



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2011년02월25일
(11) 등록번호 10-1017444
(24) 등록일자 2011년02월17일

(51) Int. Cl.

G06F 11/16 (2006.01) G06F 15/78 (2006.01)

(21) 출원번호 10-2007-7008952

(22) 출원일자(국제출원일자) 2005년10월25일

심사청구일자 2008년12월12일

(85) 번역문제출일자 2007년04월20일

(65) 공개번호 10-2007-0062566

(43) 공개일자 2007년06월15일

(86) 국제출원번호 PCT/EP2005/055511

(87) 국제공개번호 WO 2006/045784

국제공개일자 2006년05월04일

(30) 우선권주장

10 2004 051 937.4 2004년10월25일 독일(DE)

(뒷면에 계속)

(56) 선행기술조사문헌

US3783250 A

US5233615 A

전체 청구항 수 : 총 15 항

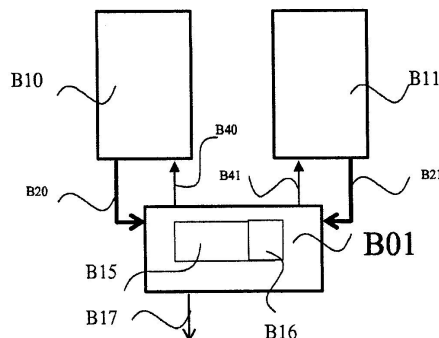
심사관 : 이경홍

(54) 적어도 2개의 처리 유닛들을 갖는 컴퓨터 시스템에서 모드전환 및 신호 비교를 위한 방법 및 장치

(57) 요약

본 발명은 각각 사전 설정된 클록으로 데이터들을 처리하는 적어도 2개의 처리 유닛을 포함하는 컴퓨터 시스템에서 데이터 비교 및 전환을 위한 방법 및 장치에 관한 것이며, 이 경우 전환 수단이 제공되어 적어도 2개의 작동 모드들 사이가 전환되고, 또한 비교 수단이 제공되어 제1 작동 모드는 비교 모드에, 제2 작동 모드는 실행 모드에 상응한다. 또한 본 발명은 처리 유닛에 따르는 클록 정보를 사전 설정된 데이터에 할당하는 동기화 수단이 제공되는 것을 특징으로 하며, 적어도 비교 수단은 상응하는 데이터에 대해서 상기 클록 정보를 고려한다.

대표도 - 도1



(72) 발명자

콜라니 요르크

독일 71717 바일슈타인 리스츠트백 9

그멜리히 라이너

독일 71254 딫칭엔 회엔백 2

빌 에버하르트

독일 72768 로이틀링엔 하임빌슈트라쎄 36

(30) 우선권주장

10 2004 051 950.1 2004년10월25일 독일(DE)

10 2004 051 952.8 2004년10월25일 독일(DE)

10 2004 051 964.1 2004년10월25일 독일(DE)

10 2004 051 992.7 2004년10월25일 독일(DE)

10 2005 037 243.0 2005년08월08일 독일(DE)

특허청구의 범위

청구항 1

각각 사전 설정된 클록으로 데이터들을 처리하는 적어도 2개의 처리 유닛을 포함하는 컴퓨터 시스템에서의 전환 및 데이터 비교 방법이며, 전환 수단이 제공되어 적어도 2개의 작동 모드들 사이가 전환되고, 또한 비교 수단이 제공되어 제1 작동 모드는 비교 모드에, 제2 작동 모드는 실행 모드에 상응하는, 컴퓨터 시스템에서의 전환 및 데이터 비교 방법에 있어서,

비교 모드에서 모든 데이터들이 비교되는 것이 아니라 목표에 따라 데이터가 선택되고, 상기 처리 유닛에 따르는 클록 정보를 선택된 데이터에 할당하는 동기화 수단이 제공되며, 적어도 비교 수단은 선택된 데이터들을 비교할 때 상기 클록 정보를 고려하는 것을 특징으로 하는, 컴퓨터 시스템에서의 전환 및 데이터 비교 방법.

청구항 2

제1항에 있어서, 상기 동기화 수단은 적어도 메모리 수단을 포함하며, 상기 메모리 수단 내에는, 할당된 클록 정보를 갖는 사전 설정된 데이터들이 저장되는 것을 특징으로 하는, 컴퓨터 시스템에서의 전환 및 데이터 비교 방법.

청구항 3

제1항 또는 제2항에 있어서, 사전 설정된 상기 데이터들이 상기 데이터들에 할당된 클록 정보에 따라 부가 처리되는 것을 특징으로 하는, 컴퓨터 시스템에서의 전환 및 데이터 비교 방법.

청구항 4

제1항 또는 제2항에 있어서, 사전 설정된 상기 데이터들이 상기 데이터들에 할당된 클록 정보로 제공되는 것을 특징으로 하는, 컴퓨터 시스템에서의 전환 및 데이터 비교 방법.

청구항 5

제1항 또는 제2항에 있어서, 상기 데이터들이 비교 수단에 입력되기 전에 버퍼링되는 것을 특징으로 하는, 컴퓨터 시스템에서의 전환 및 데이터 비교 방법.

청구항 6

제1항 또는 제2항에 있어서, 데이터 수신을 알리도록 핸드셰이크 인터페이스가 제공되는 것을 특징으로 하는, 컴퓨터 시스템에서의 전환 및 데이터 비교 방법.

청구항 7

제1항 또는 제2항에 있어서, 다음의 출력 데이터가 비교되어야 하는 프리세팅이 비교 신호를 통해서 실행되는 것을 특징으로 하는, 컴퓨터 시스템에서의 전환 및 데이터 비교 방법.

청구항 8

제1항 또는 제2항에 있어서, 비교되어야 하는 하나의 데이터에, 비교를 개시하는 식별부가 할당되는 것을 특징으로 하는, 컴퓨터 시스템에서의 전환 및 데이터 비교 방법.

청구항 9

각각 사전 설정된 클록으로 데이터들을 처리하는 적어도 2개의 처리 유닛을 포함하는 컴퓨터 시스템에서의 전환 및 데이터 비교 장치이며, 이 경우 전환 수단이 포함되어 적어도 2개의 작동 모드들 사이가 전환되고, 또한 비교 수단이 포함되어 제1 작동 모드는 비교 모드에, 제2 작동 모드는 실행 모드에 상응하는, 컴퓨터 시스템에서의 전환 및 데이터 비교 장치에 있어서,

비교 모드에서 모든 데이터들을 비교하는 것이 아니라 목표에 따라 데이터를 선택하는 수단이 제공되고, 상기 처리 유닛에 따르는 클록 정보가 선택된 데이터에 할당되도록 구성된 동기화 수단이 포함되며, 비교 수단은 선택된 데이터들을 비교할 때 상기 클록 정보를 고려하도록 구성되는 것을 특징으로 하는, 컴퓨터 시스템에서의

전환 및 데이터 비교 장치.

