



SCHWEIZERISCHE EIDGENOSSENSCHAFT
 BUNDESAMT FÜR GEISTIGES EIGENTUM

① **CH 683952 A5**

⑤ Int. Cl.⁵: **G 05 B 19/417**
H 04 L 12/02
G 06 F 9/06

Erfindungspatent für die Schweiz und Liechtenstein
 Schweizerisch-liechtensteinischer Patentschutzvertrag vom 22. Dezember 1978

⑫ **PATENTSCHRIFT** A5

⑳ Gesuchsnummer: 291/92

㉓ Inhaber:
 Prolink AG, Seengen

㉒ Anmeldungsdatum: 31.01.1992

㉗ Erfinder:
 Weber, Fritz, Wettingen

㉔ Patent erteilt: 15.06.1994

㉚ Vertreter:
 Ernst Ackermann, Patentanwalt, Egg (Flawil)

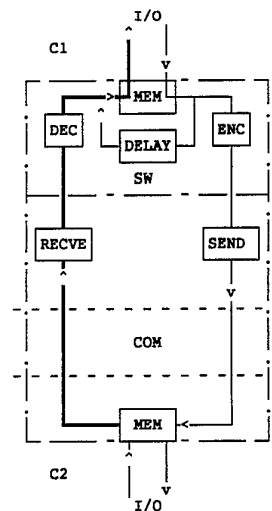
㉕ Patentschrift
 veröffentlicht: 15.06.1994

⑤④ **Verfahren und Vorrichtung zur virtuellen Verkürzung der Uebertragungszeit beim bidirektionalen Datenaustausch zwischen Computer- und/oder Steuersystemen.**

⑤⑦ Zur virtuellen Verkürzung der Übertragungszeit beim Echtzeit-Datenaustausch zwischen Computer- und/oder Steuersystemen wird, ohne die Kommunikation COM zu belasten, für die Dauer der Übertragungszeit vom zu verändernden Echtzeit-Datenbild MEM der Bedienstation C1 eine Vorhersage als Simulierbild hergestellt und im Simulierbild die Veränderung zeitgleich mit der Eingabe vorgenommen, während die Information im Hintergrund dem Zielsystem C2 zugeleitet und umgehend als neues Echtzeit-Datenbild zurückgeführt wird.

Zur schnelleren Reaktion können für die Speicheradressen des Simulierbildes der Bedienstation die entsprechenden Adressen des Echtzeit-Datenbildes verwendet werden. Zusätzlich können Send- und Empfangsinformationen über den gleichen Pfad - und im empfangenden Steuersystem ebenfalls auf identische Speicheradressen geführt werden.

Zur Entlastung der Kommunikation kann im Übertragungspfad ein Multiplexerprogramm eingebaut und die Daten zusätzlich über ein Decoder- DEC und Encoderprogramm ENC zur Daten-Umwandlung bzw. -Komprimierung geführt werden können, wobei ein Mehrfach-Decoder-/Encoderprogramm mit Ein-/Ausgabeschnittstellen I/O für mehrere Ebenen eine weitere Entlastung bringt.



Beschreibung

Die Erfindung betrifft ein Verfahren sowie eine Vorrichtung zur virtuellen Verkürzung der Übertragungszeit beim bidirektionalen Datenaustausch zwischen Computer- und/oder Steuersystemen, wobei ein vorhandenes Echtzeit-Datenbild über eine Eingabeeinrichtung verändert und sichtbar gemacht wird.

Solche Verfahren eignen sich insbesondere im Bereiche der Visualierungs- und Leitsystemtechnik wo Datenbilder mit Echtzeitverhalten dargestellt und vom Bediener beeinflusst werden. Sie sind von grossem praktischem Interesse, denn sie ermöglichen eine effiziente, prozessnahe Bedienung rechnergekoppelter Systeme mit kostengünstigen, marktgängigen Übertragungseinrichtungen bei gleichzeitiger Vereinfachung der notwendigen Übertragungsprozeduren und verbesserter Nutzung bestehender Übertragungskapazitäten.

Es sind bereits Verfahren zur Verkürzung der Übertragungszeit beim Datenaustausch zwischen Computer- und/oder Steuersystemen bekannt, bei denen entweder die Übertragungsgeschwindigkeit erhöht – oder bestimmte Daten bevorzugt behandelt werden (Prioritätssteuerung). Auch sind Verfahren zur schnellen Reaktion am Bildschirm bekannt bei denen der Bediener im wesentlichen auf ein bevorstehendes Warten bis zur Verarbeitung seiner Eingabe aufmerksam gemacht wird (Lokales Echo). Der heutige Trend geht allgemein in Richtung leistungsstärkerer Systeme (Lichtleiter, Multiprozessoren) mit immer kürzeren Verarbeitungs- und damit Reaktionszeiten.

Das Warten bis zur Verarbeitung einer Eingabe ist für den Bediener nicht nur lästig, sondern es verhindert oft ein Echtzeitverhalten und damit eine vernünftige Handhabung eines Prozesses, insbesondere bei einfacheren Computersystemen. Oft ist das Problem gar nicht ein langsamer Datenverkehr sondern die simple Tatsache, dass ein an sich schnelles Netzwerk momentan überlastet ist und das System dem Bediener keine Antwort in der von ihm erwarteten Zeit geben kann. Der vermehrte Einsatz rechnergekoppelter und vernetzter Systeme fordert daher stets höhere Übertragungsleistungen die heute nur mit modernster Technologie (Hardware), mit komplizierten Übertragungsprozeduren (Firmware, Software) oder erheblichem Datenverwaltungsaufwand erreicht werden. Trotz fortgeschrittener Technik ist der Einsatz modernster Technologie in kleineren und mittleren Anlagen oft finanziell nicht tragbar. Prioritätsgesteuerte Übertragungen sind in der Regel herstellergelbunden, meistens nicht offen und wenn nicht bereits auf dem Markt verfügbar, ebenfalls zu teuer. Solche Übertragsarten gehen aber immer zu Lasten der Gesamtübertragungsleistung eines Systems, weil ja die vorgegebene mittlere Übertragungsgeschwindigkeit nicht erhöht wird, die aufwendigere Übertragungsprozedur jedoch zusätzlich verarbeitet werden muss.

Der Erfindung liegt die Aufgabe zugrunde, mit marktgängigen Übertragungssystemen eine Systemreaktion mit Echtzeitverhalten zu erreichen, um dem Bediener eine vernünftige Handhabung eines Prozesses zu ermöglichen, wobei Echtzeit-Datenbilder von Computer- und/oder Steuersystemen verändert werden können und allenfalls bestehende Übertragungsprozeduren verwendbar sind.

Erfindungsgemäss wird dies dadurch erreicht, dass für den Bereich der Eingabeeinrichtung von dem Echtzeit-Datenbild ein Simulierbild hergestellt und im Simulierbild die Veränderung zeitgleich mit der Eingabe vorgenommen wird.

Die Erfindung zielt darauf ab, Echtzeitdaten (Rückmeldungen, Sollwerte), welche auch von anderer Stelle (etwa dem Steuersystem) generiert werden können, normalerweise prozessgetreu im lokalen System abzubilden und nur im Befehlsfalle bevorzugt während der Zeitdauer der Übertragung kurzzeitig zu simulieren. Der Bediener sieht so auf Tastendruck den von ihm befohlenen Zustand am Bildschirm, ohne die Befehls-Übertragungszeit bzw. die wirkliche Antwort des Systems abwarten zu müssen. Die Simulation bzw. Vorhersage des Echtzeit-Datenbildes erzeugt die schnellst mögliche Befehls-Reaktion des Systems mit dem Vorteil, dass die Kommunikation im Befehlsfalle nicht beeinträchtigt werden muss. Damit entfällt eine entsprechende Prioritätssteuerung und die Kommunikationsleistung steht für einen optimierten Gesamtdatendurchsatz voll zur Verfügung. Ein weiterer Vorteil ist ein erhöhter Bedienkomfort aufgrund der Simulation: Das Warten bis zur Verarbeitung einer Eingabe entfällt, der Bediener kann jederzeit nach seinem Ermessen sofort die nächste Aktion ausführen und damit auch schnelle Bedienfolgen erfüllen oder effizienter arbeiten.

