



(10) **DE 10 2013 002 085 A1** 2014.08.07

(12) **Offenlegungsschrift**

(21) Aktenzeichen: **10 2013 002 085.9**

(22) Anmeldetag: **06.02.2013**

(43) Offenlegungstag: **07.08.2014**

(51) Int Cl.: **G05B 19/05 (2006.01)**

(71) Anmelder:

Robert Bosch GmbH, 70469, Stuttgart, DE

(72) Erfinder:

**Schultze, Stephan, Dr., 97816, Lohr, DE; Osterfeld,
Henning, 97816, Lohr, DE**

(56) Ermittelter Stand der Technik:

DE 10 2011 108 964 A1

US 2004 / 0 138 787 A1

US 2011 / 0 172 838 A1

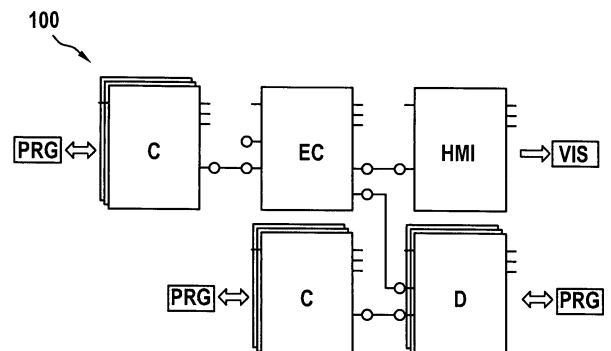
EP 2 574 996 A1

Rechercheantrag gemäß § 43 Abs. 1 Satz 1 PatG ist gestellt.

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

(54) Bezeichnung: **SPS-Funktionsbausteine für Energieverwaltungsfunktionalitäten**

(57) Zusammenfassung: Die Erfindung betrifft ein Verfahren zum Programmieren einer Speicherprogrammierbaren Steuerung zur Steuerung wenigstens eines Energieverbrauchers einer Maschine, wobei zur Programmierung eines Energieverwaltungsprogramms zur Bereitstellung wenigstens einer Energieverwaltungsfunktionalität für die Maschine in einem Programmiergerät zur Auswahl durch einen Programmierer wenigstens ein Energieverbraucher-Funktionsbaustein (D) und wenigstens ein Energiesteuerungs-Funktionsbaustein (EC) bereitgestellt werden, wobei der wenigstens eine Energieverbraucher-Funktionsbaustein (D) für den wenigstens einen Energieverbraucher spezifische Daten und wobei der wenigstens eine Energiesteuerungs-Funktionsbaustein (EC) für die wenigstens eine Energieverwaltungsfunktionalität spezifische Daten enthält.



Beschreibung

[0001] Die vorliegende Erfindung betrifft ein Verfahren zum Programmieren einer Speicherprogrammierbaren Steuerung zur Steuerung wenigstens eines Energieverbrauchers einer Maschine.

Stand der Technik

[0002] Die Erfindung befasst sich mit der vereinfachten Implementierung einer Energieverwaltung in Maschinen mit Speicherprogrammierbaren Steuerungen (SPS), englisch Programmable Logic Controller (PLC), insbesondere mit Möglichkeiten zur einfacheren Programmierung einer SPS.

[0003] In Systemen, die mittels Energieverwaltung bzgl. des Energieverbrauchs optimiert werden sollen, ist mindestens ein Energieverbraucher vorhanden, der in verschiedenen Energiemodi unterschiedlich viel Energie verbraucht, wobei die Energiemodi vorgebar sind. Beispiele hierfür sind drehzahlsteuerbare Gebläse, die in Betriebspausen ausgeschaltet werden, Servoantriebe, deren Leistungsversorgung während des Stillstandes abgeschaltet wird, oder Heizungen, die in Betriebspausen mit einem geringeren Temperatursollwert gefahren werden. Energiemodi werden durch viele Kennzahlen charakterisiert. Dies sind z. B. Energieverbrauch im jeweiligen Modus, Zeitbedarf zum Übergang zwischen den Modi, Energiebedarf während des Übergangsvorgangs, Mindest-/Maximalverweildauern. Daneben gibt es bei Kommunikationssystemen sogenannte Energieprofile, d. h. Vereinbarungen über spezifizierte Daten und deren Formate. Als Beispiele seien hier "sercos Energy" oder "PROFIenergy" genannt.

[0004] Innerhalb der energieverwaltenden Steuerungen werden diese Zustände in der Energieverwaltungsinstanz (Programmteil) behandelt. Hierzu gibt es zwei grundsätzliche Energieverwaltungsfunktionalitäten, Energiemonitoring (Energieverbrauchsbestimmung) und Energiemodusvorgabe. Diese Funktionalitäten können z. B. in SPSen mittels Funktionsbausteinen (FB) abgebildet werden, die im Kontext der SPS (PLC-Programm) abgearbeitet werden. Funktionsbausteine enthalten üblicherweise jeweils eine oder mehrere technologisch oder funktionell zusammengehörende Programmanweisungen. Ein vollständiges SPS-Programm setzt sich in der Regel aus mehreren Funktionsbausteinen zusammen, wobei die einzelnen Bausteine in unterschiedlichen SPS-anwendungsbezogenen Sprachen eingegeben sein können. Funktionsbausteine können bei jedem Aufruf (sog. Instanzen) mittels zugeordneter Instanzdatenbausteine (Instanz-DB) mit unterschiedlichen Daten versorgt werden.

[0005] Die gesamte Energieverwaltung kann als Programm oder FB in Verbindung mit einer dekla-

rierten Datenstruktur mit allen benötigten Energieverwaltungsdaten implementiert werden. Die Daten können dabei in einer globalen Variablenliste deklariert werden, d. h. diese sind für alle SPS-Programmteile sichtbar und änderbar.