청구항 10

제9항에 있어서, 상기 동기화 수단은 적어도 메모리 수단을 포함하며, 메모리 수단은 할당된 클록 정보를 갖는 사전 설정된 데이터들이 상기 메모리 수단 내에 저장되도록 구성되는 것을 특징으로 하는, 컴퓨터 시스템에서의 전환 및 데이터 비교 장치.

청구항 11

제10항에 있어서, 상기 메모리 수단은 적어도 하나의 FIFO 메모리인 것을 특징으로 하는, 컴퓨터 시스템에서의 전환 및 데이터 비교 장치.

청구항 12

제9항에 있어서, 각각의 처리 유닛에 하나의 동기화 수단이 할당되는 것을 특징으로 하는, 컴퓨터 시스템에서의 전환 및 데이터 비교 장치.

청구항 13

제9항에 있어서, 전환 수단과 비교 수단이 전환 유닛 및 비교 유닛으로서 형성되며 서로 할당되는 것을 특징으로 하는, 컴퓨터 시스템에서의 전환 및 데이터 비교 장치.

청구항 14

제9항에 있어서, 데이터가 비교 수단에 입력되기 전에 버퍼링되도록 형성되는 적어도 하나의 입력 버퍼 메모리가 제공되는 것을 특징으로 하는, 컴퓨터 시스템에서의 전환 및 데이터 비교 장치.

청구항 15

제9항에 있어서, 데이터 수신을 알리도록 핸드셰이크 인터페이스가 제공되는 것을 특징으로 하는, 컴퓨터 시스템에서의 전환 및 데이터 비교 장치.

명세서

기술분야

[0001] 비교 모드에서 에러를 검출하기 위한 방법은 WO 01/46806 A1호에 공지되어 설명되어 있다. 여기서 데이터들은 2개의 처리 유닛들(ALUs)을 갖는 하나의 처리 유닛에서 동시에 처리되고 비교된다. 에러(소프트-에러, 과도 에러) 시, 2개의 ALUs들은 에러가 있는 데이터들이 제거되고 새로운(부분적으로 반복된) 중복 처리가 실행될 수 있을 때까지, 서로 독립적으로 작업한다. 이는 2개의 ALUs들이 서로 동기화 작업하고 결과가 정확한 클록으로 비교될 수 있는 것을 전제로 한다.

배경기술

[0002] 종래 기술에는 태스크가 중복으로 처리된, 에러 검출을 위한 비교 모드와 더 높은 성능을 실현하기 위한 실행 모드 사이가 어떻게 전환될 수 있는지가 공지되어 있다. 여기서, 비교 모드를 위한 처리 유닛들이 상호간에 동기화되는 것이 전제된다. 이를 위해서는, 메모리 내의 기록 시 결과 데이터를 서로 비교할 수 있기 위해서 2개의 처리 유닛들이 정지될 수 있고, 정확한 클록으로 동기화되어 작업하는 것이 필요하다. 이를 위해 하드웨어로의 간섭이 필요하고, 개별 방법들이 제안된다.

[0003] 이에 대해 특허 문서 EP 0969373 A2에서는, 중복 작업하는 처리 유닛 또는 처리 유닛이 서로 비동기로 작업하더라도 즉, 클록이 동일하지 않거나 공지되지 않은 클록 오프셋을 가질 때에도 상기 처리 유닛들의 결과의 비교가 보장된다.

[0004] 항공기 산업으로부터는, 표준 컴퓨터의 입력을 사용할 수 있고 다수결을 통해서 이를 확실히 처리함으로써 안전성과 연관된 작동을 개시할 수 있는 보팅 시스템이 공지되어 있다. 인터-처리 유닛과 인터-컨트롤 유닛 통신이 서로 결합된 시스템은 FME-시스템이며, 상기 시스템에서는 높은 정도의 중복에 의해 개별적 또는 복수의 에러의 경우에도 시스템이 여전히 작동하며 우주 비행을 위한 DASA에 의해서 발전되어 왔다(무정지형 컴퓨팅에 대한 국

제 심포지엄, FTCS-28(1998), 우주 적용을 위한 생존 가능 항공 전자 시스템(Urban 외), 페이지 372 내지 381). 상기 시스템은 심지어 비잔티움 에러(즉, 모든 부품들이 동일한 정보를 얻는 것이 아니라, 인트리거로부터 "의도적으로" 상이한 잘못된 정보들이 다양한 부품들에 배분되는 악성 에러)를 허용한다. 상기와 같은 시스템은 높은 비용으로 인해, 소량 제조된 특히 임계적인 시스템을 위해서 상업적으로 적용될 수 있다. 양산될 수 있으며 추가적으로 전환 가능성도 갖는, 비용면에서 유리한 해결책은 공지되어 있지 않다.