Die Erfindung geht in Fällen von Steuersystemen unter anderem auch von der Tatsache aus, dass nach Eingabe von zum Beispiel einer neuen Stellgrösse, im Verarbeitungsprozess Korrekturen vorgenommen werden, die in der Mehrzahl der Fälle weitere Prozessfolgen mit Änderung anderer Daten zur Folge haben. Dies bedeutet nun aber, dass der Nutzen einer bis ins Extreme getriebene Erhöhung der Übertragungsgeschwindigkeit nur relativen Wert hat, da in dem Verarbeitungsprozess Änderungen nur in völlig anderen, bzw. viel grösseren Zeitdimensionen durchführbar sind.

Entscheidend ist die sofortige Quittierung einer Eingabe in dem Simulierbild, mit der Gewissheit, dass die Steueranlage in der je erforderlichen Zeit alle Folgemaassnahmen automatisch korrekt durchführt und gegebenenfalls sogar den Befehl mit einem entsprechenden Hinweis an den Bediener verwirft oder korrigiert.

Dem Bediener wird also für die Zeit der Informationsübertragung das Simulierbild zugänglich gemacht, d.h. für ihn erfolgt die Reaktion dem Verhalten nach (virtuell) in Echtzeit, während die Information im Hintergrund das Zielsystem erreicht und von dort als Echtzeit-Datenbild umgehend wieder erarbeitet und zurück gesendet wird (bidirektional).

Die Echtzeit-Daten für die Bedienstation entsprechen damit einem Abbild von Echtzeit-Daten im Zielsystem welches vorzugsweise als Steuersystem eine Anlage überwacht, steuert und/oder regelt. Die Bedienstation ist über eine konventionelle Einrichtung zur Kommunikation mit dem Steuersystem verbunden, so dass Echtzeit-Daten aus dem Steuersystem bidirektional über einen Empfangskanal gelesen und über einen Sendekanal in das Steuersystem geschrieben werden können. Dabei ist die Art und Weise der Übertragung (Hardware und Software) für das Verfahren unerheblich und deshalb frei wählbar.

Unabhängig von den Eingaben bzw. Simulationen an der Bedienstation kann das Ziel- bzw. Steuersystem das Echtzeit-Datenbild oder Teile davon jederzeit auch beeinflussen und gegebenenfalls dominieren. Nach einer Eingabe an der Bedienstation und Ablauf der Verzögerung wird dort jeweils immer wieder automatisch das Echtzeit-Datenbild aus dem Steuersystem dargestellt. Auf diese Weise lässt sich ein bestimmtes Echtzeit-Datenbild bzw. Teilbild in zwei oder mehreren unabhängigen Computer- und/oder Steuersystemen beeinflussen und dem Verhalten nach in Echtzeit veränderbar machen. Das Verfahren erlaubt jedoch zusätzlich Multiuser Anwendungen und bedarf keines sogenannten Record Locking für das Echtzeit-Datenbild, da stets der zeitlich letzte, zugelassene Eingriff Gültigkeit hat.

Für das Verfahren wird ein Software-Programm für die Bedienstation mit Datenspeicher für Echtzeit-Daten pro bedienbarem Element benutzt. Zentraler Teil des Verfahrens ist ein Programmteil mit vorzugsweise individuellen Verzögerungsgliedern für sämtliche zu bedienenden Elemente, wobei die Verzögerung entsprechend der Übertragungszeit für die Kommunikation angepasst werden kann. Eine Änderung des Datenbildes über die Ein/Ausgabeschnittstelle der Bedienstation wird von diesem Programmteil erfasst, wobei gleichzeitig mit der Übermittlung der Eingabeinformation eine Simulation bzw. Vorhersage des Echtzeit-Datenbildes erfolgt, indem die geänderten Daten als Simulierdaten anstelle der Echtzeit-Daten während der Verzögerung gespeichert werden. Dabei ist entscheidend, dass Änderungen, hervorgerufen durch Echtzeit-Daten aus dem Steuersystem, nicht zu einer Simulation führen, d.h. Empfangs- und Sendepfad werden durch den Programmteil getrennt, so dass keine unerwünschte Rückkopplung stattfinden kann. Die Simulierdaten werden insbesondere für die geänderten Werte solange aufrecht erhalten, bis die Übertragung und Rückmeldung des geänderten Echtzeit-Datenbildes über die Kommunikation vollständig abgeschlossen ist.

Eine schnellere Reaktion kann erreicht werden indem die Speicheradressen für das Simulierbild identisch mit den Speicheradressen für das Echtzeit-Datenbild gewählt werden.

Um den Datenverwaltungsaufwand zu verringern und damit den Verarbeitungsprozess zu beschleunigen, können sowohl Daten von Teilbildern wie auch Daten von mehreren Bildern gemeinsam als Echtzeit-Datenbild bearbeitet werden.

Echtzeit-Daten können auch für nicht sichtbare Bilder im Hintergrund fortwährend aufgefrischt werden, so dass die Daten für einen Zugriff jederzeit mit schnellst möglicher Reaktion als Echtzeit-Datenbild verfügbar sind.

Sende- und Empfangskanäle können mit herkömmlichen, kostengünstigen Kommunikationseinrichtungen über den gleichen Pfad geführt werden, wobei die Speicheradressen der Echtzeit-Daten im empfangenden Ziel- bzw. Steuersystem zur Emulation eines sogenannten Dualport RAM (Random Access Memory) für Send- und Empfangsinformation identisch gewählt werden können.

Im Übertragungspfad kann sendeseitig ein Multiplexerprogramm zur Vielfach-Adressierung eingebaut werden. Die zurückempfangene Information für das Echtzeit-Datenbild muss nicht zwangsweise identisch mit der gesendeten Information aus dem Simulierbild sein, ein Decoder-/Encoderprogramm im Empfangs- bzw. Sendepfad sorgt dafür, dass die Informationen für Echtzeit-Datenbild und Simulierbild kompatibel sind. Gleichzeitig können damit Datentyp-Umwandlungen und/oder Skalierungen bzw. Normierungen vorgenommen werden. Ein Mehrfach-Decoder/Encoder Programm mit Ein/Ausgabeschnittstellen für mehrere Ebenen ermöglicht zudem die direkte Simulation verschiedener Datenformate gleichzeitig. Sowohl Multiplexer wie auch Decoder- bzw. Encoderfunktion bewirken eine Entlastung der Kommunikation weil damit die Daten codiert d.h. in komprimierter Form übertragen werden.

Die Erfindung betrifft eine Vorrichtung und umfasst für den Bereich der Eingabeeinrichtung ein Software-Programm das es ermöglicht dass während der Übertragungszeit von dem Echtzeit-Datenbild ein Simulierbild herstellbar und im Simulierbild die Veränderung zeitgleich mit der Eingabe vornehmbar ist.

Zentraler Teil der Vorrichtung ist dabei ein Programmteil mit vorzugsweise individuellen Verzögerungsgliedern für sämtliche zu bedienenden Elemente, wobei die Verzögerung entsprechend der Übertragungszeit für die Kommunikation angepasst werden kann. Zusätzlich umfasst die Vorrichtung vorzugsweise einen Decoder zur Umwandlung (Decodierung, Skalierung) der empfangenen Echtzeit-Daten und einen Encoder zur Rückwandlung (Codierung, Normierung) der zu sendenden Daten.

