

Beschreibung

Automatisierungsgerät

- 5 Die Erfindung betrifft ein Automatisierungsgerät gemäß dem Oberbegriff des Anspruchs 1. Ferner betrifft die Erfindung ein universelles, verteiltes und durchgängiges Automatisierungs- und Management-Engineering- und Informationssystem.
- 10 Ein Automatisierungsgerät mit den Merkmalen des Oberbegriffs des Anspruchs 1 ist aus dem Siemens-Katalog ST 70, Ausgabe 1995, bekannt. Ein Anwender erstellt mit einem Programmiergerät ein Programm zur Steuerung eines technischen Prozesses, das Software-Funktionsbausteine, z. B. in Form von Organisa-
- 15 tionsbausteinen, Programmbausteinen und Instanzdatenbausteinen, umfaßt. Das Automatisierungsgerät ist mit dem Programmiergerät über ein Bussystem verbunden, über welches das Programmiergerät das Steuerprogramm in das Automatisierungsgerät überträgt. Ferner ist ein Bedien- und Beobachtungsgerät
- 20 an das Bussystem anschließbar, das zur Prozeßführung vorgesehen ist und welches die Darstellung von mehrere Bildobjekte umfassenden Prozeßbildern ermöglicht.

- Häufig ist es erforderlich, daß Software-Funktionsbausteine
- 25 eines Steuerprogramms von einem Automatisierungsgerät eines Fertigungsstandortes oder aus einem Software-Pool dieses Fertigungsstandortes in ein Automatisierungsgerät eines anderen Fertigungsstandortes zu übertragen sind. Insbesondere wenn die Fertigungsstandorte sehr weit voneinander entfernt
- 30 sind, z. B. wegen einer Globalisierung von Fertigungsaktivi-

täten, werden diese Software-Funktionsbausteine über das globale Netzwerk „INTERNET“ übertragen. Dazu sind Server mit geeigneten Kommunikationsschnittstellen notwendig, die einerseits das INTERNET-Kommunikationsprotokoll und andererseits das Kommunikationsprotokoll der Automatisierungsgeräte ermöglichen. Aufgrund dieser unterschiedlichen Protokolle und der Architektur der Automatisierungsgeräte ist eine Einbindung der Software-Funktionsbausteine zur Laufzeit des Steuerprogramms nicht möglich, insbesondere dann nicht, wenn Automatisierungsgeräte unterschiedlicher Hersteller mit diesen Software-Funktionsbausteinen zu versorgen sind.

Der vorliegenden Erfindung liegt die Aufgabe zugrunde, ein Automatisierungsgerät der eingangs genannten Art zu schaffen, welches für einen Einsatz in einem global verteilten Automatisierungsverbund geeignet ist.

Darüber hinaus ist ein universelles Management-Engineering-System in Form eines Programmiergeräts sowie eines Bedien- und Beobachtungsgeräts für einen global verteilten Automatisierungsverbund und ferner ein Management-Engineering- und Informationssystem in Form von Workstations und Datenbank-Servern anzugeben.

Diese Aufgabe wird im Hinblick auf das Automatisierungsgerät durch ein Automatisierungsgerät der eingangs genannten Art mit den im kennzeichnenden Teil des Anspruchs 1 angegebenen, im Hinblick auf ein universelles Management-Engineering-System durch die in den Merkmalen des Anspruchs 6 sowie des Anspruchs 10 angegebenen Maßnahmen gelöst.

Vorteilhafte Ausgestaltungen der Erfindung ergeben sich aus den Unteransprüchen.

Anhand der Zeichnung, in der ein Ausführungsbeispiel der Erfindung dargestellt ist, werden nachfolgend die Erfindung, deren Ausgestaltungen und Vorteile näher erläutert.

Es zeigen

- Figur 1 ein universelles, verteiltes Automatisierungs- und Management-Engineering- und Informationssystem in schematischer Darstellung,
- Figur 2 ein Programmübersetzungs-Strukturbild und
- Figuren 3 und 4 Software-Funktionsbaustein-Ablaufsysteme (PLC-Object-Engine-Systeme).