[0006] Eine geeignete Datenstruktur kann Konfigurationsdaten der Maschine (z. B. Liste aller Energiemodi der Maschine), Konfigurationsdaten der Energieverbraucher (sog. Devices) (z. B. Liste aller Energiemodi der Devices mit deren Energieverbräuchen und Übergangszeiten), HMI-Daten (z. B. Ist-Energieverbrauch der Devices), Steuerdaten des Energiemanagements (z. B. Soll-Energiemodus, Pausenzeit), Statusdaten des Energiemanagements (z. B. Ist-Energiemodus der Maschine bzw. aller Devices) beinhalten.

[0007] Die Schnittstelle zum (überlagerten) Anwenderprogramm bzgl. Logik der Energiemodusverwaltung etc. kann dabei über die Datenstruktur gehandhabt werden. Ebenso kann die Schnittstelle zu den (unterlagerten) Devices bzgl. Steuer/Status der Devicezustände über die Datenstruktur gehandhabt werden (z. B. mittels eines Device-Anwenderprogramms). Auch die Schnittstelle zur HMI kann über die Datenstruktur gehandhabt werden.

[0008] Die Datenstruktur kann dabei alle Daten für alle Energiemodi bzw. Devices dergestalt beinhalten, dass die Datenstruktur von Anfang an die gesamte Datenmenge für den Maximalausbau beinhaltet. D. h. der Speicherbedarf ist auf den definierten Maximalausbau festgelegt.

[0009] Nachteilig an einer solchen Lösung ist insbesondere eine fehlende Skalierbarkeit. Auch ist die Datenstruktur in den meisten Fällen viel zu groß, da die betriebene Maschine meist nicht maximal ausgebaut ist. Besonders nachteilig ist, dass es einen riesigen und unüberschaubaren Variablenpool gibt, bei dem der Programmierer leicht den Überblick verliert und unschlüssig ist, welche Variablen er schreiben bzw. lesen darf und welche Variablen nur intern verwendet werden. Die Implementierung zusätzlicher Funktionalitäten ist sehr aufwendig.

[0010] Es ist daher wünschenswert, die Programmierung der Energieverwaltung einer Maschine zu vereinfachen.

Offenbarung der Erfindung

[0011] Erfindungsgemäß wird ein Verfahren zum Programmieren einer Speicherprogrammierbaren Steuerung zur Steuerung wenigstens eines Energieverbrauchers einer Maschine mit den Merkmalen des Patentanspruchs 1 vorgeschlagen. Vorteilhafte Ausgestaltungen sind Gegenstand der Unteransprüche sowie der nachfolgenden Beschreibung.

Vorteile der Erfindung

[0012] Im Rahmen der Erfindung wird vorgeschlagen, ein Energieverwaltungsprogramm zur Bereitstellung wenigstens einer Energieverwaltungsfunktionalität durch mehrere unterschiedliche Funktionsbausteine zu implementieren. Es werden zumindest ein Energieverbraucher-FB (sog. Device-FB) und ein Energiesteuerungs-FB bereitgestellt, die ein Programmierer bei der Programmierung des Energieverwaltungsprogramms einsetzen kann. Dabei findet eine Trennung von energieverbraucher-spezifischen Daten, die in die Device-FBs implementiert werden, und energiefunktionalitätsspezifischen (also energieverbraucherunabhängigen) Daten, die in die Energiesteuerungs-FBs implementiert werden, statt. Dies vereinfacht die Programmierung signifikant und macht insbesondere fertige Programme übersichtlicher und leichter erfassbar.

[0013] Die Erfindung erleichtert die Programmierung der Energieverwaltung einer Maschine mit wenigstens einer SPS, da das Programm vom Programmierer aus vordefinierten FBs zu einer FB-Struktur zusammengesetzt werden kann. Das Programm ist durch Hinzufügen bzw. Zusammenschalten von FBs beliebig skalierbar. Die Struktur der Energieverwaltung ist durch erweiterbare Hierarchie- bzw. Gliederungsebenen flexibel.

[0014] Es ist vorgesehen, dass ein Device-FB programmtechnisch dazu eingerichtet ist, einen oder mehrere Energieverbraucher anzusteuern, d. h. die Ausgabe entsprechender Ansteuerbefehle bzw. Ansteuersignale durch die SPS auszulösen. Die Ansteuerung eines Energieverbrauchers kann vorzugsweise auf zwei Arten erfolgen. Gemäß einer ersten Variante beinhaltet der Device-FB intern bereits die direkte Steuerung des Energieverbrauchers. Diese Lösung ist sehr komfortabel, allerdings muss hierfür dem Device-FB die Funktionalität des Energieverbrauchers im Detail bekannt sein. Dies wird beispielsweise über sogenannte Energieprofile in Feldbussystemen ermöglicht (z. B. sercos Energy, PROFInergy oder CIPenergy), bei denen die Energieverbraucher über standardisierte Energiemodi verfügen. Ebenso ist vorteilhaft, wenn der Device-FB alle für die Ansteuerung notwendigen Informationen (Daten, Parameter usw.) vom Energieverbraucher abrufen kann. Dieser ist dazu insbesondere mit einer Speichereinrichtung ausgestattet, in der die Daten abgespeichert sind.

[0015] Ist diese Möglichkeit der generischen Behandlung von Energieverbrauchern nicht möglich, ist eine zweite Variante vorteilhaft, bei der der Device-FB Schnittstellen mit Daten, die vom Programmierer bedient werden müssen, aufweist. Die Verbindung zwischen Energieverbraucher und Device-FB ist in

diesem Fall vom Programmierer auszuprogrammieren.

[0016] Vorzugsweise ist ein Device-FB programmtechnisch dazu eingerichtet, eine Statusinformation (z. B. eine Rückmeldung darüber, in welchem Energiemodus sich der Energieverbraucher befindet) von einem angesteuerten Energieverbraucher zu empfangen bzw. auszulesen und diese (oder eine daraus generierte) Statusinformation auslesbar anzubieten oder weiter zu übertragen, so dass sie im Rahmen der Energieverwaltung weiterverarbeitet werden kann.