Die Veränderung des Echtzeit-Datenbildes durch den Bediener wird vorzugsweise über eine Tastatur oder mit einer sogenannten Pointing Device vorgenommen, wobei das Echtzeit-Datenbild auf einem Bildschirm dargestellt werden kann. Dabei können Echtzeit-Datenbilder für den Bereich der Haustechnik oder für Industrieanlagen mit Echtzeit-Daten von/für Maschinen, Anlagen, Motoren und Ventilen sowie deren Mess-, Steuer- und Regeleinrichtungen dargestellt werden.

Entgegen den heute allgemein üblichen Anstrengungen liegt die Lösung mit dem beschriebenen Verfahren bzw. mit der dazu dienenden Vorrichtung also keineswegs ausschliesslich in schnelleren oder generell leistungsfähigeren Systemen. Vielmehr wird dem Bediener die zu erwartende Antwort sofort lo-

kal mitgeteilt (Aktion=Reaktion) und gegebenenfalls später korrigiert (Management by Exception), sollte sie nicht zutreffen bzw. erfüllt werden können: Ist ein erlaubter Befehl nach Ablauf der zulässigen Reaktionszeit nämlich nicht erfüllt, so ist es gemäss heutiger Technik sinnvoller, den Bediener mit einer entsprechenden Fehlermeldung zu informieren als etwa einen Folgebefehl zuzulassen. Letzteres veranlasst den Bediener nur zum wiederholten sinnlosen Tastendrücken. Unerlaubte Eingriffe können andererseits problemlos vor der Befehlsübermittlung abgefangen und mit einer entsprechenden Fehlermeldung begleitet werden.

In der Folge wird nun die erfindungsgemässe Lösung mit weiteren Einzelheiten anhand mehrerer Ausführungsbeispiele erläutert, es zeigen:

– Fig. 1 das der Erfindung zugrunde liegende Prinzipschaltbild mit den wichtigsten Programmelementen in Bedienstation und Steuersystem sowie den Informations-Datenfluss (Doppellinie) wenn keine Eingabe stattfindet.

– Fig. 2 zeigt Fig. 1 während einer Eingabe und deren Übertragung.

– Fig. 3 als Vergleich zu Fig. 1, die prinzipielle Darstellung eines Dualport RAM (Random Access Memory).

– Fig. 4 den Datenfluss mit zeitlichem Ablauf a) → b) → c) bei einer vom Steuersystem akzeptierten Eingabe zur Steuerung eines Motors.

– Fig. 5 zeigt Fig. 4 bei einer vom Steuersystem nicht akzeptierten Eingabe zur Steuerung eines Motors.

– Fig. 6 analog Fig. 4 den Datenfluss mit zeitlichem Ablauf a) → b) → c) → d) bei einer vom Steuersystem akzeptierten Eingabe eines Sollwertes und eine spätere, automatische Veränderung des Sollwertes durch das Steuersystem.

– Fig. 7 eine Mehrfachanwendung der Erfindung im Netzverbund mit den wichtigsten Programmelementen der einzelnen Stationen.

– Fig. 8 eine Anwendung der Erfindung für sogenanntes Multiplexed Writing mit den wichtigsten Programmelementen für Bedienstation und Steuersystem.

– Fig. 9 eine Anwendung der Erfindung für Testzwecke bei unterbrochenem Datenfluss.

– Fig. 10 eine Anwendung der Erfindung als sogenannter interner Encoder/Decoder bei kurzgeschlossenem Datenfluss.

– Fig. 11 eine Anwendung der Erfindung mit Datenfluss für ein Mehrfach-Decoder/Encoder Programm.

– Fig. 12 eine Anwendung der Erfindung nach Fig. 11 mit Datenfluss für die direkte Decodierung/Encodierung eines Nibbles zur Statusanzeige von Anlagen am Bildschirm.

Nachfolgend sei nun jeweils auf die entsprechenden Figuren verwiesen:

Fig. 1 und Fig. 2 zeigen das der Erfindung zugrunde liegende Prinzipschaltbild mit den wichtigsten Funktionen. In Fig. 1 ist der Informations-Datenfluss (Doppellinie) für das Echtzeit-Datenbild dargestellt wenn keine Eingabe an der Bedienstation C1 stattfindet, Fig. 2 zeigt den entsprechenden Datenfluss während einer Eingabe an der Bedienstation C1. Dem Verfahren liegt ein Software-Programm SW (oberes strichpunktiertes Rechteck) für die Bedienstation C1 mit Datenspeicher für Echtzeit-Daten MEM pro bedienbarem Element zugrunde. Die Echtzeit-Daten MEM für die Bedienstation C1 entsprechen einem Abbild von Echtzeit-Daten MEM im Steuersystem C2. Bedienstation C1 ist über eine konventionelle Einrichtung zur Kommunikation COM (gestrichelte Trennlinien) mit dem Steuersystem C2 verbunden, so dass Echtzeit-Daten MEM aus dem Steuersystem C2 bidirektional über einen Empfangskanal RECVE gelesen und über einen Sendekanal SEND in das Steuersystem C2 geschrieben werden können. Dabei ist die Art und Weise der Übertragung (Hardware und Software) für das Verfahren unerheblich und deshalb frei wählbar.

Das Software-Programm SW umfasst vorzugsweise einen Decoder DEC zur Umwandlung (Decodierung, Skalierung) der empfangenen Echtzeit-Daten MEM und einen Encoder ENC zur Rückwandlung (Codierung, Normierung) der zu sendenden Daten.

Zentraler Teil des Verfahrens ist ein Programmteil mit vorzugsweise individuellen Verzögerungsgliedern für sämtliche zu bedienenden Elemente, wobei die Verzögerung DELAY entsprechend der Übertragungszeit für die Kommunikation COM angepasst werden kann. Eine Änderung des Datenbildes über die Ein/Ausgabeschnittstelle I/O der Bedienstation C1 wird von diesem Programmteil erfasst, wobei gleichzeitig mit der Übermittlung der Eingabeinformation eine Simulation bzw. Vorhersage des Echtzeit-Datenbildes erfolgt, indem die geänderten Daten als Simulierdaten SIM anstelle der Echtzeit-Daten MEM während der Verzögerung DELAY gespeichert werden. Dabei ist entscheidend, dass Änderungen, hervorgerufen durch Echtzeit-Daten MEM aus dem Steuersystem C2, nicht zu einer Simulation führen, d.h. Empfangs- und Sendepfad werden durch den Programmteil getrennt, so dass keine unerwünschte Rückkopplung stattfinden kann. Die Simulierdaten SIM werden solange aufrecht erhalten, bis die Übertragung und Rückmeldung des geänderten Echtzeit-Datenbildes über die Kommunikation COM vollständig abgeschlossen ist.

Unabhängig von den Eingaben bzw. Simulationen an der Bedienstation C1 kann das Steuersystem C2 das Echtzeit-Datenbild oder Teile davon jederzeit auch beeinflussen und gegebenenfalls dominieren.

Nach einer Eingabe an der Bedienstation C1 und Ablauf der Verzögerung DELAY wird dort jeweils immer wieder automatisch das Echtzeit-Datenbild aus dem Steuersystem C2 dargestellt. Auf diese Weise lässt sich ein bestimmtes Echtzeit-Datenbild bzw. Teilbild in zwei oder mehreren unabhängigen Computer- und/oder Steuersystemen beeinflussen und dem Verhalten nach in Echtzeit veränderbar machen.