Zwei Fertigungsstandorte 1 und 2 eines global verteilten Automatisierungsverbundes sind über ein an sich bekanntes globales Netzwerk „INTERNET“ 3 miteinander verbunden, wobei geeignete Einrichtungen 4, 5 vorgesehen sind, die verhindern, daß Unberechtigte Daten in die datenverarbeitenden Komponenten der Fertigungsstandorte 1, 2 übertragen. Die Fertigungsstandorte 1, 2 umfassen mehrere jeweils mit einer INTERNET-Kommunikationsschnittstelle versehene Geräte in Form von Automatisierungsgeräten 6, Programmiergeräten 7, Bedien- und Beobachtungsgeräten 8 und Workstations 9. Diese INTERNET-Kommunikationsschnittstellen ermöglichen eine TCP/IP-Protokoll-Kommunikation der Geräte untereinander. Es ist eine wesentliche Anforderung an ein Automatisierungsgerät, das während eines Steuerbetriebs ein aus mehreren Software-

Funktionsbausteinen gebildetes Steuerprogramm zyklisch und/oder interruptgesteuert bearbeitet, daß diese Software-Funktionsbausteine ladbar und zur Laufzeit des Steuerprogramms in dieses einbindbar ausgebildet sind. Damit diese Anforderung erfüllt ist und die Software-Funktionsbausteine über das INTERNET und die INTERNET-Kommunikationsschnittstelle direkt in ein Automatisierungsgerät ladbar und zur Laufzeit des Steuerprogramms in dieses einbindbar sind, sind die Software-Funktionsbausteine objektorientiert ausgebildet.

Die Software-Funktionsbausteine sind über das INTERNET dynamisch ladbar und erweiterbar, und das Automatisierungsgerät ist mit einer Software-Funktionsbaustein-Ablaufsteuerung (PLC-Object-Engine-System) versehen, die diese Software-Funktionsbausteine in das Steuerprogramm einbindet und während des Steuerbetriebs bearbeitet.

Eine Programmiersprache, die einen objektorientierten Code aus einer Quellsprache erzeugt und für einen Einsatz im INTERNET vorgesehen ist, ist aus dem Buch „Java!“, Tim Ritchey, published 1995 by New Riders Publishing, bekannt. Dort ist eine Quellsprache „JAVA C“ beschrieben, aus der ein objektorientierter Java-Bytecode erzeugbar ist. Weitere vorteilhafte Eigenschaften dieses Sprachmittels sind insbesondere die Portabilität des Codes sowie die Mechanismen zur Fehlerbehandlung. Durch die Portabilität des Codes wird sichergestellt, daß ein Automatisierungsgerät mit einer Ablaufsteuerung in Form eines Java-Bytecode-Interpreters unabhängig von einer Prozessor-Hardware-Architektur des Automatisierungsgerätes (herstellerunabhängig) die dem Automatisierungsgerät über das INTERNET zugeführten Java-Funktionsbausteine bearbeiten kann. Aus Gründen der Performance

ist es allerdings vorteilhaft, das Automatisierungsgerät mit einem Java-Prozessor 12 zu versehen, der den Java-Code direkt verarbeitet.

- 5 Die Portabilität eines objektorientierten Codes wird in Figur 2 verdeutlicht, in der ein Programmübersetzungs-Struktur-
bild dargestellt ist.
- Ein Anwender erstellt mit einem Programmiergerät nach Maßgabe einer zu lösenden Steueraufgabe ein Steuerprogramm in Form
10 eines Kontaktplanes KOP, eines Funktionsplans FuP, einer An-
weisungsliste AWL oder in einer sonstigen geeigneten, z. B.
in der Norm IEC 1131 beschriebenen, Form. Das Programmier-
gerät übersetzt auf Anwenderebene das Steuerprogramm in eine
15 Quellsprache Qu, z. B. in die Quellsprache „JAVA C“, oder di-
rekt (in der Figur mit unterbrochenen Linien dargestellt) in
eine objektorientierte Maschinensprache Ms, z. B. in den
Java-Bytecode, die in Automatisierungsgeräte AG1, AG2, AG3,
AG4 unterschiedlicher Architektur geladen wird. Auf Anwender-
ebene ist zur Erstellung der Maschinencodes nur ein Compiler
20 für alle Automatisierungsgeräte erforderlich. Es ist angenom-
men, daß das Automatisierungsgerät AG4 einen Code-Generator G
für die Verarbeitung der Maschinensprache Ms aufweist, wo-
durch dieses Automatisierungsgerät AG4 den Code direkt ver-
arbeiten kann. Ferner ist angenommen, daß die Automatisie-
25 rungsgeräte AG1, AG2, AG3 nicht mit einem derartigen Code-
Generator versehen sind, sondern unterschiedliche Prozessoren
PR1, PR2, PR3 umfassen. Damit die Automatisierungsgeräte AG1,
AG2, AG3 den Code MS verarbeiten können, sind diese Automati-
sierungsgeräte jeweils mit einem Code-Interpreter IP1, IP2,
30 IP3 versehen. Diese Interpreter IP1, IP2, IP3 erzeugen wäh-

rend der Laufzeit des Steuerprogramms jeweils einen durch die Prozessoren PR1, PR2, PR3 interpretierbaren Code.