[0017] Es ist weiter vorgesehen, dass die die Energiesteuerungs-FBs programmtechnisch dazu eingerichtet sind, die Energieverwaltung zu steuern. Die Steuerung der Energieverwaltung beinhaltet vorzugsweise ausgabeseitig die Koordination bzw. Steuerung der Device-FBs, insbesondere zum physikalischen Prozess hin. Vorzugsweise beinhaltet dies die Ermittlung und Vorgabe der jeweiligen (einzelnen) Energiemodi, die die Energieverbraucher einnehmen sollen, an die zugehörigen Device-FBs.

[0018] Vorzugsweise verfügt ein Energiesteuerungs-FB über wenigstens eine Schnittstelle, über die eingabeseitig Befehle empfangen werden, insbesondere zum Vorgeben eines Energiemodus der Maschine bzw. eines oder mehrerer Energieverbraucher.

[0019] Vorzugsweise sind für den Datenaustausch zwischen den FBs entsprechende Schnittstellen (insbesondere Methoden) vordefiniert. Dies wird am Ende der Figurenbeschreibung detailliert erläutert.

[0020] Vorzugsweise werden zusätzlich Befehls-FBs (Command-FBs) vorgesehen, die programmtechnisch dazu eingerichtet sind, Betriebsbefehle an die Energiesteuerungs-FBs und/oder Device-FBs auszugeben. Beispielsweise umfassen solche Betriebsbefehle die allgemeine Vorgabe von Energiemodi (z. B. Art, Zeiten wie Dauer, Beginn, Ende, ...). Weiter vorzugsweise umfassen solche Betriebsbefehle auch die Vorgabe von Rahmenbedingungen für Unterbrechungen (wie z. B. Länge der Pause, Mindestpausendauer, zulässige Reaktivierungszeiten usw.) anstelle eines konkreten Energiemodus. Hinsichtlich weiterer Details sei auf die nachveröffentlichte DE 10 2012 025 194.7 verwiesen, deren Offenbarung hier einbezogen wird. Diese Betriebsbefehle sind für ungeplante Pausen besonders vorteilhaft. Die Energiesteuerungs-FBs sind vorzugsweise programmtechnisch dazu eingerichtet, diese Betriebsbefehle entgegenzunehmen, daraus Betriebsbefehle für die Energieverbraucher zu erzeugen und diese an die Device-FBs zu übermitteln, welche schließlich die Energieverbraucher entsprechen ansteuern. Beispielsweise werden die Betriebsbefehle für die Energieverbraucher so erzeugt, dass diese in einen

die Rahmenbedingungen erfüllenden Energiemodus versetzt werden.

[0021] Vorzugsweise werden zusätzlich Eingabe/Ausgabe-FBs (HMI-FBs) vorgesehen, welche die Mensch-Maschinen-Schnittstelle für die Bedienung und den Betrieb der Maschine enthalten. Zweckmäßigerweise sind sie programmtechnisch dazu eingerichtet, energieverwaltungsrelevante Daten zu Visualisierungszwecken zu sammeln und zu gruppieren und/oder Benutzereingaben entgegenzunehmen und diese beispielsweise an die Command-FBs weiterzuleiten. Typischerweise werden für Ein-/Ausgabe-Zwecke Daten, wie z. B. aktueller Energiemodus der Gesamtmaschine, Energiemodi der einzelnen Verbraucher, aktueller Übergangszustand ("Ziel-Energiemodus erreicht" oder auch "im Übergangszustand"), aktueller Ziel-Energiemodus, aktive (Rest-) Pausenzeit und Fehlerzustände bzw. -Informationen bereitgestellt.

[0022] Die Implementierung der Erfindung in Form von Software ist besonders vorteilhaft, da dies besonders geringe Kosten ermöglicht, insbesondere wenn eine ausführende Recheneinheit noch für weitere Aufgaben genutzt wird und daher ohnehin vorhanden ist. Geeignete Datenträger zur Bereitstellung des Computerprogramms sind insbesondere Disketten, Festplatten, Flash-Speicher, EEPROMs, CD-ROMs, DVDs u. a. m. Auch ein Download eines Programms über Computernetze (Internet, Intranet usw.) ist möglich. Die Programmierung von SPSen geschieht üblicherweise mittels einer entsprechenden Software auf einem Programmiergerät (z. B. eine Anwendung unter Microsoft Windows oder Linux auf einem PC, oder ein zugeschnittenes Programmiersystem). Eine erfindungsgemäße Recheneinheit, z. B. ein Programmiergerät einer SPS, ist, insbesondere programmtechnisch, dazu eingerichtet, ein erfindungsgemäßes Verfahren durchzuführen.

[0023] Weitere Vorteile und Ausgestaltungen der Erfindung ergeben sich aus der Beschreibung und der beiliegenden Zeichnung.

[0024] Es versteht sich, dass die vorstehend genannten und die nachfolgend noch zu erläuternden Merkmale nicht nur in der jeweils angegebenen Kombination, sondern auch in anderen Kombinationen oder in Alleinstellung verwendbar sind, ohne den Rahmen der vorliegenden Erfindung zu verlassen.

[0025] Die Erfindung ist anhand von Ausführungsbeispielen in der Zeichnung schematisch dargestellt und wird im Folgenden unter Bezugnahme auf die Zeichnung ausführlich beschrieben.

Figurenbeschreibung

[0026] Fig. 1 zeigt grob schematisch eine bevorzugte Ausführungsform eines erfindungsgemäßen SPS-Programmiergeräts.

[0027] Fig. 2 zeigt ein Programmschema eines Energieverwaltungsprogramms, welches sich aus mehreren unterschiedlichen Funktionsbausteinen zusammensetzt.

[0028] Fig. 3 zeigt ein einfaches Programmschema eines Energieverwaltungsprogramms, welches sich aus drei unterschiedlichen Funktionsbausteinen zusammensetzt.