Das gesamte System mit Ein/Ausgabeschnittstelle I/O, Echtzeit-Daten MEM, Bedienstation C1, Kommunikation COM und Steuersystem C2 (strichpunktiertes grosses Rechteck) kann gemäss Fig. 3 mit einem Dualport RAM verglichen werden. Das Verfahren erlaubt jedoch zusätzlich Multiuser Anwendungen wie für Fig. 7 beschrieben.

Fig. 3 zeigt zum Vergleich mit Fig. 1 und Fig. 2 die prinzipielle Darstellung eines Dualport RAM (Random Access Memory). Beim Dualport RAM handelt es sich um einen Speicher-Baustein auf den mit zwei unabhängigen Bussystemen zugegriffen werden kann.

Fig. 4 und Fig. 5 zeigen ein vereinfachtes Beispiel mit zeitlichem Ablauf a) → b) → c) für den Datenfluss (Doppellinie) zur Steuerung und Visualisierung eines Motors. Dabei wird ein Motor für die Zustände ON und OFF auf dem Bildschirm der Bedienstation C1 dargestellt. Der Motor kann sowohl von der Bedienstation C1 wie auch vom Steuersystem C2 jederzeit ein- oder ausgeschaltet werden, wobei der Bediener immer (in Echtzeit bzw. mit der Verzögerungszeit der Übertragung) über den aktuellen Zustand des Motors informiert ist. In der Regel dominiert das Steuersystem C2 indem es durch Verriegelungsschaltungen die Befehle von der Bedienstation C1 zulässt oder nicht erlaubt.

a) Angenommen, der Motor sei ausgeschaltet und eine Bedienung sei zur Zeit nicht erfolgt, so ist die Ein/Ausgabeschnittstelle I/O über das für Fig. 1 beschriebene Software-Programm auf Empfang geschaltet und zeigt den zurückgemeldeten Zustand OFF des Motors an.

b) Im Falle einer Eingabe an Ein/Ausgabeschnittstelle I/O wird automatisch auf Senden umgeschaltet und die Anzeige entspricht während der Übertragungszeit bzw. der Verzögerung DELAY dem eingegebenen Befehl. Wünscht der Bediener also, den Motor einzuschalten, so wird der Einbefehl bei gleichzeitiger Anzeige am Bildschirm (Simulierbild) über das Software-Programm dem Motor zugeleitet. Der Bediener muss nun nicht auf die Rückmeldung aus Steuersystem C2 warten, er erlebt quasi eine augenblickliche, virtuell verkürzte Reaktion.

c) Nach Ablauf der Verzögerung DELAY erstellt das Programm automatisch wieder die Grundstellung – die Anzeige ist wieder «echt». Abhängig von der Verriegelungsschaltung für den Motor im Steuersystem C2 ergeben sich nun zwei Möglichkeiten:

In Fig. 4 wird der Einbefehl vom Steuersystem C2 akzeptiert und als echter Zustand nach Ablauf der Verzögerung DELAY angezeigt – die Anzeige ON bleibt erhalten.

In Fig. 5 wird der Einbefehl vom Steuersystem C2 nicht akzeptiert. Nach Ablauf der Verzögerung DELAY wird automatisch wieder der alte Zustand angezeigt – das System «korrigiert» quasi den nicht erlaubten Eingriff automatisch nach Ablauf der Übertragungszeit und die Anzeige ON wechselt zurück auf OFF.

Fig. 6 zeigt analog zu Fig. 4 den Datenfluss (Doppellinie) mit zeitlichem Ablauf a) → b) → c) → d) bei einer vom Steuersystem C2 akzeptierten Änderung eines Sollwertes von 156 auf 330 durch den Bediener, sowie

d) eine spätere Erhöhung des Sollwertes von 330 auf 712 durch das Steuersystem C2 selbst, z.B. für einen Regler mit automatischer Sollwertführung.

Auf dieselbe Weise werden auch mehrstufige Eingaben z.B. für Startprozeduren mit Anfahrsequenz-Rückmeldungen (siehe Fig. 12), Stellgrössen oder Grenzwerte behandelt: Der Bediener sieht immer den aktuellen Anlagenzustand und auf Tastendruck sofort den von ihm befohlenen Zustand ohne die Übertragungs-Reaktionszeit abzuwarten.

Fig. 7 zeigt eine Mehrfachanwendung der Erfindung im Netzverbund mit den wichtigsten Programm-elementen in den einzelnen Stationen. Die Bedienstation C1 ist für das Beispiel mit drei weiteren Bedienstationen C3, C4 und C5 über Empfangskanäle RECVE und Sendekanäle SEND mit einem Local Area Network LAN verbunden, wobei die Bedienstation C1 wie für Fig. 1 beschrieben mit dem Steuersystem C2 gekoppelt ist. Selbstverständlich sind auch andere Konfigurationen möglich, beispielsweise könnte das Steuersystem C2 auch direkt mit dem Local Area Network LAN gekoppelt werden oder es existiert gar kein Steuersystem C2 für einen Echtzeit-Datenaustausch in einem Netzwerksystem.

Alle vier Bedienstationen C1, C3, C4 und C5 erlauben, wie für Fig. 1 beschrieben, eine Beeinflussung des Echtzeit-Datenbildes mit entsprechender lokaler Simulation an der Ein/Ausgabeschnittstelle I/O der jeweils gerade bedienenden Station. Dabei können Datenbilder an allen Bedienstationen gleichzeitig verändert werden, die zeitlich letzte Eingabe dominiert. Die nicht bedienenden Stationen empfangen das neue Echtzeit-Datenbild jeweils ohne Simulation automatisch nach der normalen Übertragung. Dabei ist es auch möglich, dass ein Teilbild als Echtzeit-Datenbild gerade empfangen und dargestellt wird, während ein anderer Teil desselben Bildes als Simulierbild erscheint. Unabhängig von den Eingaben bzw. Simulationen an den Bedienstationen kann das Steuersystem C2 das Echtzeit-Datenbild jederzeit auch beeinflussen und gegebenenfalls absolut dominieren.

Auf diese Weise lässt sich ein Echtzeit-Datenbild bzw. Teilbild in mehreren unabhängigen Computer- und/oder Steuersystemen beeinflussen und dem Verhalten nach in Echtzeit verändern. Das Verfahren

unterstützt daher Multiuser Anwendungen und bedarf keines sogenannten Record Locking für das Echtzeit-Datenbild, da stets der zeitlich letzte, zugelassene Eingriff Gültigkeit hat.

Fig. 8 zeigt eine Anwendung der Erfindung für sogenanntes Multiplexed Writing mit den wichtigsten Programmelementen für Bedienstation C1 und Steuersystem C2. Dabei wird der Encoder ENC in Fig. 1 zu einem Multiplexer MUX ausgebaut. Der Multiplexer nützt einen Sendekanal oder Speicherplatz mehrfach aus und entlastet das System indem sämtliche zu sendenden Daten nur gerade während einer Bedienung generiert und übertragen werden, d.h. wenn sie auch wirklich anfallen. Der Zugriff auf die Echtzeit-Daten MEM im Steuersystem C2 wird nicht mehr vom Kommunikationsprotokoll gesteuert, sondern erfolgt dann mit einem Demultiplexer DMX im Steuersystem C2 selbst.

Fig. 9 zeigt eine Anwendung der Erfindung für Testzwecke bei unterbrochenem Datenfluss. Weil das System einen Eingriff des Bedieners an der Ein/Ausgabeschnittstelle I/O automatisch nach Ablauf der Verzögerung DELAY «korrigiert», ergibt sich bei unterbrochenem Datenfluss ein Verhalten wie für Fig. 5 beschrieben. Eine unterbrochene oder willkürlich nicht in Betrieb gesetzte Kommunikation COM eignet sich daher zum Austesten von Applikationen und Kommunikationsdaten ohne Steuersystem C2 indem jeweils Aktionen an der Ein/Ausgabeschnittstelle I/O eingegeben und kurzzeitig simuliert werden.