Die Programmierung der objektorientierten Software-Funktions-
5 bausteine erfolgt durch die jeweiligen Programmiergeräte 7
(Figur 1) der Fertigungsstandorte 1, 2 oder durch ein eben-
falls an das INTERNET angeschlossenes Programmiergerät 14.
Neben den Bedien- und Beobachtungsgeräten 8 und den Work-
stations 9 sind diese Programmiergeräte 8, 14 Bestandteile
10 des Management-Engineering-Systems. Die Programmiergeräte
führen diese Softwarebausteine den entsprechenden Automati-
sierungsgeräten über die jeweilige INTERNET-Kommunikations-
schnittstelle und das INTERNET zu. Für den Fall, daß z. B.
Bausteine geändert werden müssen, überträgt zunächst das
15 Automatisierungsgerät 6 oder ein Server 13 einem der Pro-
grammiergeräte 7 den entsprechenden Software-Funktionsbau-
stein über das INTERNET. Schließlich ergänzt bzw. modifiziert
das Programmiergerät 7 diesen Baustein und kann ihn wieder in
eines der Automatisierungsgeräte übertragen. Das Programmier-
20 gerät ist ferner mit einem Software-Funktionsbaustein-Ablauf-
system (PLC-Object-Engine-System; Bos, ExE, Wd, IO) versehen,
das zur Simulation des Steuerprogramms vorgesehen ist.

Die Prozeßbedienung und Prozeßführung der zu steuernden Pro-
25 zesse in den Fertigungsstandorten 1, 2 erfolgt durch eben-
falls an das INTERNET anschließbare und am INTERNET betreib-
bare Bedien- und Beobachtungsgeräte 8. Ein Bedien- und Beob-
achtungsgerät 8, z. B. das Bedien- und Beobachtungsgerät 8
des Fertigungsstandortes 1, erzeugt ein Bedien- und Beobach-
30 tungs-Softwarebausteine umfassendes Bedien- und Beobachtungs-

programm zur Erstellung und Darstellung eines mehrere Bild-
objekte umfassenden Prozeßbildes, wobei die Bildobjekte zu
Software-Funktionsbausteinen des Steuerprogramms in Beziehung
(in Wechselwirkung) stehen. Die Bedien- und Beobachtungs-
5 Softwarebausteine sind objektorientiert ausgebildet und di-
rekt über das INTERNET übertragbar. Es ist selbstverständlich
möglich, das Prozeßbild auf dem Programmiergerät 7 zu erstel-
len und zur Prozeßführung über das INTERNET dem Bedien- und
Beobachtungsgerät 8 zuzuführen.

10

Um in Automatisierungssystemen hohen Ausbaugrades die Anzahl
der in ein Automatisierungsgerät eingebauten Ein- und Aus-
gabekomponenten zu verringern, werden dezentrale Subsysteme,
z. B. in Form von intelligenten Feldgeräten, eingesetzt. Das
15 verteilte Automatisierungs- und Management-Engineering-System
weist ein hier nicht dargestelltes intelligentes Feldgerät
auf, welchem mindestens ein Software-Funktionsbaustein eines
Steuerprogramms zuführbar ist, welches das Feldgerät während
eines Steuerbetriebs zyklisch und/oder interruptgesteuert
20 bearbeitet, wobei der Software-Funktionsbaustein ladbar und
zur Laufzeit des Steuerprogramms in dieses einbindbar aus-
gebildet ist. Die Software-Funktionsbausteine sind objekt-
orientiert ausgebildet und über das INTERNET und eine
INTERNET-Kommunikationsschnittstelle des Feldgerätes in
25 dieses ladbar, wobei das Feldgerät ein Software-Funktions-
baustein-Ablaufsystem (PLC-Object-Engine-System; Bos, ExE,
Wd, IO) zur Einbindung des Software-Funktionsbausteins SF01,
..., SF04 und Bearbeitung des Steuerprogramms aufweist.