[0029] Fig. 4 zeigt ein erweitertes Programmschema eines Energieverwaltungsprogramms, welches u. a. zwei Energiesteuerungs-Funktionsbausteine aufweist.

Detaillierte Beschreibung der Zeichnung

[0030] In Fig. 1 ist eine bevorzugte Ausführungsform eines SPS-Programmiergeräts **10** grob schematisch dargestellt, welches programmtechnisch zur Durchführung eines erfindungsgemäßen Verfahrens eingerichtet ist. Bei dem SPS-Programmiergerät **10** kann es sich beispielsweise um einen herkömmlichen Computer handeln, auf dem eine SPS-Programmiersoftware ausgeführt wird und der eine Programmierumgebung **20** bereitstellt.

[0031] Die Figur zeigt den Programmiervorgang einer SPS **40** durch einen Programmierer unter Verwendung des SPS-Programmiergeräts **10**.

[0032] Das SPS-Programmiergerät **10** ist über eine entsprechende Verbindung **30**, beispielsweise eine Ethernet-Verbindung oder eine serielle Verbindung, mit der zu programmierenden SPS **40** verbunden. Die SPS **40** verfügt zumindest über eine Speichereinrichtung **41** zum Aufnehmen des erzeugten SPS-Anwenderprogramms sowie über eine CPU **42** zum Ausführen des gespeicherten SPS-Anwenderprogramms. Die SPS **40** ist weiterhin mit Sensoren **43** sowie Aktoren **44** verbunden, beispielsweise mittels einer Feldbus-Verbindung, insbesondere einer Ethernet-Feldbusverbindung wie z. B. SERCOS III, um gemäß dem SPS-Anwenderprogramm eine Maschine (nicht gezeigt) anzusteuern. Ein Sensor ist bspw. ein Temperatur-, Druck-, Drehzahl-, Lagesensor usw., jedoch auch ein Schalter, Drehknopf usw., und ist zur Erfassung von Eingangsgrößen (z. B. Temperatur, Druck, Drehzahl, Position, Stellung eines Schalters oder eines Drehknopfes) vorgesehen. Von dem SPS-Anwenderprogramm werden daraus Ausgangsgrößen (z. B. Temperatur, Druck, Drehzahl, Stromstärke, Spannung, Einspritzmenge, Position usw.) berechnet und mittels der Aktoren (z. B. elektronischer Schal-

ter, Relais, Ventil usw.) eingestellt. Die Sensoren und Aktoren sind üblicherweise Bestandteil der Maschine. Die Aktoren dienen insbesondere zur Ansteuerung von Energieverbrauchern.

[0033] Die Programmierumgebung **20** ist eine programmtechnische Lösung, die dem Programmierer eine Benutzeroberfläche für die Erstellung des Anwenderprogramms, hier insbesondere mit einer Energieverwaltungsfunktionalität bereitstellt. Dabei definiert der Programmierer das Anwenderprogramm bspw. durch Hinzufügen von Programmcode, Funktionsbausteinen, Dokumentationsdaten und Konfigurationsdaten. Das Programm wird in der auf einem Bildschirm **11** des Programmiergeräts **10** dargestellt. Bevorzugte Ausführungsformen von Energieverwaltungsprogrammen sind in den **Fig. 2** bis **Fig. 4** dargestellt.

[0034] In **Fig. 2** ist schematisch ein Ausschnitt **100** eines Energieverwaltungsprogramms für die Energieverwaltung einer Maschine umfassend mehrere Energieverbraucher gezeigt, wie es insbesondere in einer Programmierumgebung für eine SPS dargestellt sein kann. Dargestellt sind eine Anzahl von mit "C" bezeichneten Command-FBs, ein mit "EC" bezeichneter Energiesteuerungs-FB, eine Anzahl von mit "D" bezeichneten Device-FBs und ein mit "HMI" bezeichneter HMI-FB. Die FBs sind mit anderen Teilen des SPS-Programms verknüpft, welche mit "PRG" bezeichnet sind.

[0035] Zentraler Bestandteil der Energieverwaltung ist der Energiesteuerungs-FB EC, der programmtechnisch dazu eingerichtet ist, die gesamten Abläufe der Energieverwaltungs-Vorgabe zu steuern. Unterlagert sind die Device-FBs, die die Schnittstelle zu den Energieverbrauchern darstellen. Die Energieverbraucher werden vom Energiesteuerungs-FB gesteuert bzw. verwaltet, d. h. der Energiesteuerungs-FB gibt beispielsweise den gewünschten Energiemodus an einen Device-FB, der diesen dann für die ihm zugehörigen Geräte umsetzt.

[0036] Die Command-FBs können einerseits auf der Ebene des Energiesteuerungs-FBs Vorgaben für die gesamte Maschine machen, oder optional auch auf Device-Ebene direkte Vorgaben für den Energiemodus eines Device-FBs machen.

[0037] Der HMI-FB ist programmtechnisch dazu eingerichtet, mit "VIS" bezeichnete Visualisierungen auszugeben, beispielsweise auf einem Bildschirm für einen Benutzer. Der HMI-FB dient dazu, für (unterschiedliche) HMI-Zwecke Daten zu sammeln und gruppiert zur Verfügung zu stellen. Die benötigten Daten sind dabei bei objektorientiertem Programmierungskonzept innerhalb der deklarierten Daten der jeweiligen FBs enthalten (jede FB-Instanz bringt dabei ihre eigenen Daten mit). Ein HMI-FB dient so-

mit – genauso wie ein Leitsystem-FB (nicht in **Fig. 2** dargestellt) – dazu, diese verteilten Daten für den Maschinenbediener darzustellen. Dies ist besonders bei einem objektorientierten Programmierungskonzept vorteilhaft, da hierbei auf diese Daten, die innerhalb der jeweiligen FBs enthalten sind, nicht von außen zugegriffen werden können soll. Vorzugsweise werden hierbei die Daten nur mittels sog. Methoden bereitgestellt (Kapselung der Daten). Zweckmäßigerweise wird ein eigener HMI-FB für jede Gruppierung von Daten eingesetzt. Beispielsweise ist der Energiesteuerungs-FB programmtechnisch dazu eingerichtet, Daten an den HMI-FB auszugeben, welche der Energiesteuerungs-FB beispielsweise aus Statusmeldungen generiert, die er von den Device-FBs erhält.