Fig. 10 zeigt eine Anwendung der Erfindung mit intern kurzgeschlossenem Datenfluss als sogenannter Encoder/Decoder. Bei direkter Rückführung LINK des Datenflusses aus dem Encoder ENC zum Decoder DEC mit entsprechendem Zugriff auf die codierten Daten, erhält man quasi ein Dualport RAM gemäss Fig. 3 mit internem Datenumwandler, wobei Daten sowohl vorwärts wie auch rückwärts gewandelt werden können z.B. Skalieren ↔ Normieren oder Decodieren ↔ Codieren. Die nachfolgende Code-Tabelle Tab. 1 zeigt Darstellungen für eine mögliche Datenumwandlung.

Fig. 11 zeigt eine Anwendung der Erfindung mit Datenfluss für ein Mehrfach-Decoder/Encoder Programm. In diesem Fall wird für die Bedienstation C1 ein 16-Bit Wort in Bits 00..15, Nibbles N0..N4, Bytes B0..B1 und wieder in ein Wort W0 «zerlegt» und zur Eingabe bzw. Anzeige an der Ein/Ausgabeschnittstelle I/O gebracht, wobei sämtliche Elemente als Echtzeit-Daten MEN gemäss Beschreibung für Fig. 1 verwendet werden können. Ein Steuersystem C2 hat in der Regel direkten Zugriff auf einzelne Worte, Bytes und Bits. Damit können z.B. 16 Bit-Zustände oder etwa zwei Messwerte als Bytes mit einem einzigen Wort übermittelt und beidseitig beeinflusst werden. Die nachfolgende Code-Tabelle Tab. 1 zeigt weitere Darstellungen für eine mögliche Datentyp-Umwandlung.

Tab.1 Code-Tabelle für Datentyp-Umwandlung:

5	Codierte Form <-ENC/DEC->	Decodierte Form																																

10	<p>Datenwort enthaltend sechzehn einzelne Bit-Zustände 15 bis 00</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td style="width: 15px; height: 15px;"></td><td style="width: 15px; height: 15px;"></td><td style="width: 15px; height: 15px;"></td><td style="width: 15px; height: 15px;"></td><td style="width: 15px; height: 15px;"></td><td style="width: 15px; height: 15px;"></td><td style="width: 15px; height: 15px;"></td><td style="width: 15px; height: 15px;"></td><td style="width: 15px; height: 15px;"></td><td style="width: 15px; height: 15px;"></td><td style="width: 15px; height: 15px;"></td><td style="width: 15px; height: 15px;"></td><td style="width: 15px; height: 15px;"></td><td style="width: 15px; height: 15px;"></td><td style="width: 15px; height: 15px;"></td><td style="width: 15px; height: 15px;"></td></tr> <tr><td colspan="16">16 Bits</td></tr> </table> </div> <p>Bit 15.....0</p>																	16 Bits																<p>16 einzeln ansprechbare Digitalelemente Bit</p> <p>Code 00...15</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <p>Log"1" oder "0"</p> </div>
16 Bits																																		
20	<p>Datenwort enthaltend vier Nibbles N3 N2 N1 N0</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td style="width: 20px; height: 20px;"></td><td style="width: 20px; height: 20px;"></td><td style="width: 20px; height: 20px;"></td><td style="width: 20px; height: 20px;"></td></tr> <tr><td colspan="4">4 Nibbles</td></tr> </table> </div> <p>Bit ..12 ..8 ..4 ..0</p>					4 Nibbles				<p>4 einzeln ansprechbare Analogelemente Nibble</p> <p>Code N0...N3</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <p>0...15</p> </div>																								
4 Nibbles																																		
30	<p>Datenwort enthaltend zwei Bytes (Signed oder Unsigned)</p> <p>BH/B1 BL/B0</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td style="width: 40px; height: 20px;">±HiByte</td><td style="width: 40px; height: 20px;">±LoByte</td></tr> </table> </div> <p>Bit 15.....8 7.....0</p>	±HiByte	±LoByte	<p>2 einzeln ansprechbare Analogelemente Byte Signed</p> <p>Code B0 od. B1</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <p>-128...+127</p> </div> <p>Code BL od. BH</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <p>oder Unsigned</p> </div>																														
±HiByte	±LoByte																																	
40	<p>Datendoppelwort enthaltend ein 32-Bit Wert (Signed Long Integer)</p> <p>LH LL</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td style="width: 60px; height: 20px;">± High Value</td><td style="width: 60px; height: 20px;">Low Value</td></tr> </table> </div> <p>31....Bit.....16 15....Bit.....0</p>	± High Value	Low Value	<p>Einzeln ansprechbares Analogelement Long</p> <p>Code LL+LH</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <p>± 2.14*10⁹</p> </div>																														
± High Value	Low Value																																	
50	<p>Datendoppelwort enthaltend eine 32-Bit Gleitpunktzahl</p> <p>FH FL</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td style="width: 40px; height: 20px;">± Exp.</td><td style="width: 80px; height: 20px;">± Mantissa</td></tr> </table> </div> <p>31....24 23.....Bit.....0</p>	± Exp.	± Mantissa	<p>Einzeln ansprechbares Analogelement Float</p> <p>Code FL+FH</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <p>± 3.4*10³⁸</p> </div>																														
± Exp.	± Mantissa																																	
60	<p>Mehrfachwort enthaltend Text-Zeichen als Zahl z.B. -0.37096E5 TX</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <p>Character String</p> </div>	<p>Einzeln ansprechbares Analogelement Double Float</p> <p>Code TX</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <p>± 1.7*10³⁰⁸</p> </div>																																

Fig. 12 zeigt eine Anwendung der Erfindung gemäss Beschreibung für Fig. 11 mit Datenfluss für die direkte Decodierung/Encodierung eines Nibbles (4-Bit Analogwert) zur Statusanzeige von Anlagen am Bildschirm. Dabei wird ein Anlagenzustand STATUS in Form eines Nibbles N0 gemäss nachfolgender Wahrheitstabelle Tab. 2 zur Anzeige gebracht.

5

Tab. 2 Wahrheitstabelle für Nibble:

	FAULT Bit 03	RUN Bit 02	AUTO Bit 01	ON/OFF Bit 00	STATUS N0	Anlagenzustand, Bildschirmanzeige
10	0	0	0	0	0	Steht
	0	0	0	1	1	Start
	0	0	1	0	2	Steht Automatik
15	0	0	1	1	3	Start Automatik
	0	1	0	0	4	Stop
	0	1	0	1	5	Läuft
	0	1	1	0	6	Stop Automatik
20	0	1	1	1	7	Läuft Automatik
	1	0	0	0	8	Fehler
	1	0	0	1	9	Fehler Start
25	1	0	1	0	10	Fehler Automatik
	1	0	1	1	11	Fehler Automatik Start

30 Der Anlagenzustand STATUS kann z.B. mit verschiedenen Farben oder Symbolen auf dem Bildschirm animiert werden und setzt sich aus vier Bit-Informationen 00..03 zusammen, wovon Bit 00 gleichzeitig den Ein/Ausbefehl ON/OFF darstellt und Bit 03 als Fehlermeldung weiter verarbeitet werden kann. Der Ein/Ausbefehl ON/OFF kann gegebenenfalls mit der AUTO Freigabe Bit 01 in der Bedienstation C1 verriegelt werden, d.h. eine Bedienung über die Ein/Ausgabeschnittstelle I/O wird nur akzeptiert wenn das Steuersystem C2 die entsprechende AUTO Freigabe vorgängig erteilt hat.