- Im folgenden wird auf Figur 3 und 4 verwiesen, in denen ein Software-Funktionsbaustein-Ablaufsystem (PLC-Object-Engine-System) eines Automatisierungsgerätes und/oder eines intelligenten Feldgerätes und/oder eines Programmiergerätes (zur Si-
5 mulation eines Steuerprogramms) dargestellt ist. Es ist zunächst angenommen, daß ein Steuerprogramm zyklisch zu bearbeiten ist (Figur 3), was bedeutet, daß unabhängig von den Signalzuständen von Prozeßein- und -ausgängen eines zu steuernden technischen Prozesses z. B. die CPU eines Automati-
10 sierungsgerätes zyklisch
- a. die Signalzustände der Prozeßeingänge abfragt und in einem Prozeßabbild der Eingänge hinterlegt,
 - b. entsprechend den Vorgaben des zu bearbeitenden Steuerprogramms dieses schrittweise abarbeitet und
 - 15 c. die errechneten Signalzustände in einem Prozeßabbild der Ausgänge hinterlegt, wobei diese Signalzustände von dort zu den Prozeßausgängen gelangen.

- Wesentliche Bestandteile des Software-Funktionsbaustein-
20 Ablaufsystems sind objektorientiert programmierte Einheiten in Form eines Bootstraps Bos, eines Ein-/Ausgabe-Moduls IO, eines Exe-Engine-Objekts ExE und eines Watchdogs Wd. Der Watchdog Wd braucht selbstverständlich nicht als Softwaremodul ausgebildet sein, sondern kann hardwaremäßig verwirk-
25 licht werden. In einem praktischen Ausführungsbeispiel der Erfindung sind die Einheiten Exe-Engine-Objekt ExE und Watchdog Wd sogenannte „threads“. Die Funktions- und Wirkungsweise eines „threads“ ist aus der Druckschrift „Supporting Microsoft Windows 95, Student Workbook“, 07/95,
30 der Fa. Microsoft bekannt und braucht daher nicht näher

erläutert zu werden. In der Bootstrap-Einheit Bos sind eine Klasse von Software-Funktionsbausteinen und eine Klasse von Ein-/Ausgabe-Modulen hinterlegt. Diese Klassen werden z. B. von einem Anwender entsprechend den Vorgaben einer zu lösenden Steuerungsaufgabe auf einem Programmiergerät erstellt und z. B. in ein Automatisierungsgerät oder in ein Feldgerät übertragen. Die Bootstrap-Einheit Bos erzeugt vor Beginn des Steuerbetriebes aus der Klasse Software-Funktionsbausteine Software-Funktionsbaustein-Objekte und aus der Klasse Ein-/Ausgabe-Module Ein-/Ausgabe-Modul-Objekte. Im vorliegenden Beispiel sind lediglich vier Software-Funktionsbaustein-Objekte SFO1 ... SFO4 und ein Ein-/Ausgabe-Modul-Objekt IO dargestellt, in welchem ein Prozeßabbild von Ein- und Ausgängen hinterlegt ist und welchem Signalzustände von Prozeßeingängen des technischen Prozesses zuführbar sind und durch welches Signalzustände Prozeßausgängen dieses technischen Prozesses zuführbar sind. Ferner führt die Bootstrap-Einheit Bos zu Beginn des Steuerbetriebs dem Exe-Engine-Objekt ExE eine Liste der zu bearbeitenden Software-Funktionsbaustein-Objekte SFO1 ... SFO4 zu. Zu Beginn des Steuerbetriebes überträgt die Bootstrap-Einheit Bos eine Nachricht Nas (Methodenaufruf), wodurch das Exe-Engine-Objekt ExE gestartet wird. In einem ersten Bearbeitungsschritt führt das Exe-Engine-Objekt ExE dem Watchdog Wd eine Nachricht Naw zu, was bewirkt, daß der Watchdog Wd die Zykluszeit des Exe-Engine-Objekts ExE überwacht. Für den Fall, daß das Exe-Engine-Objekt ExE die vorgesehene Zykluszeit überschreitet, setzt der Watchdog Wd das Exe-Engine-Objekt ExE zurück, indem der Watchdog Wd dem Exe-Engine-Objekt ExE eine Nachricht Nar überträgt. Ferner setzt der Watchdog Wd im Falle der Zyklus-

zeitüberschreitung die Ausgänge des Prozeßabbildes und die Prozeßausgänge zurück, wobei der Watchdog Wd dazu dem Ein-/Ausgabe-Modul-Objekt IO eine Nachricht Nia zuführt.