발명의 상세한 설명

- [0005] 따라서 본 발명의 목적은, 2개 또는 복수의 처리 유닛들로부터 작동 모드를 전환할 수 있는 전환 유닛 및 비교 유닛을 제공하는 것이며, 이 경우 처리 유닛의 구조에 대한 간섭 없이도 충분하고, 상기 목적을 위해 추가의 신호도 필요하지 않다. 이 경우, 다양한 처리 유닛들에 의해 다양한 디지털 또는 아날로그 신호들은 하나의 비교 모드 내에서 상호간에 비교되어야 한다. 상기 비교는 경우에 따라, 처리 유닛들이 다양한 클록 신호들로 작동되고 서로 동기화 작업되지 않을 때에도 가능해야 한다. 또한 본 발명의 목적은 상기 비교가 다양한 클록에 의해서도 실행될 수 있게 하는 수단 및 방법을 제공하는 것이다.
- [0006] 바람직하게, 각각 사전 설정된 클록으로 데이터들을 처리하는 적어도 2개의 처리 유닛들을 갖는 컴퓨터 시스템에서 전환 및 데이터 비교를 위한 방법이 사용된다. 이 경우 전환 수단이 제공되어 적어도 2개의 작동 모드들 사이가 전환되고 비교 수단이 제공되어 제1 작동 모드는 비교 모드에, 제2 작동 모드는 실행 모드에 상응하고, 처리 유닛에 따르는 클록 정보를 사전 설정된 데이터에 할당하는 동기화 수단이 제공되며, 적어도 비교 수단은 상응하는 데이터의 경우 상기 클록 정보를 고려하는 것을 특징으로 한다.
- [0007] 바람직하게, 할당된 클록 정보를 갖는 사전 설정된 데이터들이 저장된 메모리 수단을 적어도 동기화 수단이 포함하는 방법이 사용된다.
- [0008] 바람직하게, 사전 설정된 데이터들이 상기 데이터들에 할당된 클록 정보에 따라 부가 처리되는 방법이 사용된다.
- [0009] 바람직하게, 사전 설정된 데이터들이 상기 데이터들에 할당된 클록 정보로 제공되는 방법이 사용된다.
- [0010] 바람직하게, 상기 데이터들이 비교 수단에 입력되기 전에 버퍼링되는 방법이 사용된다.
- [0011] 바람직하게, 데이터 수신을 알리도록 핸드셰이크 인터페이스가 제공되는 방법이 사용된다.
- [0012] 바람직하게, 다음 출력 데이터가 비교되어야 하는 프리세팅이 비교 신호를 통해서 실행되는 방법이 사용된다.
- [0013] 바람직하게, 비교를 개시할 수 있는 식별부가, 비교되어야 하는 데이터에 할당된 방법이 사용된다.
- [0014] 바람직하게, 각각 하나의 사전 설정된 클록으로 데이터들을 처리하는 적어도 2개의 처리 유닛들을 갖는 컴퓨터 시스템에서 전환 및 데이터 비교를 위한 장치가 사용되며, 이 경우 전환 수단이 포함되어 적어도 2개의 작동 모드 사이가 전환되며, 비교 수단이 제공되어 제1 작동 모드는 비교 수단에, 제2 작동 모드는 실행 모드에 상응하며, 상기 장치는, 처리 유닛에 따르는 클록 정보가 사전 설정된 데이터에 할당되도록 구성된 동기화 수단이 포함되고 비교 수단은 상응하는 데이터들의 경우 클록 정보를 고려하도록 구성되는 것을 특징으로 한다.
- [0015] 바람직하게, 할당된 클록 정보를 갖는 사전 설정된 데이터들이 저장되도록 구성된 메모리 수단을 동기화 수단이 적어도 포함하는 장치가 사용된다.
- [0016] 바람직하게, 메모리 수단이 적어도 하나의 FIFO 메모리인 장치가 사용된다.
- [0017] 바람직하게, 각각의 처리 유닛에 하나의 동기화 수단이 할당된 장치가 사용된다.
- [0018] 바람직하게, 전환 수단과 비교 수단이 전환 및 비교 유닛으로서 형성되고 서로 할당되는 장치가 사용된다.
- [0019] 바람직하게, 적어도 하나의 입력 버퍼 메모리가 제공되고, 상기 메모리는 데이터가 비교 수단으로의 입력 전에 버퍼링되도록 형성되는 장치가 사용된다.
- [0020] 바람직하게, 데이터 수신을 알리도록 핸드셰이크 인터페이스가 제공된 장치가 사용된다.
- [0021] 바람직한 다른 실시예와 장점들은 청구범위 및 상세한 설명에 제시된다.

실시예

- [0042] 실행 유닛 또는 처리 유닛은 이하에서 프로세서/코어/CPU뿐만 아니라 FPU(플로팅 포인트 유닛), DSP(디지털 신

호 프로세서), 코프로세서로도 또는 ALU(산술 논리 유닛)으로서 표현될 수 있다. 하나의 시스템은 2개 또는 복수의 처리 유닛들에 의해서 고려된다. 기본적으로, 안전성과 연관된 시스템에서는 다양한 처리 유닛들에 가급적 다양한 태스크들이 제공됨으로써, 성능을 높이기 위해 상기 유형의 리소스들을 사용할 가능성이 있다. 대안적으로, 상기 리소스들 중 몇몇은, 동일한 태스크가 제공됨으로써 서로 중복적으로 사용될 수 있으며, 동일하지 않은 결과일 때 에러가 인식된다. 얼마나 많은 처리 유닛들이 제공되는지에 따라 복수의 모드들이 고려될 수 있다. 앞서 설명한 바와 같이, 듀얼-시스템에는 "비교" 및 "실행"의 2개의 모드가 존재한다. 3중-시스템에서는 3개의 모든 처리 유닛들이 동시에 작업하는 순수한 실행-모드와, 3개의 모든 처리 유닛들이 중복으로 계산되고 비교되는 순수한 비교 모드 외에도, 3개의 모든 처리 유닛들이 중복으로 계산되고 다수결 선택이 실행되는 2아웃오브 3-보팅 모드가 구현될 수 있다. 또한 예컨대 처리 유닛들 중 2개가 서로 중복으로 계산되어 그 결과가 비교되는 반면, 제3 처리 유닛은 다른, 병렬 태스크를 처리하는 혼합된 모드도 구현될 수 있다. 4개의- 또는 그 이상의 처리 유닛들을 갖는 시스템에서 명백하게 추가의 다른 결합 형태도 고려될 수 있다.

[0043] 달성해야 할 목적은, 제공된 처리 유닛들이, 상기 처리 유닛들에 구성된(예컨대 동기화의 목적으로) 구조에 대한 간섭 없이도, 작동 시 하나의 시스템에서 가변적으로 사용될 수 있는 것이다. 특수한 실시예에서, 각각의 처리 유닛들은 고유의 클럭으로 작동할 수 있으며 즉, 동일한 태스크의 처리가 비교의 목적으로 서로 비동기적으로 처리될 수 있다. 상기 목적은 일반적으로 넓게 사용될 수 있는 IP가 생성됨으로써 달성될 수 있으며, 이는 처리 유닛들의 사전 스위치오프 없이 임의의 시점에 작동 모드(예컨대 비교 모드, 실행 모드 또는 보팅 모드)가 전환될 수 있게 하며 서로 비동기일 수 있는 데이터 흐름의 비교 또는 보팅을 관리한다. 상기 IP는 칩으로서 실시될 수 있거나, 하나 또는 복수의 처리 유닛들과 함께 하나의 칩 상에 집적될 수 있다. 또한 상기 칩이 단 하나의 부재의 실리콘으로 구성된다는 것은 전체 조건이 아니며, 상기 칩이 분리된 모듈로 구현되는 것도 가능하다.