35 Der Ein/Ausbefehl ON/OFF wird bei erlaubter Eingabe an der Ein/Ausgabeschnittstelle I/O sowohl dem Sendekanal SEND wie auch dem entsprechenden Bit 00 und damit zur sofortigen STATUS Anzeige dem Nibble N0 zugeführt. Der Bediener sieht also auf Tastendruck z.B. den simulierten STATUS «Start Automatik» der, sofern das Steuersystem C2 den Befehl zulässt, nach der Übertragungszeit zum Echtzeit-Datenbild wird und abhängig von weiteren Verlauf des Bearbeitungsprozesses später auf «Läuft Automatik» oder etwa «Fehler Automatik Start» wechselt (vergleiche auch Beschreibung für Fig. 6).

40

Patentansprüche

45 1. Verfahren zur virtuellen Verkürzung der Übertragungszeit beim bidirektionalen Datenaustausch zwischen Computer- und/oder Steuersystemen, wobei ein vorhandenes Echtzeit-Datenbild über eine Eingabeeinrichtung verändert und sichtbar gemacht wird, dadurch gekennzeichnet, dass für den Bereich der Eingabeeinrichtung von dem Echtzeit-Datenbild ein Simulierbild hergestellt und im Simulierbild die Veränderung zeitgleich mit der Eingabe vorgenommen wird.

50 2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass das Simulierbild während der Übertragungszeit aufrechterhalten wird.

3. Verfahren nach Anspruch 1 oder 2, dadurch gekennzeichnet, dass die Speicheradressen für das Simulierbild identisch sind mit den Speicheradressen für das Echtzeit-Datenbild.

55 4. Verfahren nach Anspruch 1 oder 2, dadurch gekennzeichnet, dass das Echtzeit-Datenbild aus Daten von Teilbildern und/oder mehreren Bildern besteht, wobei vorzugsweise nur die durch die Eingabeeinrichtung veränderten Daten im Simulierbild bis zur Rückmeldung der tatsächlichen Werte fixiert sind, während alle übrigen Daten als Echtzeitdaten normal fortwährend aufgefrischt werden.

5. Verfahren nach Anspruch 4, dadurch gekennzeichnet, dass das Echtzeit-Datenbild auch für nicht sichtbare Bilder fortwährend aufgefrischt wird.

60 6. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass die Sende- und Empfangskanäle über den gleichen Übertragungspfad geführt werden.

7. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass die Speicheradressen der Echtzeit-Daten im empfangenden Steuersystem für Sende- und Empfangsinformation identisch sind.

65 8. Verfahren nach Anspruch 6, dadurch gekennzeichnet, dass im Übertragungspfad sendeseitig ein Multiplexerprogramm zur Vielfach-Adressierung eingebaut wird.

9. Verfahren nach Anspruch 6, dadurch gekennzeichnet, dass im Übertragungspfad empfangsseitig ein Decoder- bzw. sendeseitig ein Encoderprogramm zur Datentyp-Umwandlung bzw. Skalierung oder ein Mehrfach-Decoder/Encoder Programm mit Ein/Ausgabeschnittstellen für mehrere Ebenen eingebaut wird.

5 10. Vorrichtung zur Durchführung des Verfahrens nach Anspruch 1, dadurch gekennzeichnet, dass für den Bereich der Eingabeeinrichtung während der Übertragungszeit von dem Echtzeit-Datenbild mittels eines Software-Programmes ein Simulierbild herstellbar und im Simulierbild die Veränderung zeitgleich mit der Eingabe vornehmbar ist.

10 11. Vorrichtung nach Anspruch 10, dadurch gekennzeichnet, dass die Eingabeeinrichtung eine Tastatur und/oder eine sogenannte Pointing-Device zur Veränderung des Echtzeit-Datenbildes aufweist und für die Darstellung des Echtzeit Datenbildes ein Bildschirm verwendbar ist.

12. Vorrichtung nach Anspruch 10 oder 11, dadurch gekennzeichnet, dass ein Echtzeit-Datenbild für den Bereich der Haustechnik oder für eine Industrieanlage mit Echtzeit-Daten von bzw. für Maschinen, Anlagen, Motoren und Ventilen sowie deren Mess-, Steuer- und Regeleinrichtungen darstellbar ist.