[0038] Die Erfindung führt zu einer flexiblen und beliebig skalierbaren Programmarchitektur, da insbesondere jeweils genauso viele FBs eingesetzt werden können wie benötigt. Insbesondere können demnach ein oder mehrere Device-FBs, ein oder mehrere Command-FBs, kein, ein oder mehrere HMI-FBs und ein oder mehrere (vgl. **Fig. 4**) Energiesteuerungs-FBs eingesetzt und zusammenschaltet werden. Gleichzeitig kann eine Hierarchie implementiert werden, indem die FBs auf unterschiedlichen Ebenen angeordnet werden.

[0039] Im Folgenden soll nun unter Bezugnahme auf **Fig. 3**, in der eine einfache Ausführungsform **200** eines Energieverwaltungsprogramms mit lediglich einem Command-FB "C", einem Energiesteuerungs-FB "EC" und einem Device-FB "D" gezeigt ist, ein Beispiel einer Energieverwaltung für eine Maschine umfassend einen Verbraucher erläutert werden.

[0040] Als Beispiel gibt der Command-FB "C" eine Länge einer anstehenden Betriebspause (Pausenzeit) vor. Der Energiesteuerungs-FB "EC" ist programmtechnisch dazu eingerichtet, anhand seiner internen Datenbasis einen dazu passenden Verbraucher-Energiemodus auszuwählen. Dabei wird zweckmäßigerweise anhand von Energiesparattributen (wie z. B. minimale Pausenzeit, Energieverbrauch eines Energiemodus, ...) der jeweils optimale Verbraucher-Energiemodus ausgewählt, um z. B. die Leerlaufverluste in den Pausenzeiten auf ein Minimum zu reduzieren. Der Energiesteuerungs-FB "EC" ist programmtechnisch dazu eingerichtet, den ausgewählten Verbraucher-Energiemodus an den Device-FB auszugeben, welcher wiederum programmtechnisch dazu eingerichtet, den Verbraucher entsprechend anzusteuern, so dass er den gewünschten Verbraucher-Energiemodus einnimmt.

[0041] Gemäß einem anderen Beispiel gibt der Command-FB "C" einen Maschinen-Energiemodus für die Maschine vor. Der Energiesteuerungs-FB "EC" ist programmtechnisch dazu eingerichtet, anhand seiner internen Datenbasis einen dazu pas-

senden Verbraucher-Energiemodus auszuwählen. Beispielweise beinhaltet die Datenbasis des Energiesteuerungs-FB bereits eine Verknüpfung zwischen Maschinen-Energiemodus und Verbraucher-Energiemodus. Der Energiesteuerungs-FB ist programmtechnisch dazu eingerichtet, den ausgewählten Verbraucher-Energiemodus an den Device-FB auszugeben, welcher wiederum programmtechnisch dazu eingerichtet, den Verbraucher entsprechend anzusteuern, so dass er den gewünschten Verbraucher-Energiemodus einnimmt.

[0042] Die Verschaltung der FBs dient dazu, die jeweiligen FBs mit ihren kooperierenden FBs zu verknüpfen. Dadurch kann insbesondere auch der Datenaustausch zwischen den kooperierenden FBs automatisiert werden. Die Verschaltung der FBs kann dabei beispielsweise dadurch erfolgen, dass die FBs programmtechnisch dazu eingerichtet sind, sich gegenseitig an den Eingängen mitzuteilen, welche unterlagerten bzw. überlagerten FBs vorhanden sind. Eine alternative Art der Verschaltung umfasst das Angeben des jeweilig unterlagerten FBs in der Deklaration eines übergeordneten FBs. Dies ermöglicht, dass sich die FBs z. B. während einer Initialisierungsphase des fertigen SPS-Programms gegenseitig melden und dadurch ihre jeweiligen Instanzen kennen, um mit diesen dann Daten auszutauschen. Die genannten Ausführungsformen bieten den Vorteil, dass sich der Programmierer keine Verschaltung im SPS-Programmcode programmieren muss.

[0043] Im Folgenden soll nun unter Bezugnahme auf **Fig. 4**, in der eine Ausführungsform **300** eines Energieverwaltungsprogramms mit mehreren Command-FB "C", zwei Energiesteuerungs-FB "EC1" und "EC2" und mehreren Device-FBs "D" gezeigt ist, ein Beispiel des Einsatzes mehrerer Energiesteuerungs-FB für die Energieverwaltung einer Maschine erläutert werden. Die zwei Energiesteuerungs-FB "EC1" und "EC2" sind hierarchisch zusammengeschaltet, wobei der untergeordnete Energiesteuerungs-FB "EC2" programmtechnisch dazu eingerichtet ist, mehrere Device-FBs D zu gruppieren und als gemeinsame Ressource wie einen einzigen Device-FB dem übergeordneten Energiesteuerungs-FB "EC1" darzustellen, also einen Device-FB zu simulieren. Zusätzlich (nicht gezeigt) können dem übergeordneten Energiesteuerungs-FB "EC1" weitere Energiesteuerungs-FBs und/oder Device-FBs untergeordnet sein. Die Gruppierung mittels untergeordneter Energiesteuerungs-FBs kann besonders vorteilhaft zur Strukturierung von Maschinenaggregaten verwendet werden, wobei jedes Maschinenaggregat mehrere Energieverbraucher beinhaltet und der übergeordnete Energiesteuerungs-FB nur noch eine Sicht auf Maschineaggregat besitzt.

