen Programmiersprachen-Entwicklung

SPS wird die Protesttechnik gegen PR-Welt und die Dominanz der und Hochsprachen-Programmierung. Es bilden sich unterschiedliche nationale und professionelle Programmierstile heraus, von denen keiner die Oberhand gewinnt. Der Pluralismus der Programmierkonzepte führt zum unerwarteten Aufstieg der SPS-Technik.

Richtungsstreit über die Entwicklung der Programmierkonzepte

- Annahme eines schnellen Wechsels zur Hochsprache (C, Pascal, Ada) aufgrund der größeren Sprachmächtigkeit und der DV-Kompatibilität
- Bewahrung anwenderspezifischer Programmierstile als Übergangs- oder Dauerlösung
- Aufkommen neuer textueller und grafischer Programmieransätze für unterschiedliche Anwendungszwecke
- Die Vorteil-/Nachteilbilanz ergibt keinen Vorrang einer Programmierertechnik
- Die Folge ist ein Turmbau-zu-Babel-Syndrom: unüberschaubare Vielfalt der SPS-Programmiersprachen. Die proprietäre Schließung von Geräte- und Programmierkonzept führt zu Herstellerabhängigkeit der Anwender, kleinen Losgrößen und relativ hohen Preisen.
- Kompromißlösung der Normung eines Standardsatzes von Programmierstilen, die jeweils über Postprozessoren in einander überführbar sind.

Normungsprozeß bei Programmiersprachen:

Vornorm IEC 65 a, Norm IEC 1131-3, DIN EN 61131-3)

Fünf verschiedene mit einander kompatible Programmierkonzepte:

Zwei textuelle, mnemotechnische Sprachen:

Anweisungsliste AWL (Instruction List): assemblerartig, maschinennah, universell einsetzbar, Hersteller-unabhängig, zeilenorientiert; aufgrund der prozessornahen Befehle sehr schnelle Abarbeitung der Anweisungen, AWL fungiert daher auch als Zwischensprache bzw. allgemeine Sprache für die anderen textuellen und grafischen SPS-Programmiersprachen.

Strukturierter Text ST (Structured Text): hochsprachenartig, abstraktere Befehle für mächtigere Sprachkonstrukte

Drei grafische Sprachen:

Kontaktplan KOP (Ladder Diagram): grafische Verknüpfungssprache aus dem Bereich elektromechanischer Relaissysteme abgeleitet, nahe am Stromlaufplan, daher von Elektrikern und Anlagenfahrern bevorzugt

Funktionsbausteinsprache FBS (Function Block Diagram): aus dem Bereich der Signalverarbeitung stammende, funktionsorientierte Sprache für logische Ablaufketten unabhängig von der steuerungstechnischen Lösung, erleichtert die Verständigung zwischen Fachleuten verschiedener Disziplinen und zwischen Hersteller, Auftraggeber und Anwender.

Ablaufsprache AS (Sequential Flow Chart): graphisch/textuell; Petrinetz-artige Netzmodellierung für komplexere Kontrollflüsse, Weiterentwicklung der Sprache Grafcet

Diverse Hersteller-spezifische inkompatible Programmierkonzepte:

Siemens STEP 1-7 ist aufgrund der langjährigen Weltmarktführung von Siemens-SPS noch immer stark verbreitet. Zur Erhaltung der Marktstellung ständige Weiterentwicklung und dabei zunehmende Anlehnung an die genormten Sprachen.

Entwicklungsdynamik und Entwicklungsrichtungen der SPS:

Substitutionskonzept der frühen 70er Jahre: Bloßer Ersatz der Schütz- und Relaissteuerungen, traditionale Ausrichtung der amerikanischen SPS Technik

Innovationskonzept der späten 70er und frühen 80er Jahre
Funktionsausweitung und systemische Ausrichtung der europäischen SPS Technik

Größensteigerung der SPS, Mimikry in Richtung Prozeßrechner (Upscaling, Upgrading)

Größenverkleinerung, Einbruch in den Schützen- und Kleinststeuerungsmarkt (Downscaling, Downgrading), Erschließen neuer Anwendungsfelder durch miniaturisierte Bauweisen und Niedrigpreis-Geräte speziell der japanischen SPS-Technik

Koevolution von SPS und Mikroelektronik , Miniaturisierung der Baugrößen,
Funktionsausweitung infolge verbesserter Prozessoren und Speicher
Das Moore'sche Gesetz wirkt auch hier als Technologietreiber.