Nachdem das Exe-Engine-Objekt ExE den Watchdog Wd gestartet
5 hat, nimmt das Exe-Engine-Objekt ExE den Steuerbetrieb auf und führt zunächst einen Bearbeitungszyklus aus, welcher die Schritte umfaßt:

- 10 A) Aktualisieren der Eingänge des Prozeßabbildes, indem das Exe-Engine-Objekt ExE die Eingangssignalzustände des Prozeßabbildes aus dem Ein-/Ausgabe-Modul-Objekt IO ausliest, wobei die Signalzustände durch einen Methodenaufruf Nae zwischen den Objekten ausgetauscht werden,
- 15 B) Bearbeiten jeweils eines Bearbeitungsschrittes der Software-Funktionsbaustein-Objekte SFO1 ... SFO4, wobei das Exe-Engine-Objekt ExE entsprechende Methodenaufrufe NSF1 ... NSF4 den Software-Funktionsbaustein-Objekten SFO1 ... SFO4 zuführt, die Software-Funktionsbaustein-Objekte SFO1 ... SFO4 über Methodenaufrufe Nso auf das Prozeßabbild des Ein-/Ausgabe-Modul-Objekts IO zugreifen, und schließlich
- 20 C) Aktualisieren der Ausgänge des Prozeßabbildes, indem das Exe-Engine-Objekt ExE die Ausgänge des Prozeßabbildes im Ein-/Ausgabe-Modul-Objekt IO einschreibt, wobei wiederum die Signalzustände durch einen Methodenaufruf Naa zwischen den Objekten ausgetauscht werden.

25

Das Ein-/Ausgabe-Modul-Objekt IO versorgt über geeignete Schnittstellen die Prozeßausgänge des zu steuernden technischen Prozesses und - wie beschrieben - die Software-Funktionsbaustein-Objekte SFO1 ... SFO4 mit den Ein- und Ausgangssignalzuständen des Prozeßabbildes.

30

Es wird nun der Fall angenommen, daß ein Steuerprogramm interruptgesteuert zu bearbeiten ist (Figur 4), was bedeutet, daß im Falle von Signalzustandsänderungen an einem der Prozeßeingänge eines zu steuernden technischen Prozesses unverzüglich geeignete Steuermaßnahmen zu ergreifen sind. Die in den Figuren 3 und 4 gleichen Teile sind mit gleichen Bezugszeichen versehen. Im folgenden werden nur die Unterschiede zu der zyklischen Bearbeitung des Steuerprogramms beschrieben (Figur 3). Im Falle der interruptgesteuerten Bearbeitung des Steuerprogramms überträgt die Bootstrap-Einheit Bos zu Beginn des Steuerbetriebs dem Exe-Engine-Objekt ExE nicht wie im Falle einer zyklischen Bearbeitung die Liste der zu bearbeitenden Software-Funktionsbaustein-Objekte, sondern eine Liste von „aufzurufenden“ Software-Funktionsbaustein-Objekten pro Prozeßeingang. Das bedeutet, jedem Prozeßeingang ist ein Software-Funktionsbaustein-Objekt zugeordnet und bei einer Signalzustandsänderung an einem Prozeßeingang ist das diesem Eingang zugeordnete Software-Funktionsbaustein-Objekt zu starten. Im Unterschied zur zyklischen Bearbeitung im Hinblick auf den Bearbeitungsschritt B) führt das Exe-Engine-Objekt ExE während des Bearbeitungszyklus einen Schritt D) aus, welcher umfaßt:

D) Feststellen von Signalzustandsänderungen an den Eingängen des Prozeßabbildes und Bearbeiten der diesen Eingängen zugeordneten Software-Funktionsbaustein-Objekte.

Die in einem Unternehmen anfallenden weiteren Aufgaben wie Materialwirtschaft, Fertigungsplanung, Personaleinsatz usw. sind unter dem Oberbegriff Management-Informationssysteme zusammengefaßt und werden durch ebenfalls an das INTERNET

angeschlossene Workstations oder Server bearbeitet. Sie bedienen sich großer Datenbanken, die persistente objektorientierte Software-Funktionsbausteine als Repräsentanten von Teilprozessen halten und bei Bedarf zum Ablauf bringen.

5

Die Software-Funktionsbausteine des Automatisierungsgerätes sowie des intelligenten Feldgerätes sind im Hinblick auf Entwurf, Projektierung, Programmierung von Automatisierungsaufgaben und Kommunikation zwischen den Komponenten des Automatisierungsverbundes kompatibel zu denen im Management-Engineering-System und zu denen im Management-Engineering- und Informationssystem. Aufgabenerweiterungen bzw. Aufgabenverlagerungen sind dadurch übersichtlich und einfacher als bisher zu bewerkstelligen.