[0044] 상이한 처리 유닛들 사이의 동기화를 보장하기 위해, 일정하게 지속적인 개별 처리 유닛들의 프로그램 처리를 방지하는 신호들이 필요하다. 이를 위해 일반적으로 WAIT-신호가 제공된다. 실행 유닛에 WAIT-신호가 제공되지 않으면, 상기 유닛은 인터럽트에 의해서 동기화될 수도 있다. 이를 위해 동기화 신호(예컨대 도2의 M140)는 WAIT-입력부에 안내되지 않으며, 하나의 인터럽트 상에 인가된다. 상기 인터럽트는 정상적인 작동을 인터럽트하기 위해서, 처리 프로그램에 대해, 또한 다른 인터럽트들에 대해서도 충분히 높은 우선순위를 가져야 한다. 해당 인터럽트-루틴은 인터럽트된 프로그램으로 다시 복귀되기 전에, 특정의 수의 NOPs(데이터에 대해 작용하지 않는 무의미 명령어)만을 실시하며, 이로써 처리 프로그램의 추가의 처리를 지연시킨다. 경우에 따라 인터럽트 루틴에서는 시작과 종료 시 일반적인 메모리 작동이 실행되어야 하며, 이는 인터럽트를 통해 정상적인 프로그램 처리가 저해되지 않도록 하기 위해서이다.

[0045] 상기 과정은 동기화가 형성되었을 때까지(예컨대 다른 처리 유닛들은 예상되는 비교 데이터들을 전달), 계속 진행된다. 그러나 정확한 클럭 동기화 및, 다른 처리 유닛들과의 동위상은 상기 방법에 의해서 제한적으로만 보장될 수 있다. 따라서, 동기화를 위한 인터럽트-신호의 사용 시 비교될 데이터들은 비교 전에 UVE 내에 일시 저장되는 것이 추천된다.

[0046] 본 발명의 장점은 상용될 수 있는 임의의 표준-구조들이 사용될 수 있는 것인데, 이는 추가의 신호들이 필요하지 않고(하드웨어 구조에 대한 간섭 없음) 상기 부품들의 임의의 출력 신호들이 모니터링될 수 있기 때문이며, 상기 출력 신호들은 예컨대 액추에이터를 직접 제어하기 위해 사용될 수 있다. 이는 DACs 및 PWMs와 같은 변환기 구조들의 검사도 포함하며, 이는 이제까지 종래기술에 따라서는 비교에 의해서 직접 검사될 수는 없었다. 개별적 태스크 또는 SW-태스크들에 대한 검사가 필요하지 않는 한, 상이한 태스크들이 다양한 처리 유닛들에 배분되는 하나의 실행 모드로의 전환도 가능하다. 추가의 장점은, 하나의 비교 모드 또는 보팅 모드에서 모든 데이터들이 비교되지 않아도 되는 것이다. 비교될 또는 보팅될 데이터들만이 전환 유닛 및 비교 유닛에서 서로 동기화된다. 상기 데이터들의 선택은 전환 유닛 및 비교 유닛의 의도된 반응으로 인해 가변적이며(프로그래밍 가능하며), 각각의 처리 유닛 구조 및 적용예에 맞게 조정될 수 있다. 이로써 다양한 MC 또는 소프트웨어 부품들의 사용도 쉽게 가능한데, 중요하게 비교될 수 있는 결과만이 실제로도 비교될 수 있기 때문이다.

[0047] 또한 하나의 (외부) 메모리에 대한 각각의 액세스가 모니터링될 수 있으며, 또는 외부 I/O 모듈의 제어만이 모니터링될 수 있다. 내부 신호들은 외부 데이터 버스 및/또는 어드레스버스의 전환 모듈에 대해 소프트웨어 제어된 추가의 출력을 통해서 검사된다.

[0048] 비교 연산을 위한 모든 제어 신호들은 바람직하게는 프로그래밍 가능한 전환-유닛 및 보팅-유닛 내에서 발생되며, 비교 역시 상기 유닛에서 실행된다. 그 출력들이 서로 비교되어야 하는 처리 유닛들(예컨대 프로세서들)은

동일한 프로그램을 사용할 수 있으며, 이는 복사된 프로그램(이는 메모리 액세스 시 추가적으로 에러의 인식을 가능하게 한다) 또는 소프트웨어-에러를 인식하기 위한 다양화된 프로그램이다. 이 경우 처리 유닛들로부터 제공된 모든 신호들이 서로 비교될 필요는 없을 뿐만 아니라, 비교를 위한 특정의 신호를 식별부(어드레스 신호 또는 제어 신호)에 의해서 제공하거나 제공하지 않는 것도 가능하다. 상기 식별부는 전환 장치 및 비교 장치 내에서 평가되며 이로써 비교가 제어된다. 별도의 타이머는 사전 설정된 한계를 초과하는 시간 응답 내의 편차를 모니터링한다. 전환 유닛 및 비교 유닛의 수개의 모듈 또는 모든 모듈들은 하나의 칩, 하나의 공통 보드 상에 집적될 수 있으며 또는 공간적으로 분리되어 설치될 수도 있다. 이 경우 데이터와 제어 신호들은 적합한 버스 시스템에 의해서 서로 교환된다. 국부적으로 레지스터는 버스 시스템에 의해서 기록되며, 그 안에 저장된 데이터들 및/또는 어드레스/제어 신호들을 이용해서 과정들을 제어한다.

[0049] 도1에는 2개의 처리 유닛들(B10, B11)과 연결된 적용예를 위한 본 발명에 따른 전환 유닛(B01)의 기본 기능이 도시된다. 처리 유닛들(B10, B11)의 데이터, 제어 신호 및 어드레스 신호(B20 또는 B21)와 같은 다양한 출력 신호들이 전환 유닛(B01)에 연결된다. 또한 본 발명에 따른 구성 내에서 적어도 하나의 동기화 신호 즉, 비교 유닛들 중 하나에 연결된 2개의 출력 신호들(B40, B41)이 있다. 전환 유닛은, 2진수[비트(B16)]를 위해 적어도 하나의 메모리 요소를 갖는 적어도 하나의 제어 레지스터(B15)를 포함하며, 상기 비트는 비교 유닛의 모드를 전환한다. B16은 적어도 2개의 값 0, 1을 취하며, 처리 유닛의 신호들(B20, B21)을 통해서 또는 전환 유닛의 내부 프로세스를 통해서 세팅 또는 리세팅될 수 있다. B16이 제1 값으로 세팅되면, 전환 유닛은 비교 모드 내에서 작동한다. 상기 모드 내에서, B20으로부터 얻어질 모든 데이터 신호들은, 신호들(B20, B21)로부터의 제어 신호 및/또는 어드레스 신호들의 특정의 사전 설정된 비교-조건들이 충족되는 한, B21로부터의 데이터 신호들과 비교되며, 상기 신호들(B21, B21)은 데이터들의 타당성과 상기 데이터들을 위해 제공된 비교를 신호화한다. 2개의 신호들(B20, B21)에서의 비교-조건들이 동시에 충족되면, 상기 신호들로부터의 데이터들이 직접 비교되며 동일하지 않은 경우 에러 신호(B17)가 세팅된다. 단지 신호(B20 또는 B21)로부터의 비교-조건이 충족되면, 상응하는 동기화 신호(B40, B41)가 세팅된다. 상기 신호는 상응하는 처리 유닛(B10 또는 B11)에서 처리를 정지시키므로, 종래까지는 서로 비교될 수 없었던 상응하는 신호들의 릴레이는 방지된다. 신호(B40 또는 B41)는 각각의 다른 처리 유닛(B21 또는 B20)의 상응하는 비교-조건이 충족될 때까지, 세팅되어 있다. 이 경우 비교가 실시되며 상응하는 동기화 신호가 리세팅된다. 비교될 데이터들이 설명한 바와 같이 2개의 처리 유닛을 통해 다른 시간에 제공될 때 비교를 보장하기 위해, 상응하는 동기화 신호(B40 또는 B41)가 리세팅될 때까지 상응하는 처리 유닛의 데이터 및 비교 조건들이 상응하는 값으로 유지되는 것이 필요하거나, 먼저 제공된 데이터들은 비교가 실행될 때까지 전환 유닛 내에 저장되어야 한다. 어떤 처리 유닛이 먼저 데이터를 제공하는지에 따라, 상기 처리 유닛은, 다른 처리 유닛이 상응하는 비교 데이터를 제공할 때까지, 자신의 프로그램 또는 프로세스가 계속 처리되면서 대기해야 한다. 도1에 따른 전환 유닛의 특수한 실시예에서, 다른 처리 유닛이 아닌 해당 처리 유닛이 비교 데이터를 제공하지 않을 때 항상, 신호들(B40 또는 B41) 중 하나가 생략될 수 있다. B16이 제2 값으로 세팅되면, 동기화 신호들(B20, B21) 및 에러 신호(B17)는 항상 비활성이며 예컨대 값 0으로 세팅된다. 또한 비교는 발생하지 않으며 2개의 처리 유닛들은 서로 독립적으로 작동한다.