15

20

25

30

35

40

45

50

55

60

65

Fig.1

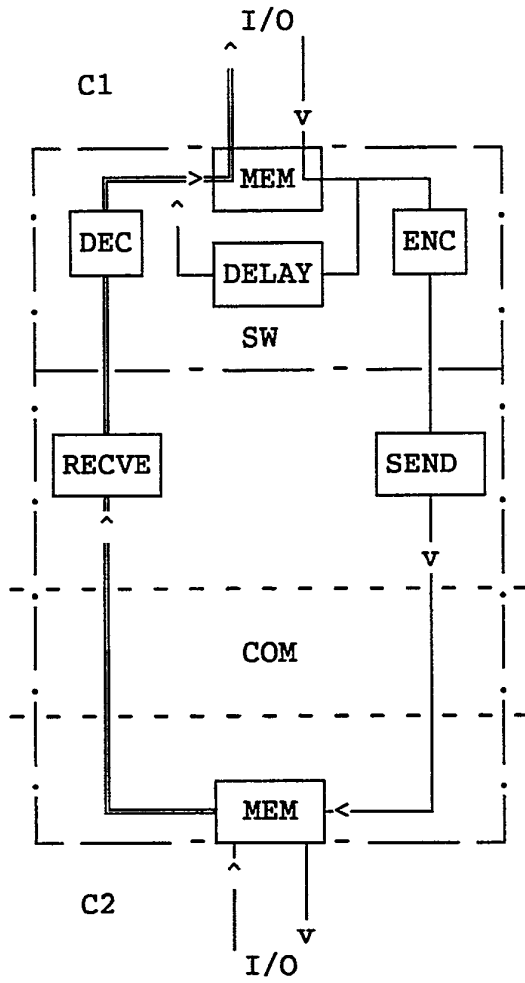


Fig.2

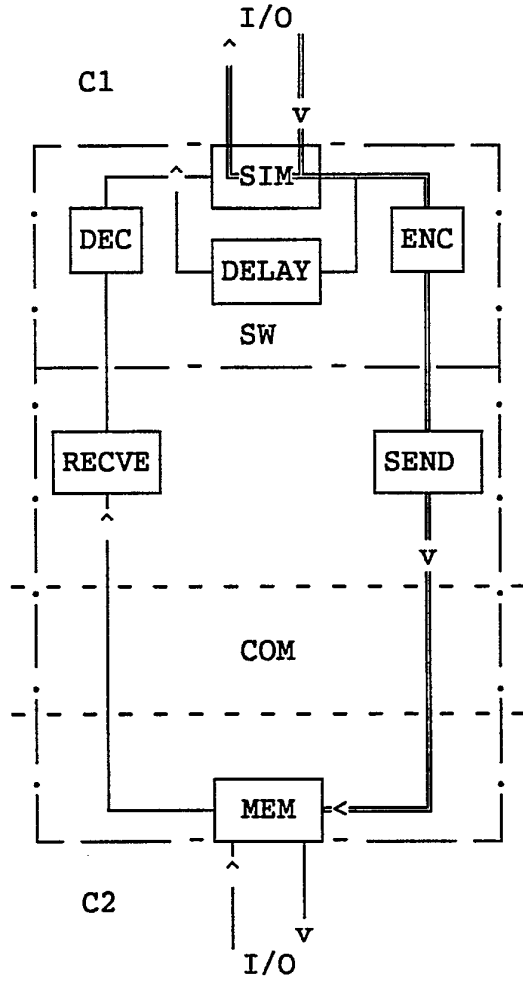


Fig.3

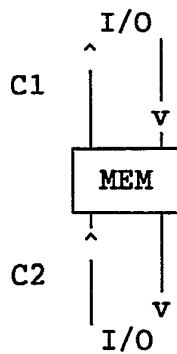


Fig. 4

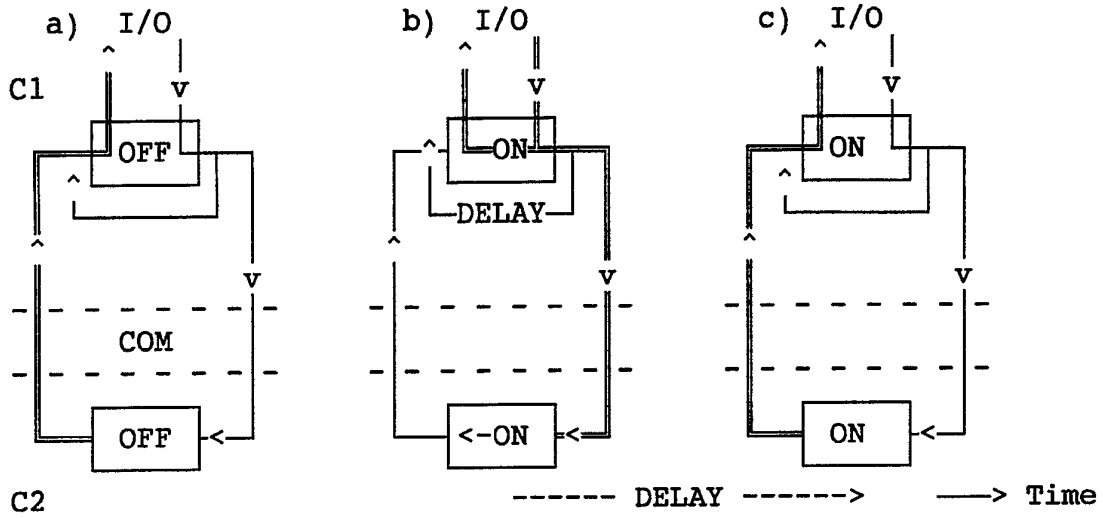


Fig. 5

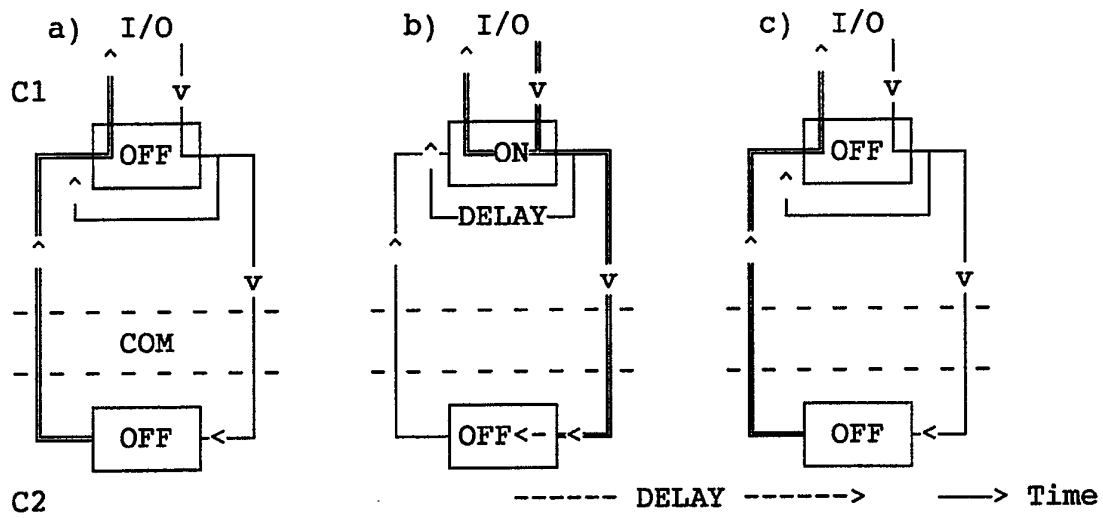


Fig. 6