15

Durch die Erfindung wird ein durchgängiges Automatisierungssystem geschaffen, das einen weltweiten Betrieb ermöglicht. Die aus dem Stand der Technik bekannte weitgehende Entkopplung von Automatisierungsgeräten und Standard-Computern wird vermieden. Es wird die Möglichkeit eröffnet, sämtliche Unternehmenselemente wie Prozesse, Ressourcen und Organisationen objektorientiert zu modellieren. Die Implementierung von Software aus diesen Objektmodellen heraus erfolgt über eine einheitliche Werkzeugkette im Rahmen einer durchgängigen Architektur. In der Prozeßoptimierungs- bzw. -änderungsphase sind durch eine Verschiebung von Objekten einfache Softwareanpassungen möglich. Ferner wird eine weltweite, flexible und verteilte Fertigung mit zentraler Planung, Simulation und Optimierung ermöglicht.

30



Patentansprüche

1. Automatisierungsgerät, welchem Software-Funktionsbausteine eines Steuerprogramms zuführbar sind, welches das Automati-
- 5 sierungsgerät während eines Steuerbetriebs zyklisch und/oder interruptgesteuert bearbeitet, wobei die Software-Funktionsbausteine ladbar und zur Laufzeit des Steuerprogramms in dieses einbindbar ausgebildet sind, dadurch gekennzeichnet,
- 10 - daß die Software-Funktionsbausteine (SF01, ..., SF04) objektorientiert ausgebildet und über das INTERNET und eine INTERNET-Kommunikationsschnittstelle des Automatisierungsgerätes in dieses ladbar sind und
- daß das Automatisierungsgerät ein Software-Funktionsbau-
- 15 stein-Ablaufsystem (PLC-Object-Engine-System; Bos, ExE, Wd, IO) zur Einbindung der Software-Funktionsbaustein-Objekte (SF01, ..., SF04) und zur Bearbeitung des Steuerprogramms aufweist.
- 20 2. Automatisierungsgerät nach Anspruch 1, dadurch gekennzeichnet,
- daß das Software-Funktionsbaustein-Ablaufsystem ein Exe-Engine-Objekt (ExE), einen Watchdog (Wd), einen Bootstrap (Bos) und ein Ein-/Ausgabe-Modul-Objekt (IO) umfaßt, in
- 25 welchem ein Prozeßabbild von Ein- und Ausgängen hinterlegbar ist und welchem Signalzustände von Prozeßeingängen zuführbar sind und durch welches Signalzustände Prozeßausgängen zuführbar sind,
- daß der Bootstrap (Bos) vor Beginn des Steuerbetriebes die
- 30 Software-Funktionsbaustein-Objekte (SF01, ..., SF04) und





das Ein-/Ausgabe-Modul-Objekt (IO) erzeugt sowie dem Exe-Engine-Objekt (ExE) zuführt:

- 5 - für den Fall einer zyklischen Bearbeitung des Steuerprogramms eine Liste der zu bearbeitenden Software-Funktionsbaustein-Objekte (SF01, ..., SF04),
- für den Fall einer interruptgesteuerten Bearbeitung des Steuerprogramms eine Liste der zu bearbeitenden Software-Funktionsbaustein-Objekte (SF01, ..., SF04) für jeden Prozeßeingang,
- 10 - daß der Bootstrap (Bos) zu Beginn des Steuerbetriebes das Exe-Engine-Objekt (ExE) startet, welches zunächst den Watchdog (Wd) startet, welcher beim Überschreiten der Zykluszeit das Exe-Engine-Objekt (ExE) zurücksetzt, und anschließend zyklisch
- 15 - die Eingänge des Prozeßabbildes aktualisiert,
- für den Fall einer zyklischen Bearbeitung des Steuerprogramms jeweils einen Bearbeitungsschritt der Software-Funktionsbaustein-Objekte (SF01, ..., SF04) bearbeitet,
- 20 - für den Fall einer interruptgesteuerten Bearbeitung des Steuerprogramms Änderungen von Signalzuständen an den Eingängen feststellt und die diesen Eingängen zugeordneten Software-Funktionsbaustein-Objekte (SF01, ..., SF04) bearbeitet,
- 25 - die Ausgänge des Prozeßabbildes aktualisiert.

3. Automatisierungsgerät nach Anspruch 2, dadurch gekennzeichnet, daß das Exe-Engine-Objekt (ExE) sowie der Watchdog (Wd) als „threads“ ausgebildet sind.





4. Automatisierungsgerät nach einem der Ansprüche 1 bis 3, dadurch gekennzeichnet, daß die Kommunikationsschnittstelle eine TCP/IP-Protokoll-Kommunikation ermöglicht.

- 5 5. Automatisierungsgerät nach einem der Ansprüche 1 bis 4, dadurch gekennzeichnet, daß die Software-Funktionsbausteine (SF01, ..., SF04) Java-bytecodiert sind und in der Programmiersprache „JAVA C“ oder in einer Programmiersprache nach der Norm IEC 1131 erstellbar sind.