Funktionsausweitung der klassischen SPS:

Protokollierung von Betriebszuständen

Signalisierung von Alarmzuständen

Berechnen von Betriebszuständen

Regelungsfunktionen

Übernahme von NC/CNC-Funktionen

Bearbeitung ereignisgesteuerter Reaktionen

Prozeßvisualisierung

Prozeßkommunikation

Vernetzung

Muster-/Objekterkennung

Datenverwaltung

Fehlerdiagnose

Simulation

Fuzzy-Control

SPS-Produktpolitik und Marktstrategiekonzepte in den 80er/90er Jahren

Japanische Downgrading und Mengenskalierung-Strategie

Klein- und Kleinststeuerungen für den 'Massenmarkt'

Europäische und amerikanische Gegenoffensive

Europäisches SPS Familienkonzept: von kleinsten Kompaktgeräten bis zu Prozeßrechner-äquivalenten Höchstleistungs-SPS-Systemen, proprietäre komplette SPS-Familien, branchenspezifische Gesamtlösungen

Amerikanische Gegenstrategie mit PC-orientierten offenen Steuerungen

Next Generation Controller 1990: Integration von SPS, NC, CNC, RC-Steuerungen

Nachziehen der europäischen Hersteller: OSACA-Projekt ab 1991/92

Konzept des Universalsystems für alle Steuerungen auf PC-, Windows-Basis

Problem der großen Komplexität von Universalsteuerungen

Modularisierung des Geräte- und Programmkonzeptes der SPS

Zielkonflikt Kompakt-SPS und Modular-SPS,

Aufteilung der Funktionalität in Grundfunktionen (logische Verknüpfungen, Speicher-, Zähl-, Zeit-, Vergleichs- und Schiebe-Funktionen, Arithmetik,

Zusammenfassung von häufig benutzten Folgen von Funktionen zu Funktionsbausteinen,

Aufgaben-orientierte Strukturierung der Programme nach dem Vorbild der strukturierten

Programmierung (Task als virtuelle Maschine)

Angebot spezifischer Funktionsbaustein-Bibliotheken durch Hersteller (proprietär!)

Dezentralisierung: verteilte SPS, Einbeziehung in Prozeßleitsysteme

Die SPS ermöglichte gegenüber zentralen Prozeßrechnern eine Dezentralisierung von Automatisierungssystemen, sie war aber zunächst selber ein zentralistisches System mit einer großen Zahl von Ein- und Ausgängen. Die Konkurrenz mit Prozeßrechnern führte zur Größensteigerung und Funktionsausweitung (zentrale Groß-SPS). Die Probleme von Zentralsystemen und die Entwicklung der Mikroelektronik führten auch hier zu stärker dezentralen Architekturen zur Entlastung der Hauptsteuerung.

- Aufteilung einer Groß-SPS in verkettete Einzelsteuerungen (zentrale SPS mit dezentralen E/A, zentrale SPS mit Netz intelligenter dezentraler SPS)
- Integration der SPS in das hierarchische Automatisierungskonzept (Fabrikpyramide, Steuerungspyramide) über eine Hierarchie von Bussystemen
- Auslagerung von Vorverarbeitungsfunktionen in die Feldebene durch Intelligente Sensoren und Aktoren, die über einen Sensor-Aktorbus bzw. Feldbus vernetzt sind

SPS in Systemkonkurrenz mit IPC und Feldbus

Ziele und Vorteile des IPC:

- Überbrückung der unterschiedlichen Steuerungsarten und proprietären Inselwelten
- Verbesserte Netzwerkfähigkeit, Kommunikationsmöglichkeiten zwischen Prozessen
- Deutliche Funktionsausweitung: Visualisierung, Graphische Benutzer-Interfaces, Datenbankfunktionen, komplexere Prozeßdarstellung, Überwindung der Mängel der herstellerspezifischen SPS-Programmiergeräte
- Größere Flexibilität und Erweiterbarkeit

Nachteile des IPC:

- Erhöhte Hardware-Empfindlichkeit (Kälte, Hitze, EMV)
- Größere Software-Komplexität und dadurch Absturzgefährdung (Windows-NT, Windows-2000, Windows-XP)
- Insgesamt geringere Verfügbarkeit
- Unzureichende Echtzeitfähigkeit