[0050] 본 발명에 따른 시스템에서 실질적인 부품은 비교기이다. 가장 단순한 형태가 도1a에 도시된다. 비교 부품(M500)은 2개의 입력 신호들(M510, M511)을 수신한다. 상기 비교 부품은 상기 신호들을 동일성에 대해 비교하며, 이 경우 바람직하게는 비트에 관한 동일성이다. 동일하지 않음이 검출되면 에러 신호(M530)가 활성화되고 신호(M520)는 비활성화된다. 동일한 경우, 입력 신호들(M510, M511)의 값이 출력 신호(M520)로 주어지며, 에러 신호(M530)는 활성화되지 않고 즉, "양호한"-상태를 신호화한다. 상기의 베이스-시스템으로부터, 다수의 다른 변형된 실시예들이 고려될 수 있다. 우선 부품(M500)은 소위 TSC-부품(totally self checking)으로서 실시될 수 있다. 이 경우 에러 신호(M530)는 적어도 2개의 라인("듀얼 레일") 상에서 외부 쪽으로 안내되며, 내부 디자인 검출 조치 및 에러 검출 조치에 의해, 비교 부품의 모든 가능한 에러 상황일 때 상기 신호가 적합하게 존재하는지 또는, 부적합하지만 검출 가능한 상태로 존재하는지가 보장된다. 본 발명에 따른 시스템의 사용 시 바람직한 실시예는 상기와 같은 TSC-비교기를 사용하는 것이다. 실시예의 제2 클래스는 2개의 입력들(M510, M511)(또는 M610, M611)이 동기화의 어떤 등급을 가져야 하는지와 관련하여 구별될 수 있다. 가능한 변형예는 클록 동기화로 특징되며 즉, 데이터들의 비교는 하나의 클록으로 실행될 수 있다. 입력들 사이의 위상 오프셋이 고정된 경우 동기 지연 요소가 사용됨으로써 약간의 변동이 발생하며, 상기 지연 요소는 상응하는 신호들을 예컨대 정수의 또는 절반의 클록 주기만큼 지연시킨다. 상기와 같은 위상 오프셋은 커먼 코즈 에러를 방지하기 위해 사용되며 즉, 이는 복수의 처리 유닛들에 동시에 작용할 수 있는 에러이다. 도1c에서, 이미지(M5)의 부품들 위에는 부품(M640)이 삽입되며, 이는 이전의 입력을 위상 오프셋만큼 지연시킨다. 바람직하게 상기 지연 요소는 비교 모드에서만 상기 요소가 사용되도록 비교기 내에 배치된다. 대안적으로 또는 보충적으로, 마찬가지로

로 비동기화를 허용할 수 있기 위해, 중간 버퍼가 입력 체인에 위치할 수 있다. 바람직하게 이는 FIFO-메모리로서 설계된다. 상기와 같은 버퍼가 제공되면, 버퍼의 최대 깊이까지의 비동기화도 허용될 수 있다. 이 경우에러 신호는 버퍼가 오버플로될 때에도 송출되어야 한다. 또한 비교기의 실시예는 신호(M520)(또는 M620)가 어떻게 생성되는지에 따라 구별될 수 있다. 바람직한 실시예는 입력 신호(M510, M511)(또는 M610, M611)를 출력부에 인가하고, 스위치를 통해서 상기 연결이 중단될 수 있게 하는 것이다. 상기 변형예의 특수한 장점은 실행 모드와 가급적 다양한 비교 모드 사이를 전환하기 위해 동일한 스위치가 사용될 수 있는 것이다. 대안적으로 상기 신호들은 비교기 내부의 버퍼 메모리로부터 생성될 수 있다. 상기 실시예의 마지막 클래스는 얼마나 많은 입력들이 비교기에 존재하고 상기 비교기가 어떻게 반응해야 하는지에 따라 구별될 수 있다. 3개의 입력일 때, 3개의 모든 신호들의 비교 또는 단 2개의 신호의 비교인 다수결 보팅이 실행될 수 있다. 4개 또는 그 이상의 입력일 때, 상응하는 복수의 변형예들이 고려될 수 있다. 상기 변형예들은 바람직하게 전체 시스템의 다양한 작동 모드들에 결합된다.

[0051] 일반적인 경우를 제시하기 위해, 도1b에는 바람직하게 사용되어야 하는 전환 유닛 및 비교 유닛의 일반적인 도면이 도시되어 있다. 고려될 n 실행 유닛으로부터, n 신호들(N140, ..., N14n)은 전환 부품 및 비교 부품(N100)으로 간다. 상기 부품들은 입력 신호들로부터 n 출력 신호들(N160, ..., N16n)까지를 발생시킬 수 있다. 가장 단순한 경우, "순수한 실행 모드"에서는 모든 신호들(N14i)이 상응하는 출력 신호들(N16i) 상에 유도된다. 이에 반대되는 한계 상황에서 "순수한 비교 모드"에서는 모든 신호들(N140, ..., N14n)이 단지 출력 신호들(N16i) 중 정확히 하나에만 유도된다.