10

6. Programmiergerät zur Erstellung von Software-Funktionsbausteinen eines Steuerprogramms, das einem Automatisierungsgerät zuführbar ist, welches während eines Steuerbetriebs das Steuerprogramm zyklisch und/oder interruptgesteuert bearbeitet, wobei die Software-Funktionsbausteine ladbar und zur Laufzeit des Steuerprogramms in dieses einbindbar ausgebildet sind,

dadurch gekennzeichnet,

- daß das Programmiergerät die Software-Funktionsbausteine (SF01, ..., SF04) objektorientiert ausgebildet erstellt,
- daß das Programmiergerät dem Automatisierungsgerät über das INTERNET und eine INTERNET-Kommunikationsschnittstelle des Programmiergerätes die Software-Funktionsbausteine (SF01, ..., SF04) zuführt und/oder
- daß dem Programmiergerät über das INTERNET und die INTERNET-Kommunikationsschnittstelle die Software-Funktionsbausteine (SF01, ..., SF04) zuführbar sind.

7. Programmiergerät nach Anspruch 6, dadurch gekennzeichnet, daß das Programmiergerät zur Simulation des





Steuerprogramms ein Software-Funktionsbaustein-Ablaufsystem (PLC-Object-Engine-System; Bos, ExE, Wd, IO) aufweist.

8. Programmiergerät nach Anspruch 6 oder 7, dadurch gekennzeichnet, daß die Kommunikationsschnittstelle eine TCP/IP-Protokoll-Kommunikation ermöglicht.

9. Programmiergerät nach einem der Ansprüche 6 bis 8, dadurch gekennzeichnet, daß die Software-Funktionsbausteine (SF01, ..., SF04) in der auf dem Programmiergerät ablauffähigen Programmiersprache „JAVA C“ oder in einer Programmiersprache nach der Norm IEC 1131 erstellbar und durch das Programmiergerät Java-bytecodiert übersetzbar sind.

10. Bedien- und Beobachtungsgerät mit Bedien- und Beobachtungs-Softwarebausteinen eines Bedien- und Beobachtungsprogramms zur Erstellung und Darstellung eines mehrere Bildobjekte umfassenden Prozeßbildes, das zur Prozeßführung vorgesehen ist, wobei die Bildobjekte zu Software-Funktionsbausteinen eines Steuerprogramms in Beziehung stehen, welches ein Automatisierungsgerät während eines Steuerbetriebs bearbeitet, wobei die Bedien- und Beobachtungs-Softwarebausteine ladbar und zur Laufzeit des Bedien- und Beobachtungsprogramms in dieses einbindbar ausgebildet sind, dadurch gekennzeichnet,

- daß das Bedien- und Beobachtungsgerät die Bedien- und Beobachtungs-Softwarebausteine objektorientiert ausgebildet erstellt,
- daß die Bedien- und Beobachtungs-Softwarebausteine durch das Bedien- und Beobachtungsgerät über das INTERNET und





eine INTERNET-Kommunikationsschnittstelle des Bedien- und Beobachtungsgerätes übertragbar sind und/oder

- daß dem Bedien- und Beobachtungsgerät über das INTERNET und die INTERNET-Kommunikationsschnittstelle Bedien- und Beobachtungs-Softwarebausteine und/oder Prozeßgrößen zuführbar sind und
- daß das Bedien- und Beobachtungsgerät ein Bedien- und Beobachtungs-Softwarebaustein-Ablaufsystem (Bedien- und Beobachtungs-Object-Engine-System) zur Bearbeitung der Bedien- und Beobachtungs-Softwarebausteine aufweist.

11. Bedien- und Beobachtungsgerät nach Anspruch 10, dadurch gekennzeichnet, daß die Kommunikationsschnittstelle eine TCP/IP-Protokoll-Kommunikation ermöglicht.

15

12. Bedien- und Beobachtungsgerät nach Anspruch 10 oder 11, dadurch gekennzeichnet, daß die Bedien- und Beobachtungs-Softwarebausteine in der auf dem Bedien- und Beobachtungsgerät ablauffähigen Programmiersprache „JAVA C“ oder in einer Programmiersprache nach der Norm IEC 1131 erstellbar und durch das Bedien- und Beobachtungsgerät Java-bytecodiert übersetzbar sind.