[0044] Die FBs sind gemäß einer bevorzugten Ausgestaltung der Erfindung programmtechnisch dazu

eingerichtet, ihre jeweiligen angeschlossenen (unterlagerten bzw. überlagerten) FBs zu kennen. Dadurch können sie direkt ohne zusätzliches Anwenderprogramm Daten mit diesen FBs austauschen. Dies erspart dem Programmierer, einen Programmcode zu implementieren, um die Daten der jeweiligen FBs selbst zwischen den FBs umzukopieren. Beispielsweise würden ansonsten FB-Ausgänge (Statusmeldungen) eines Device-FBs vom Programmierer an Eingänge eines Energiesteuerungs-FBs umkopiert werden müssen.

[0045] Im Falle objektorientierter Programmierung mit der oben dargestellten Definition der Verschaltungsstruktur bei der FB-Deklaration melden sich die FBs entsprechend der Deklaration aneinander an und kennen dadurch die jeweiligen korrespondierenden FBs. Dies wird anhand eines Beispiels erläutert: fbBasicDevice: IL_4EE_BasicDeviceType01(fbEnergyControl);

[0046] Durch die Deklaration ist dem Device-FB "fbBasicDevice" bekannt, welcher Instanz des Energiesteuerungs-FBs er zugeordnet ist, nämlich "fbEnergyControl". Hierdurch wird eine unidirektionale Zuordnung geschaffen.

[0047] Der Device-FB meldet sich innerhalb seiner Initialisierungsphase am übergeordneten Energiesteuerungs-FB an, so dass damit wiederum dem Energiesteuerungs-FB der unterlagerte Device-FB bekannt ist. Diese Anmeldung wird zweckmäßigerweise mittels objektorientierter Methoden durchgeführt. Der Energiesteuerungs-FB beinhaltet dafür vorzugsweise eine Methode (d. h. Aufrufschnittstelle), die innerhalb der Initialisierungsphase vom Device-FB aufgerufen wird. Dies ist besonders vorteilhaft, da die Methoden dem Programmierer nicht bekannt sein müssen, da sie nur von den FBs zur internen Kommunikation verwendet werden. Vorteilhafterweise werden alternativ oder zusätzlich auch die Daten der zyklischen Übertragung mittels (insbesondere nicht öffentlich bekannter) Methoden übertragen.

[0048] Insgesamt kann somit jeglicher Datenaustausch innerhalb des Energiemanagements verborgen ausgeführt werden. Lediglich die Schnittstellen zum Anwenderprogramm (vgl. PRG in **Fig. 2**) müssen dem Programmierer bekannt sein, da er diese in seinem Anwenderprogramm bedienen muss.

[0049] Alternativ oder zusätzlich kann ein FB nicht-dokumentierte interne Variablen besitzen, d. h. Daten, die nicht als Ein- bzw. Ausgang an dem FB anliegen. Für die Kommunikation zwischen FBs werden diese Variablen geschrieben und gelesen. Dies ist insbesondere dadurch möglich, dass den anderen FBs die Variablennamen bekannt sind. Insofern sind Methodenaufrufe und Datenaustausch mittels Variablen grundsätzlich gleichwertig ausführbar.

ZITATE ENTHALTEN IN DER BESCHREIBUNG

Diese Liste der vom Anmelder aufgeführten Dokumente wurde automatisiert erzeugt und ist ausschließlich zur besseren Information des Lesers aufgenommen. Die Liste ist nicht Bestandteil der deutschen Patent- bzw. Gebrauchsmusteranmeldung. Das DPMA übernimmt keinerlei Haftung für etwaige Fehler oder Auslassungen.

Zitierte Patentliteratur

- DE 102012025194 [0020]

Patentansprüche

1. Verfahren zum Programmieren einer Speicherprogrammierbaren Steuerung (**40**) zur Steuerung wenigstens eines Energieverbrauchers einer Maschine, wobei zur Programmierung eines Energieverwaltungsprogramms zur Bereitstellung wenigstens einer Energieverwaltungsfunktionalität für die Maschine in einem Programmiergerät (**10**) zur Auswahl durch einen Programmierer wenigstens ein Energieverbraucher-Funktionsbaustein (D) und wenigstens ein Energiesteuerungs-Funktionsbaustein (EC, EC1, EC2) bereitgestellt werden, wobei der wenigstens eine Energieverbraucher-Funktionsbaustein (D) für den wenigstens einen Energieverbraucher spezifische Daten und wobei der wenigstens eine Energiesteuerungs-Funktionsbaustein (EC, EC1, EC2) für die wenigstens eine Energieverwaltungsfunktionalität spezifische Daten enthält.

2. Verfahren nach Anspruch 1, wobei der wenigstens eine Energieverbraucher-Funktionsbaustein (D) programmtechnisch dazu eingerichtet ist, den wenigstens einen Energieverbraucher anzusteuern.

3. Verfahren nach Anspruch 2, wobei der wenigstens eine Energieverbraucher-Funktionsbaustein (D) programmtechnisch dazu eingerichtet ist, den wenigstens einen Energieverbraucher unter Verwendung eines Energieprofils, wie sercos Energy, PRO-Flenergy oder CIPenergy, intern direkt anzusteuern.

4. Verfahren nach einem der vorstehenden Ansprüche, wobei der wenigstens eine Energieverbraucher-Funktionsbaustein (D) programmtechnisch dazu eingerichtet ist, Statusinformationen von dem wenigstens einen Energieverbraucher zu empfangen.

5. Verfahren nach einem der vorstehenden Ansprüche, wobei der wenigstens eine Energieverbraucher-Funktionsbaustein (D) programmtechnisch dazu eingerichtet ist, sich an dem wenigstens einen Energiesteuerungs-Funktionsbaustein (EC, EC1, EC2) anzumelden.