[0052] 상기 도면에는, 다양하게 고려될 수 있는 모드들이 어떻게 형성될 수 있는지가 도시된다. 이를 위해 상기 도면에는 스위칭 로직(N100)의 로직 부품이 포함된다. 상기 부품은 이와 같은 부품으로 존재하지 않아도 되며, 그 기능이 제공되는지가 결정적이다. 부품은 우선 얼마나 많은 출력 신호들이 주로 제공되는지를 정한다. 또한 스위칭 로직(N110)은 입력 신호들 중 어떤 것이 출력 신호들 중 어떤 것에 기여하는지를 고정한다. 이 경우 하나의 입력 신호는 정확히 하나의 출력 신호에 기여할 수 있다. 수학식에서 다르게 형성되며 즉, 스위칭 로직을 통해서 하나의 기능이 규정되고, 상기 기능은 집합{N140, ..., N14n}의 각각의 요소에 양집합{N160, ..., N160n}의 하나의 요소를 할당시킨다.

[0053] 처리 로직(N120)의 기능은 어떠한 형태 내에서 입력부들이 상기 출력 신호에 기여하는지를, 출력부들(N16i) 각각에 대해 정한다. 상기 부품들은 고유의 부품들로서 제공되지 않아도 된다. 설명된 기능들이 시스템에서 구현되는 것은 또한 중요하다. 예컨대 일반성을 제한하지 않고서 다양한 변형 가능성을 설명하기 위해, 출력부(N160)가 신호들(N141, ..., N14m)에 의해 발생하는 것이 제시된다. $m=1$ 인 경우, 이는 간단히 신호의 도통에 상응하며, $m=2$ 인 경우 신호들(N141, N142)이 비교된다. 상기 비교는 동기 또는 비동기식으로 실행될 수 있으며, 비트 단위로 또는 유효 비트에 대해서만 또는 허용 오차 밴드를 가지면서도 실행될 수 있다. $m \geq 3$ 인 경우, 더 많은 가능성들이 있다. 제1 가능성은, 모든 신호들을 비교하는 것과, 적어도 2개의 다른 값들이 존재할 때 선택적으로 신호화될 수 있는 하나의 에러를 검출하는 것이다. 제2 가능성은, m 으로부터 k 를 선택하는 것이다($k \geq m/2$). 이는 비교기를 사용함으로써 구현될 수 있다. 선택적으로 신호들 중 하나가 편차가 있는 것으로 인식될 때, 하나의 에러 신호가 생성될 수 있다. 3개의 모든 신호들이 다를 때, 다를 수도 있는 하나의 에러 신호가 생성될 수 있다. 제3 가능성은, 상기 값을 하나의 알고리즘에 제공하는 것이다. 이는 평균값, 중앙값의 형성을 나타내거나 무정지형 알고리즘(FTA)의 사용을 나타낼 수 있다. 상기와 같은 FTA는 입력값의 극값을 삭제하며 나머지 값들 이상의 평균값을 산정하는 것을 기본으로 한다. 상기 평균값 산정은 나머지 값들의 전체 집합 이상으로, 또는 바람직하게는 HW 내에 쉽게 형성될 부분 집합 이상으로 실행될 수 있다. 이 경우 상기 값들을 실제로 비교하는 것이 항상 필요한 것은 아니다. 평균값 형성 시, 예컨대 단지 가산되거나 나누어져야 하며 FTM, FTA 또는 중앙값은 부분적인 분류를 필요로 한다. 경우에 따라, 충분히 큰 극값의 경우 선택적으로 하나의 에러 신호가 송출될 수 있다. 하나의 신호에 대한 복수의 신호들의, 언급한 다양한 처리 가능성은 간결함을 위해 비교 연산으로서 표현된다. 처리 로직의 과제는 각각의 출력 신호를 위한, 이로써 해당 입력 신호들을 위한 비교 연산의 정확한 구성을 정하는 것이다. 스위칭 로직(N110)(즉, 기능)의 정보와 처리 로직(즉, 출력 신호마다 즉, 기능값마다 비교 연산을 정함)의 결합 형태는 모드 정보이며 상기 모드 정보는 모드를 정한다. 상기 정보는 일반적인 경우 자연적으로 많은 값을 가지며 즉, 단지 하나의 로직 비트로 나타낼 수 없다. 이론적으로 고려할 수 있는 모든 모드들이 주어진 구현 내에서 의미있는 것이 아니면, 바람직하게는 허용된 모드들의 수가 제약된다. 강조하자면, 단 하나의 비교 모드가 있는 단지 2개의 실행 유닛들의 경우 전체 정보는 단 하나의 로직 비트로 압축될 수 있다. 하나의 실행 모드로부터 하나의 비교 모드로의 전환은 일반적인 경우, 실행 모드에서 상이한 출력들로 나타나고 비교 모드에서는 동일한 출력으로 나타나는 실행 유닛들을 특징으로 한다. 바람직하게 이는 실행 유닛의 하나의 부분 시스템이 제공됨으로써 구현되며, 이 경우 실행 모드에서는

부분 시스템에서 고려되는 모든 입력 신호들(N14i)이 상응하는 출력 신호들(N16i)로 바로 전환되는 반면, 비교 모드에서는 모두 하나의 출력으로 나타난다. 선택적으로 상기와 같은 전환은 페어링이 변경됨으로써도 구현될 수 있다. 이는 일반적인 경우 본 발명의 주어진 형태 내에서, 실행 모드와 비교 모드가 해당되도록 허용된 모드의 집합을 제한할 수 있더라도, 실행 모드와 비교 모드를 언급할 수 없는 것으로 나타난다. 그러나 실행 모드로부터 비교 모드로의 전환(또는 그 반대)에 대해서는 항상 언급할 수 있다. 소프트웨어에 의해서 제어되는 상기 모드들 사이의 작동 시 동역학적으로 전환될 수 있다. 전환은 예컨대 특수한 전환 명령어, 특수한 명령열, 명시적으로 표시된 명령어의 실행에 의해서 또는, 멀티프로세서 시스템의 실행 유닛들의 적어도 하나를 통한 특정 어드레스에 대한 액세스를 통해서 개시된다.