Vorteile der SPS:

- Höhere Verfügbarkeit und Unempfindlichkeit im Industrieinsatz
- Größere Gerätevielfalt und Konfigurationsmöglichkeiten in der Hardware (von kleinster Mikro- / Nano-Kompakt-SPS bis zur modularen Groß-SPS)
- Längere Produktlaufzeiten, Unabhängigkeit von Software-Spirale, Aufwärts- und Abwärts-Kompatibilität

Weltmarkt-Zahlen:

1980	SPS = 2,5 Mrd DM	
1988	SPS = 3 Mrd DM	
1990	SPS = 7 Mrd DM	
1991	SPS = 7,6 Mrd DM	
1992	SPS = 7,2 Mrd DM	
1996/97	SPS = 8 Mrd DM	Industrie-PC= 2 Mrd. DM
2003	SPS = 6 Mrd Euro	
2004	SPS = 6,2 Mrd Euro	
2005		Industrie-PC= 8 Mrd. Euro
2007	SPS = 7 Mrd Euro	Industrie-PC= 8,8 Mrd. Euro

Auswirkungen des IPC auf SPS

Scheitern des erhofften Blitzstartes und der schnellen Ablösung der SPS durch den IPC in der 2. Hälfte der 80er Jahre, erneuter Anlauf Mitte der 90er mit Windows-NT (1994); heiße Phase der Systemkonkurrenz ab 1995/96; Zangenangriff durch Windows-NT/CE und die Fieldbus-Foundation, aber nach wie vor Probleme mit der Sicherheit und Echtzeitfähigkeit.

- IPC ist noch oft Ergänzung zur SPS für Programmierung und Visualisierung
- Keine Bedrohung der SPS in Anwendungen mit Sicherheitsanforderungen
- Barriere für weiteres Upgrading der SPS durch IPC
- Barriere für weiteres Downgrading der SPS durch Sensor-Aktorbus- und Feldbus-Systeme
- Konkurrenz durch PC-basierte Soft-SPS in unkritischen Einsatzfeldern
- Konkurrenz neuer Totalsystemlösungen: Open Control-Konzept als SPS-Ablösung, Totally Integrated Automation-Konzept als dauerhafte Hybridlösung
- Länger andauernde Koexistenz mit kleiner werdendem Marktanteil für die SPS; aber keine Totalverdrängung, neue Expansion durch Ausweitung der Automatisierung in Schwellenländern, auf Dauer wohl Überleben der SPS in stabiler Nischenposition.

Zusammenfassung: Ablaufmuster von Systemstreiten

- 1 **Ausscheidungskampf:** Sieg einer Richtung über die andere(n)
 - 1a schnelle Verdrängung / Substitution
 - 1b langsamer Ablösungsprozeß
- 2 **Dominanz** einer Richtung, **Nischenposition** der unterlegenen
- 3 **Dauerkonkurrenz** (ähnliche Nutzungspotentiale, wechselseitige Imitation von Funktionalitäten)
- 4 **Koexistenz** (abweichende Nutzungspotentiale)
- 5 **Hybridlösung** bzw. **Kompromiß**
- 6 **Integration** in einem neuen Lösungsansatz
- 7 **Wiederbelebung** einer unterlegenen Richtung infolge technischer Neuerungen

Die auch bei dem SPS-IPC-Systemstreit erkennbare Vielfalt von Typen und Ablaufmustern von Technikkontroversen zeigt, daß der Alleinvertretungsanspruch eines Technologiepfades in den wenigsten Fällen gerechtfertigt ist. Denn der schnelle und vollständige Sieg einer Entwicklungsrichtung über konkurrierende Lösungswege bildet eher die Ausnahme. Da die komparativen Nutzungsvorteile bzw. -nachteile selten klar und eindeutig auf die streitenden Entwicklungsrichtungen verteilt sind, sind Teilsiege, Mischlösungen oder andere Formen des Nebeneinanders sowie langwierige Ablösungsprozesse weitaus häufiger vertreten.