6. Verfahren nach einem der vorstehenden Ansprüche, wobei der wenigstens eine Energiesteuerungs-Funktionsbaustein (EC, EC1, EC2) programmtechnisch dazu eingerichtet ist, den wenigstens einen Energieverbraucher-Funktionsbaustein (D) anzusteuern.

7. Verfahren nach einem der vorstehenden Ansprüche, wobei der wenigstens eine Energiesteuerungs-Funktionsbaustein (EC, EC1, EC2) programmtechnisch dazu eingerichtet ist, den wenigstens einen Energieverbraucher-Funktionsbaustein (D) zu simulieren.

8. Verfahren nach einem der vorstehenden Ansprüche, wobei der wenigstens eine Energiesteuerungs-Funktionsbaustein (EC, EC1, EC2) programmtechnisch dazu eingerichtet ist, eine extern aufrufbare Anmeldemethode bereitzustellen.

9. Verfahren nach einem der vorstehenden Ansprüche, wobei der wenigstens eine Energiesteuerungs-Funktionsbaustein (EC, EC1, EC2) programmtechnisch dazu eingerichtet ist, wenigstens eine andere Instanz des Energiesteuerungs-Funktionsbausteins (EC, EC1, EC2) anzusteuern.

10. Verfahren nach einem der vorstehenden Ansprüche, wobei der wenigstens eine Energiesteuerungs-Funktionsbaustein (EC, EC1, EC2) programmtechnisch dazu eingerichtet ist, sich bei einer anderen Instanz des Energiesteuerungs-Funktionsbausteins (EC, EC1, EC2) anzumelden.

11. Verfahren nach einem der vorstehenden Ansprüche, wobei zur Auswahl durch den Programmierer wenigstens ein Befehls-Funktionsbaustein (C) bereitgestellt wird, der programmtechnisch dazu eingerichtet ist, einen Betriebsbefehl der Energieverwaltungsfunktionalität an den wenigstens einen Energiesteuerungs-Funktionsbaustein (EC, EC1, EC2) und/oder den wenigstens einen Energieverbraucher-Funktionsbaustein (D) auszugeben.

12. Verfahren nach Anspruch 11, wobei der Befehls-Funktionsbaustein (C) programmtechnisch dazu eingerichtet ist, sich an dem wenigstens einen Energiesteuerungs-Funktionsbaustein (EC, EC1, EC2) anzumelden.

13. Verfahren nach Anspruch 11 oder 12, wobei der Betriebsbefehl der Energieverwaltungsfunktionalität eine Energiemodusvorgabe und/oder eine Zeitvorgabe, insbesondere eine Pausenzeit und/oder eine Verfügbarkeitszeit, beinhaltet.

14. Verfahren nach Anspruch 11, 12 oder 13, wobei der wenigstens eine Energiesteuerungs-Funktionsbaustein (EC, EC1, EC2) programmtechnisch dazu eingerichtet ist, den Betriebsbefehl der Energieverwaltungsfunktionalität in einen Betriebsbefehl für den wenigstens einen Energieverbraucher umzuwandeln.

15. Verfahren nach einem der vorstehenden Ansprüche, wobei zur Auswahl durch den Programmierer wenigstens ein Eingabe/Ausgabe-Funktionsbaustein (HMI) bereitgestellt wird, der programmtechnisch dazu eingerichtet ist, energieverwaltungsrelevante Daten zu Visualisierungszwecken zu sammeln und zu gruppieren.

16. Recheneinheit (**10**), die dazu eingerichtet ist, ein Verfahren nach einem der vorstehenden Ansprüche durchzuführen.

17. Computerprogramm mit Programmcodemitteln, die eine Recheneinheit veranlassen, ein Verfahren nach einem der Ansprüche 1 bis 15 durchzuführen, wenn sie auf der Recheneinheit, insbesondere nach Anspruch 16, ausgeführt werden.