[0054]

도2에는 본 발명에 따른 전환 유닛 및 비교 유닛(M100)을 갖는 2-프로세서 시스템 또는 2-MC-시스템이 상세하게 설명되어 도시되며, 이 경우 선택적으로, 도시된 신호들 중 다양한 신호들이 생략될 수 있다. 상기 시스템은 2개의 처리 유닛들(M110, M111)과 하나의 전환 유닛 및 비교 유닛(M100)으로 구성된다. 각각의 처리 유닛으로부터, 데이터 신호(M120, M121) 및 어드레스/제어 신호(M130, M131)가 전환 유닛으로 가며, 각각의 처리 유닛은 선택적으로 전환 유닛으로부터 데이터(M150, M151) 및 제어 신호(M140, M141)를 다시 받는다. 유닛(M100)은 데이터(M160, M161)와 상태 정보(M169)를 송출하며, 처리 유닛들에도 전달될 수 있는 데이터(M170, M171) 및 제어 신호(M179)와 같은 신호들을 수신한다. M170, M171 및 M179에 의해 선택적으로, 유닛(M100)의 작동 모드도 처리 유닛과 무관하게 세팅될 수 있으며: 마찬가지로 프로세서들은 출력들(M120, M121)(예컨대 데이터 버스), 제어 신호 및 어드레스 신호(M130, M131)(기록)를 통해서 유닛(M100)에서 작동 모드를 세팅할 수 있으며, 작동 모드는 예컨대 실행 모드(비교 없음) 또는 비교 모드(신호들(M120, M121) 및/또는 예컨대 주변 유닛들로부터 나온 신호들(M170, M171)의 비교를 가짐)이다. 실행 모드에서 출력들(M120, M121)은 경우에 따라 제어 신호들과 연결되어 출력들(M160, M161)로 전달되며 반대로 입력들(M170, M171)은 M150, M151로 전달된다. 비교 모드에서는 출력들이 비교되고 바람직하게 예러가 있는 경우에만 M160, M161로 전달되며, 선택적으로 2개의 출력들이 사용되거나 2개 중 하나만 사용된다. 마찬가지로 처리 유닛들에 전달된 입력 데이터들(M170, M171)의 검사가 가능하다. 비교 모드에서 신호들의 비교에 예러가 있을 때, 예컨대 에러-세이프(error-safe)식으로 외부 쪽으로 신호화되는(상태 정보(M169)의 성분) 더블-레일 신호들에 의해 에러 신호가 생성된다. 상태(M169)는 실행 유닛의 신호들의 시간적 오프셋에 대해 작동 모드 또는 정보들을 포함할 수 있다. 사전 설정된(프로그래밍 가능한) 시간 인터벌 내에 처리 유닛의 비교 데이터들이 제공되지 않은 경우, 에러 신호도 활성화된다. 에러의 경우, 출력들(M160, M161)이 중단될 수 있다(페일 사일런트 특성). 이는 디지털 및 아날로그 신호에도 연관될 수 있다. 그러나 상기 출력 구동단은 지연되지 않은(일시 저장되지 않은) 처리 유닛의 출력 신호들(M120, M121)도 송출할 수 있으며, 추후의 에러 검출 가능성도 갖는다. 이는 에러 허용 시간 즉, (지연된) 시스템이 아직 심하게 에러에 반응하지는 않으므로 보정할 가능성도 아직 있는 그러한 시간을 초과하지 않는 한, 안전성과 연관된 시스템에 의해서 허용된다. 또한 UVE 내에 안내되지 않는 출력 신호들(M180, M181)과 처리 유닛의 내부 신호들은 적어도, 계산된 그 값에 대해서 비교될 수 있으며, 이는 상기 값이 비교의 목적으로 출력들(M120, M121)에 출력됨으로써 가능하다. 이에 상응하게, M100을 통해서 오는 것이 아닌 입력 신호들(M190, M191)에 의해서도 실행될 수 있다. 유닛(M100)을 모니터링하기 위해, 선택된, 또는 모든 신호들(M160, M161)에 대해서, M170, M171 또는 M190, M191을 통해서 이들을 다시 판독하는 것이 가능할 수 있다. 이로써 비교 모드에서도, 유닛(M100)으로부터 에러가 있는 신호가 검출되는 것이 보장될 수 있다. (OR-연결에서) M100, M110, M111이 액세스되는 적합한 스위치오프 경로를 통해, 전체 시스템의 페일 사일런트 특성이 형성될 수 있다.

[0055]

도3에는 도2로부터의 전환 유닛 및 비교 유닛(M100)의 가능한 구현이 상세하게 도시된다. 유닛(M100)은 모드(실행/비교)를 나타내는 적어도 하나의 비트를 갖는 컨트롤 레지스터(M200) 및, 비교 모드에서의 에러 상태를 나타내는 적어도 하나의 비트를 갖는 상태 레지스터(M220)를 포함한다. WAIT 및 인터럽트 신호는 각각 2개의 처리 유닛들을 위한 컨트롤 레지스터 내에서 추가의 비트에 의해 제어된다. 이 경우 또한 경우에 따라, 예컨대 작동 모드 전환에 대한 준비를 위한 동기화 목적으로 또한 에러 처리 목적으로 다양한 인터럽트들 사이가 구분된다. 선택적으로 예컨대, 내부 또는 외부 위치독의 제어를 위해 처리 유닛들 사이에 최대 허용된 시간차(클록 주기수 내에서)를 갖는 M240 및, 예컨대 데이터 레지스터의 오버플로를 방지하기 위해 WAIT 신호 또는 인터럽트 신호들을 이용해 가장 신속한 프로세서가 임시로 정지 또는 지연되어야 하는 시간차값(클록 주기수)을 갖는 M241과 같은 추가의 컨트롤 레지스터가 있다. 상태 레지스터(M220) 내에는, 예컨대 에러-비트 외에도, 처리 유닛들 사이의 클록 오프셋이 현재 얼마나 큰지가 저장된다. 이를 위해 예컨대 적어도 하나의 타이머(230)는, (어드레스 및 제어 신호에 의한 예컨대 특정의 어드레스 영역) 특수하게 나타나는 데이터값이 먼저 제공될 때 하나의 처리 유닛에 의해 시동되며, 또한 상응하는 데이터값이 제2 처리 유닛에 의해서 제공될 때마다 항상 상기 타이머의 값은 상태 레지스터 내로 전달된다. 또한 타이머는, 프로그램 플로우들이 상이할 때에도 WCET(최