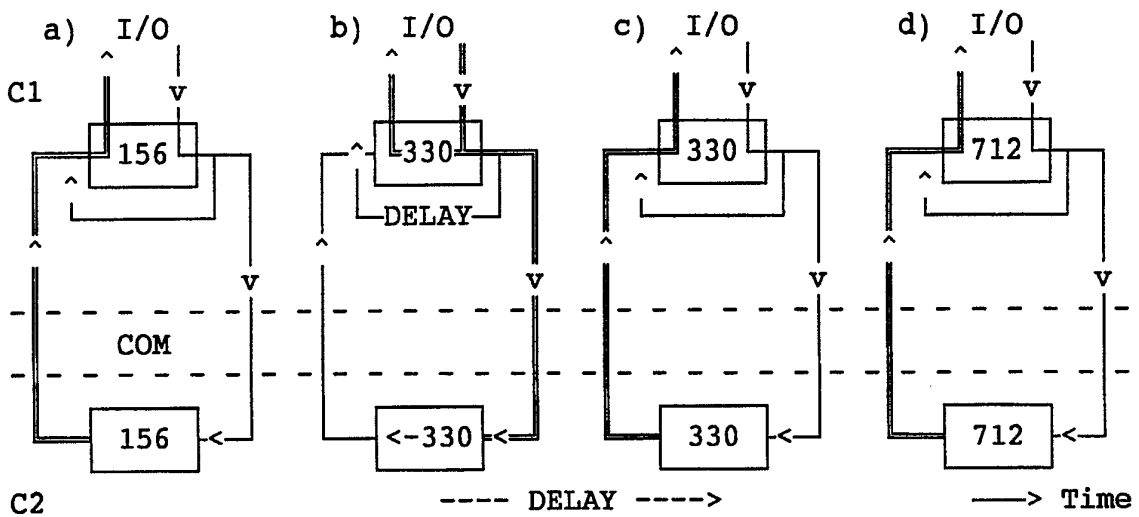


Fig.7

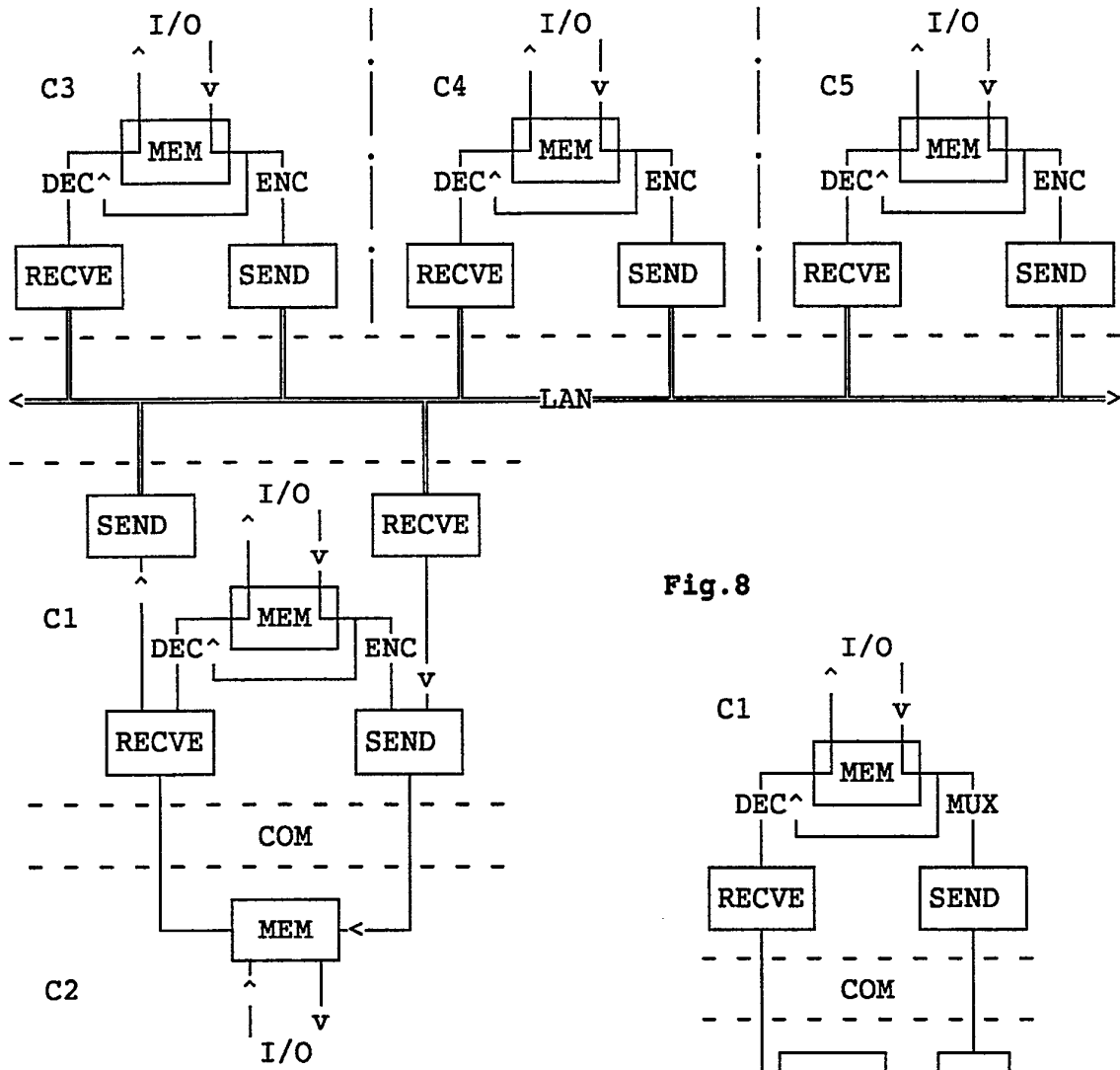


Fig.8

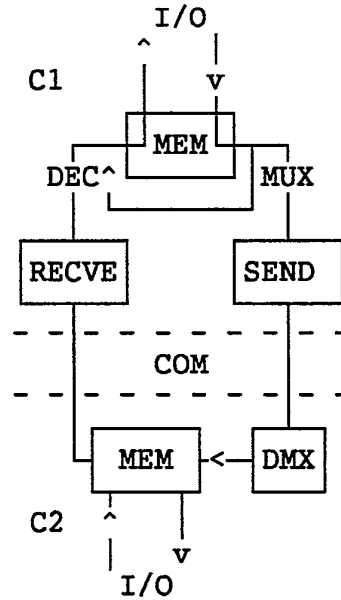


Fig.9

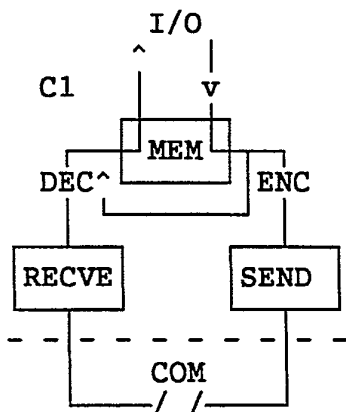


Fig.10

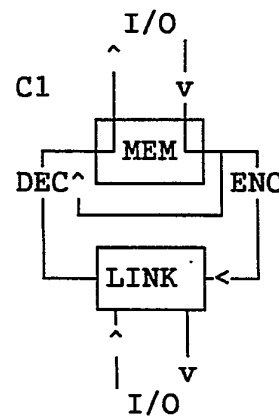


Fig.11

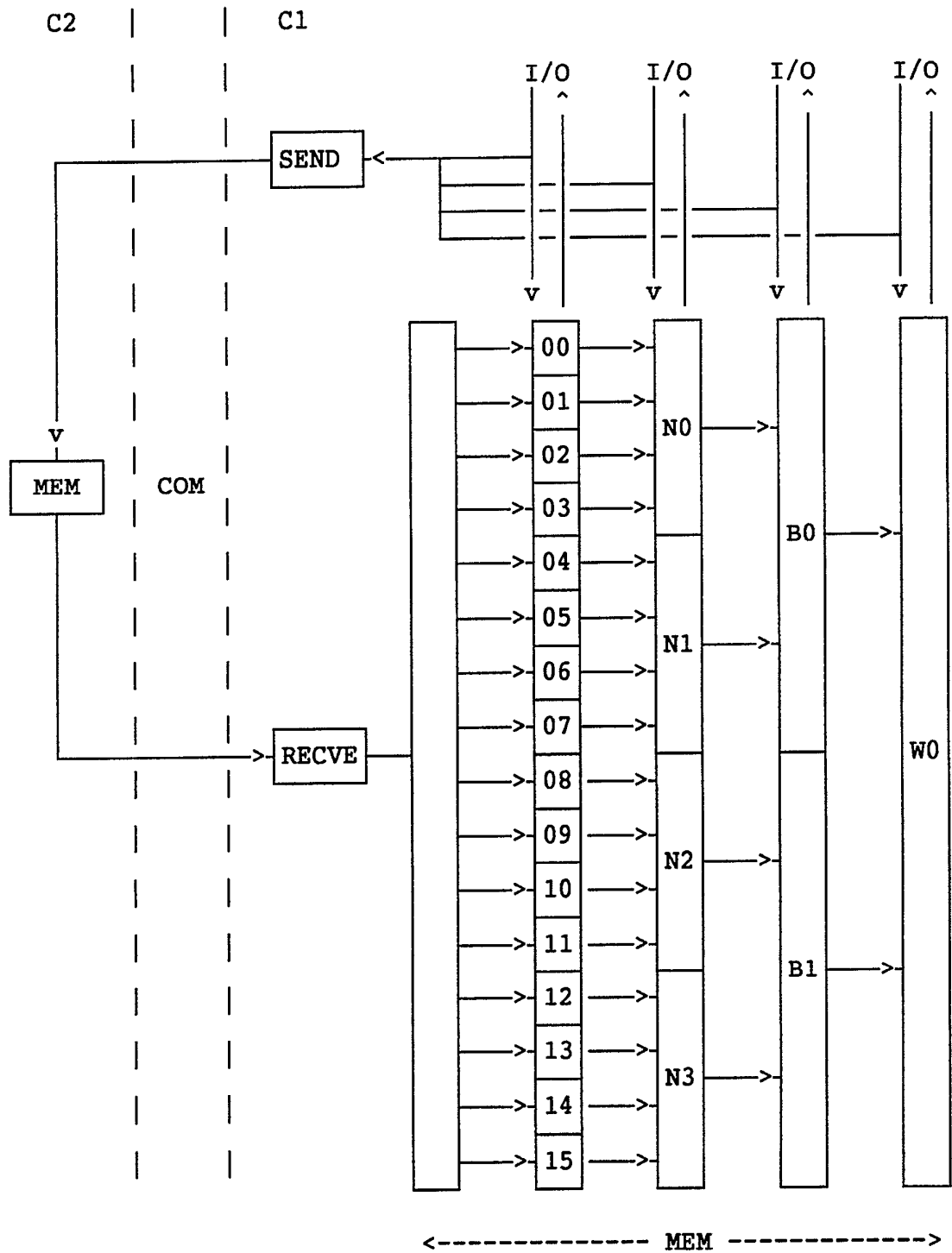


Fig.12