18. Maschinenlesbares Speichermedium mit einem darauf gespeicherten Computerprogramm nach Anspruch 17.

Es folgen 3 Seiten Zeichnungen

Anhängende Zeichnungen

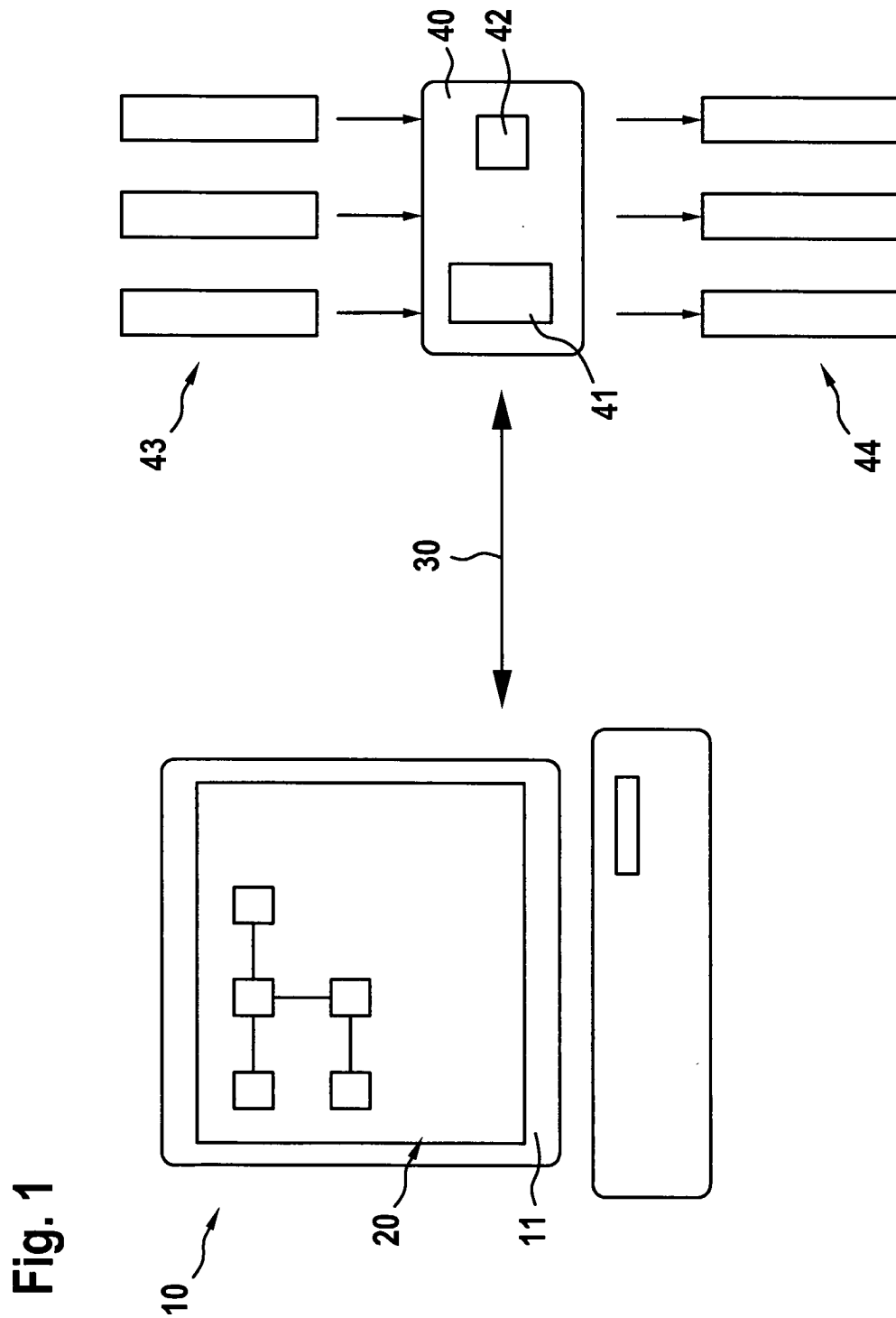


Fig. 2

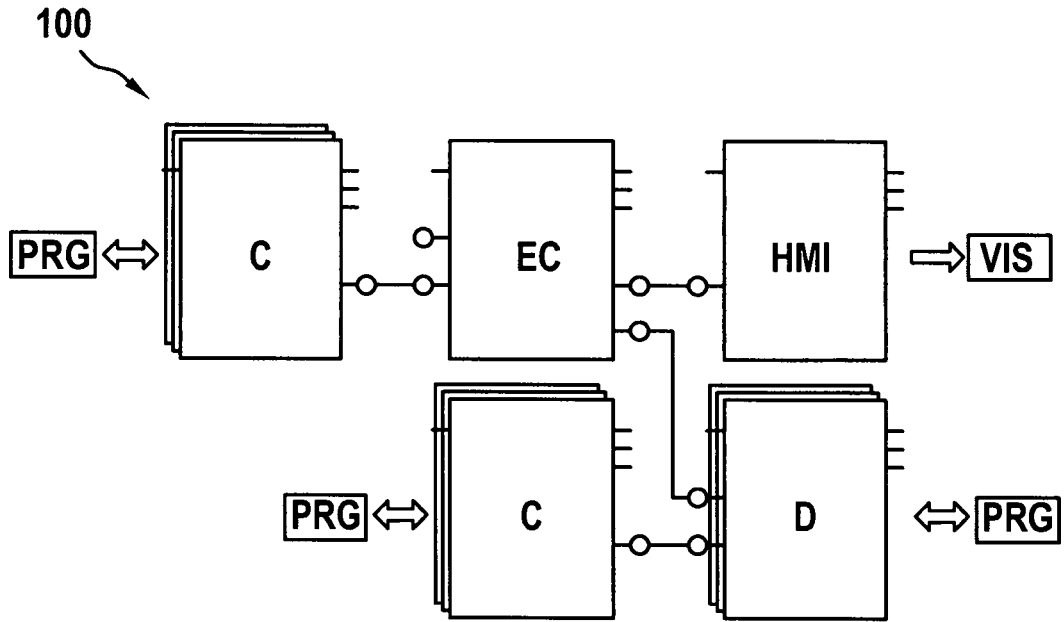


Fig. 3

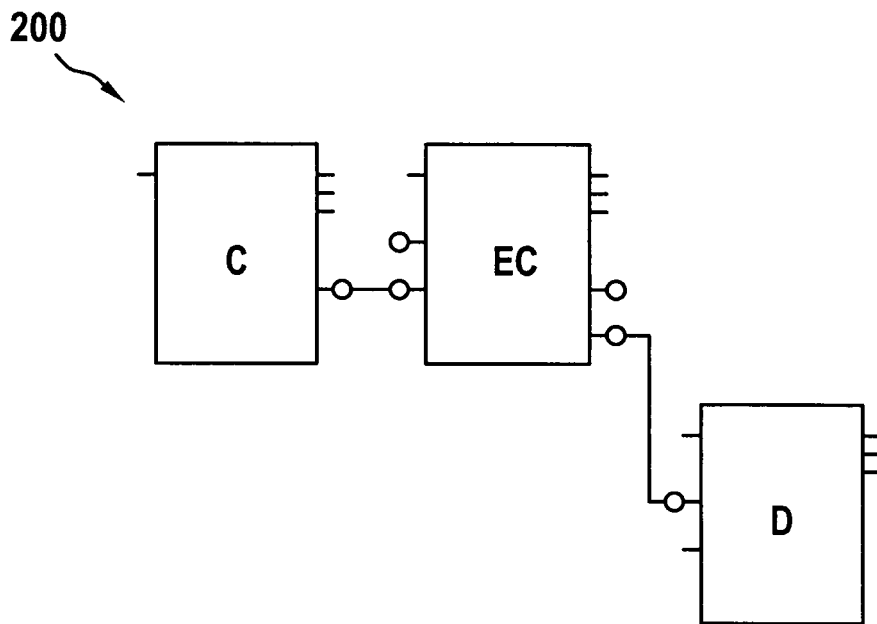


Fig. 4