13. Intelligentes Feldgerät, welchem mindestens ein Software-Funktionsbaustein eines Steuerprogramms zuführbar ist, welches das Feldgerät während eines Steuerbetriebs zyklisch und/oder interruptgesteuert bearbeitet, wobei der Software-Funktionsbaustein ladbar und zur Laufzeit des Steuerprogramms in dieses einbindbar ausgebildet ist, dadurch gekennzeichnet,



- daß der Software-Funktionsbaustein (SF01, ..., SF04) objektorientiert ausgebildet und über das INTERNET und eine INTERNET-Kommunikationsschnittstelle des Feldgerätes in dieses ladbar ist und
 - 5 - daß das Feldgerät ein Software-Funktionsbaustein-Ablaufsystem (PLC-Object-Engine-System; Bos, ExE, Wd, IO) zur Einbindung des Software-Funktionsbausteins (SF01, ..., SF04) und Bearbeitung des Steuerprogramms aufweist.
- 10 14. Intelligentes Feldgerät nach Anspruch 13, dadurch gekennzeichnet,
- daß das Software-Funktionsbaustein-Ablaufsystem ein Exe-Engine-Objekt (ExE), einen Watchdog (Wd), einen Bootstrap (Bos) und ein Ein-/Ausgabe-Modul-Objekt (IO) umfaßt, in
 - 15 welchem ein Prozeßabbild von Ein- und Ausgängen hinterlegbar ist und welchem Signalzustände von Prozeßeingängen zuführbar sind und durch welches Signalzustände Prozeßausgängen zuführbar sind,
 - daß der Bootstrap (Bos) vor Beginn des Steuerbetriebes die
 - 20 Software-Funktionsbaustein-Objekte (SF01, ..., SF04) und das Ein-/Ausgabe-Modul-Objekt (IO) erzeugt sowie dem Exe-Engine-Objekt (ExE) zuführt:
 - für den Fall einer zyklischen Bearbeitung des Steuerprogramms eine Liste der zu bearbeitenden Software-
 - 25 Funktionsbaustein-Objekte (SF01, ..., SF04),
 - für den Fall einer interruptgesteuerten Bearbeitung des Steuerprogramms eine Liste der zu bearbeitenden Software-Funktionsbaustein-Objekte (SF01, ..., SF04) für jeden Prozeßeingang,



- 5 - daß der Bootstrap (Bos) zu Beginn des Steuerbetriebes das Exe-Engine-Objekt (ExE) startet, welches zunächst den Watchdog (Wd) startet, welcher beim Überschreiten der Zykluszeit das Exe-Engine-Objekt (ExE) zurücksetzt, und anschließend zyklisch
- 10 - die Eingänge des Prozeßabbildes aktualisiert,
- 10 - für den Fall einer zyklischen Bearbeitung des Steuerprogramms jeweils einen Bearbeitungsschritt der Software-Funktionsbaustein-Objekte (SF01, ..., SF04) bearbeitet,
- 15 - für den Fall einer interruptgesteuerten Bearbeitung des Steuerprogramms Änderungen von Signalzuständen an den Eingängen feststellt und die diesen Eingängen zugeordneten Software-Funktionsbaustein-Objekte (SF01, ..., SF04) bearbeitet,
- 15 - die Ausgänge des Prozeßabbildes aktualisiert.
- 20 15. Intelligentes Feldgerät nach Anspruch 14, dadurch gekennzeichnet, daß das Exe-Engine-Objekt (ExE) sowie der Watchdog (Wd) als „threads“ ausgebildet sind.
- 25 16. Intelligentes Feldgerät nach einem der Ansprüche 13 bis 15, dadurch gekennzeichnet, daß die Kommunikationsschnittstelle eine TCP/IP-Protokoll-Kommunikation ermöglicht.
- 30 17. Intelligentes Feldgerät nach einem der Ansprüche 13 bis 16, dadurch gekennzeichnet, daß die Software-Funktionsbausteine (SF01, ..., SF04) Java-bytecodiert sind und in der Programmiersprache „JAVA C“ oder in einer Programmiersprache nach der Norm IEC 1131 erstellbar sind.



18. Automatisierungssystem

- mit mindestens einem Automatisierungsgerät nach einem der Ansprüche 1 bis 5,
- mit mindestens einem Programmiergerät nach einem der Ansprüche 6 bis 9 und/oder
- mit mindestens einem Bedien- und Beobachtungsgerät nach einem der Ansprüche 10 bis 12.

19. Automatisierungssystem nach Anspruch 18 mit mindestens einem intelligenten Feldgerät nach einem der Ansprüche 13 bis 17.

20. Automatisierungsverbund

- mit einem Automatisierungssystem nach Anspruch 18 oder 19 und
- mit mindestens einer Workstation und/oder einem Server, welche Mittel zum Erstellen und Bearbeiten von objekt-orientierten Software-Funktionsbausteinen (SF01, ..., SF04) aufweisen.