Literatur

- Andratscke, Wolfgang, Der sichere Einstieg in speicherprogrammierbare Steuerungen. Von der festverdrahteten zur flexiblen Lösung, 1986
- Auer, Adolf, Steuerungstechnik und Synthese von SPS-Programmen, Heidelberg 1994
- Auer, Adolf, SPS - Aufbau und Programmierung, 4. Aufl Heidelberg 1991
- Auer, Adolf, SPS-Programmierung: Beispiele und Aufgaben, 4. Aufl Heidelberg 1994
- Behrendt; Borelbach, K. H., Steuerungstechnik mit speicherprogrammierbaren Steuerungen SPS, 4. Aufl. Europa-Lehrmittel Verlag 1992
- Benda, Dietmar, Speicherprogrammierbare Steuerungen für Praktiker, Berlin, Offenbach 1991
- Bewersdorff, Hans Jürgen, Ein universelles Programmiersystem für speicherprogrammierbare Steuerungen, n: Speicherprogrammierbare Steuerungsgeräte, Tagung Stuttgart-Fellbach 1978, VDI-Berichte 327, Düsseldorf 1978, S. 69-71
- Borelbach, K. H. u.a. (Hrsg.), Steuerungstechnik mit speicherprogrammierbaren Steuerungen SPS, 4. Aufl. Europa Lehrmittel Verlag 1992
- Dirnfelder, Martin, SPS - Programmieren statt verdrahten. Ein Leitfadens für Einsteiger, München 1991
- Drigalsky, Ingo, Praxisbuch SPS. Grundlagen, Anwendungen, Simulation, Bonn, Albany u. a. 1995
- Friedrich, Alfred, Mit SPS erfolgreich automatisieren, München 1994
- Grötsch, Eberhard E., SPS. Speicherprogrammierbare Steuerungen vom Relaisersatz bis zum CIM-Verbund. Einführung und Übersicht, 3. Aufl. München, Wien 1995
- Grötsch, Eberhard E., SPS : Speicherprogrammierbare Steuerungen als Bausteine verteilter Automatisierung, 5. Aufl. München, Wien 2004
- Guggisberg, Simon, Tendenzen bei SPS, in : etz 107 (1986) 9,S.380-385
- Hengel, K., Softwareentwicklungen für speicherprogrammierbare Steuerung im integrierten rechnergestützten Konstruktionsprozeß, Berlin, Heidelberg, New York 1994
- Högener, J., Erfolgreich automatisieren mit Fuzzy-SPS, München 1994
- Holtkamp, Edwin, Das zweite Leben der SPS. Löst die Software-SPS die klassische SPS ab?, in: Elektronik 8/1998, S. 50-54
- John, Karl-Heinz; Tiegelkamp, Michael, SPS - Programmierung mit IEC 1131-3. Konzepte und Programmiersprachen, Anforderungen an Programmiersysteme, Entscheidungshilfen, Berlin, Heidelberg, New York 1995, 3. Aufl. 2000
- John, Karl-Heinz; Tiegelkamm, Michael, SPS-Programmierung mit IEC 1131-3, Konzepte, Anforderungen an Programmiersysteme, Entscheidungshilfen, 2. Auflage, Berlin, Heidelberg, New York 1997
- Kaftan, Jürgen, Analogverarbeitung mit speicherprogrammierbaren Steuerungen, Würzburg 1991
- Kaftan, Jürgen, SPS-Grundkurs. Aufbau und Funktion Speicherprogrammierbarer Steuerungen, Programmieren mit STEP 5, Anleitungen, Übungen, Lösungen, Würzburg 1989
- Kielhorn, Otto, Speicherprogrammierbare Steuerungen. Bisherige Erfahrungen, Ausblick in die Zukunft, in: etz 102 (1981) 18, S. 952-957
- Kleim, Dieter; Wolfgarten, Walter, Programmierbare Steuerungen, Düsseldorf 1975
- Klingenberg, Georg; Verhaag, Elmar, Der Einsatz programmierbarer Steuerungen aus der Sicht des Anwenders, in: VDI-Berichte 263, Düsseldorf 1976, S. 21-27
- Klinker, W., atp-Marktanalyse:SPS, in : automatisierungstechnische praxis 32 (1990) 4,S.153ff.
- Kopp, H, Kriterien bei der Projektierung und Anwendung von SPS, in: Speicherprogrammierbare Steuerungsgeräte, Tagung Frankfurt 1981, VDI-Berichte 396, Düsseldorf 1981, S. 21-26
- Krätzig, Joachim, Speicherprogrammierbare Steuerungen verstehen und anwenden. Grundlagen und Programmierung mit zahlreichen praxisbezogenen Aufgaben und Lösungen, München, Wien 1992
- Kriesel, Werner; Rohr, Hans; Koch, Andreas, Geschichte und Zukunft der Meß- und Automatisierungstechnik, Technikgeschichte in Einzeldarstellungen, Düsseldorf 1995
- Kröck, Alwin, SPS,der Schlüssel zur Automa-tisierung, in : Technische Rundschau, 47/1991,S.68-73.
- Leinemann, ; Wiederhold, SPS. Grundlagen der Technik und Programmierung, 3. Aufl. Handwerk und Technik Verlag 1993
- Litz, Lothar; Frey, Gat 4/99, S. 145-156 eorg, Methoden und Werkzeuge zum industriellen Steuerungsentwurf – Historie, Stand, Ausblick, in:
- Merz, Reiner, Der Weg zur SPS-Fachkraft, Teil 1: Einführung in das Programmieren einer SPS, München 1993
- Mollath, G., Speicherprogrammierbare Steuerungsgeräte. Stand der Technik, in: Speicherprogrammierbare Steuerungsgeräte, Tagung Frankfurt 1981, VDI-Berichte 396, Düsseldorf 1981, S.
- Neumann, Peter; Grötsch, Eberhard E.; Lubkoll, Christoph; Simon, René, SPS-Standard IEC-1131. Programmierung in verteilten Automatisierungssystemen, München, Wien 1995

- Petry, Jochen, Speicherprogrammierbare Steuerungen. Projektierung und Programmierung, Heidelberg 1989
- Plagemann, Bernhard, Methoden der Programmierung von SPS. Grundlagen und Beispiele mit firmenspezifischen Lösungen, Würzburg 1990
- Rolle, Ingo (Hrsg.), IEC 61131 - wozu? : SPS: Programmiersprachen, Kommunikation, Fuzzy-Control, Sicherheit und EMV nach VDE 0411 Teil 500, Berlin, Offenbach 1998
- Scharf, Achim, Speicherprogrammierbare Steuerungen : Mehr Leistung und Komfort, in: Hard and Soft 6 (1989) 7/8, S. 8-15
- Seitz, Matthias, Speicherprogrammierbare Steuerungen : von den Grundlagen der Prozessautomatisierung bis zur vertikalen Integration München, Leipzig 2003
- SPS-Programmierung: Beispiele und Aufgaben, 4. Aufl Heidelberg 1994
- Staniczek, Dieter, Auswirkungen der technologischen Fortentwicklung auf speicherprogrammierbare Steuerungssysteme, in: Speicherprogrammierbare Steuerungsgeräte, Tagung Stuttgart-Fellbach 1978, VDI-Berichte 327, Düsseldorf 1978, S. 9-13
- Thiele, G.; Renner, L.; Neimeier, R.: Programmable Logic Controllers, in: H. Unbehauen (Ed.): Control Systems, Robotics and Automation, in Encyclopedia of Life Support Systems (EOLSS), 2004
- Weck, Manfred; Schäfer, Klaus, Stand und Technik programmierbarer Steuerungen, in: Speicherprogrammierbare Steuerungsgeräte, Tagung Stuttgart-Fellbach 1978, VDI-Berichte 327, Düsseldorf 1978, S.1-6
- Weiss, W., Entwicklungen von Steuerungen mit Hilfe programmierbarer Systeme, in: wt-z, ind. fertigung, (1972), S. 539-543
- Wellenreuther, ; Zastrow, Steuerungstechnik mit SPS. Bitverarbeitung und Wortverarbeitung, Wiesbaden 1993
- Wölfel, Helmut, Die Entwicklung der digitalen Prozeßleittechnik – Ein Rückblick, in atp 40 (1998) 4, S. S1-28
- Wolfram, Fritz, Regelungstechnik mit Speicherprogrammierbaren Steuerungen, Würzburg 1991
- Wratil, Peter, Speicherprogrammierbare Steuerungen in der Automatisierungstechnik, Würzburg 1989
- Wratil, Peter, Moderne Programmierertechnik für Automatisierungssysteme, Würzburg 1996 (SPS)