악 실행 시간)에 상응하게 모든 처리 유닛들이 하나의 데이터를 제공해야 하도록, 바람직하게 세팅된다. 사전 설정된 타이머의 값이 초과되면, 에러 신호가 송출된다. 처리 유닛들의 출력들(M120, M121)은, 상기 출력들이 디지털 데이터일 때 또한 이들이 정확한 클럭으로 제공되지 않는 한, M100에서 특히 비교 모드를 위해 버퍼 메모리(M250, M251) 내에 저장된다. 바람직하게 상기 메모리는 FIFO로서 실시될 수 있다. 상기 메모리가 단지 1의 깊이만을 가지면(레지스터), 데이터 손실을 막기 위해 예컨대 WAIT-신호는 비교가 실행될 때까지 추가의 값의 출력을 지연시킬 수 있다. 또한 입력 메모리들(M250, M251)로부터의, 또한 직접적 입력들(M120, M121 또는 M170, M171)로부터의 디지털 데이터들을 서로 비교하는 하나의 비교 유닛(M210)이 있다. 상기 비교 유닛은 메모리 유닛(M250, M251) 내에 직렬 데이터들이 수신되어, M210에서 비교되는 병렬 데이터들로 변환될 수 있을 때, 직렬 디지털 데이터들(예컨대 PWM-신호들)도 서로 비교할 수 있다. 마찬가지로 비동기 디지털 입력 신호들(M170, M171)은 추가의 메모리 유닛들(M270, M271)에 의해서 동기화될 수 있다. 입력 신호들(120, 121)과 마찬가지로 상기 디지털 입력 신호들도 바람직하게 FIFO 내에 일시 저장된다. 실행 모드와 비교 모드 사이의 전환은 컨트롤 레지스터에서 모드-비트의 세팅 또는 리세팅을 통해서 실행되므로, 예컨대 2개의 처리 유닛들 내에 상응하는 인터럽트가 야기된다. 상기 비교 자체는 제공된 데이터들(M120, M121) 및, 이에 속하는 어드레스 및 제어 신호들(M130, M131)을 통해서 유발된다. 이 경우 M120 및 M130 또는 M121 및 M131로부터 특정의 신호들이 식별부로서 기능할 수 있으며, 이들은 할당된 데이터들의 비교가 실행되어야 하는지의 여부를 제공한다.

[0056] 이는 도1의 단순한 전환에 대한 추가의 실시예이다. 여기서 바람직하게, 인터럽트-루틴에 의한 비교 모드로의 전환 시 2개의 처리 유닛들을 위해 동일한 시작-조건들이 형성되도록 다양한 준비들이 제공된다. 이로써 상기 처리 유닛이 완성되면, 상기 처리 유닛에 의해, 프로세서에 맞춰진 레디-비트가 컨트롤 레지스터 내에 세팅되며, 처리 유닛은 다른 처리 유닛이 그 레디-비트를 통해 백업을 신호화할 때까지(도6의 컨트롤 레지스터의 상세한 설명 참조), 대기 상태에 머무른다.

[0057] 비교 유닛에서, 마찬가지로 아날로그 데이터들은 그를 위해 특수하게 적합한 아날로그 비교 유닛(M211)(analog compare unit) 내에서 서로 비교될 수 있다. 그러나, 아날로그 신호들의 출력이 서로 충분히 동기화되어 실행되거나, 아날로그 비교 유닛에서 구현된 ADC를 통해 디지털화된 데이터들이 상기 유닛에 제공되는(도12 내지 도14의 추가의 실시예들 참조) 것이 전제된다. 동기화는, 앞서 설명한 바와 같이 처리 유닛들(데이터, 어드레스 신호 및 제어 신호들)의 디지털 출력을 서로 비교해서 매우 신속한 처리 유닛을 대기하게 함으로써, 도달된다. 이러한 목적으로, 아날로그 신호의 원으로서 처리 유닛에서 처리된 디지털 신호들도, 이와 같지 않은 경우 상기 신호들이 외부에서는 필요하지 않더라도, 출력들(M120, M121)을 통해서 유닛(M100)에 주어질 수 있다. 아날로그 신호들의 비교에 대해 추가적인, 상기의 중복 비교에 의해, 계산상의 에러가 매우 일찍 검출될 수 있으며 또한 처리 유닛들의 동기화가 쉬워진다. 아날로그 신호들의 비교는 처리 유닛의 DAC(디지털 투 아날로그 변환기)를 위해 추가로 에러를 검출한다. DCSL-아키텍처의 다른 구조에서는 상기와 같은 가능성이 주어지지 않는다. 주변 유닛의 아날로그 입력 신호들을 위해 비교도 가능하다. 특히 동일한 시스템 변수의 중복 센서 신호들에 관한 것일 때, 추가의 동기화 조치는 필요하지 않으며 경우에 따라 센서 신호들의 유효성을 나타내는 제어 신호만이 필요하다. 아날로그 신호의 비교의 구현은 더 상세하게 도시된다.

[0058] 도4는 적어도 $n+1$ 처리 유닛을 갖는 멀티프로세서 시스템을 도시하며, 또한 상기 부품들은 다시 복수의 부분-처리 유닛들(상응하는 추가 부품들을 갖는 CPUs, ALUs, DSPs)로 구성될 수 있다. 상기 처리 유닛들의 신호들은 도2에 따른 2-시스템에서 설명되었던 바와 같이, 정확히 전환 유닛 및 비교 유닛에 접속된다. 따라서 상기 도면에서 모든 부품들 및 신호들은 도2의 상응하는 부품들 및 신호들과 내용적으로 동일한 의미를 갖는다. 전환 유닛 및 비교 유닛(M300)은 멀티프로세서-시스템 내에서 실행 모드(모든 처리 유닛들이 상이한 태스크들을 처리), 다양한 비교 모드(2개 또는 그 이상의 처리 유닛들의 데이터들이 비교되어야 하며 편차 시 에러가 신호화되어야 한다) 및 다양한 보팅 모드(사전 설정된 상이한 알고리즘에 따른 편차 시 다수결) 사이를 구분할 수 있다. 각각의 처리 유닛에 대해서는, 어떤 모드에서 상기 처리 유닛이 작동하는지와 경우에 따라 다른 어떤 처리 유닛과 함께 상기 모드에서 작동하는지 별도로 결정될 수 있다. 전환이 정확히 실행되는 것은, 이하에서 도6에 따른 컨트롤 레지스터의 설명 시에 계속해서 제시된다.

[0059] 도5는 $n+1$ 처리 유닛들을 갖는 멀티프로세서 시스템을 위한 전환 유닛의 가능한 구현을 도시한다. 각각의 처리 유닛에 대해, 적어도 하나의 컨트롤 레지스터(M44i)가 전환 모듈 및 비교 모듈의 제어 유닛 내에 제공된다. 컨트롤 레지스터의 바람직한 세트는 도6에 실시적으로 도시되며 설명된다. 이 경우 M44i는 각각 컨트롤 레지스터(Ci)에 상응한다. 컨트롤 레지스터 내의 다양한 실시예들이 고려될 수 있다. 에러 검출 샘플 또는 에러 허용 샘플이 사용되어야 하는지는 적합한 비트 조합 형태에 의해서 설명될 수 있다. 유닛(M300) 내로 삽입된 입력에 따라, 어떤 타입의 에러 허용 샘플(2 아웃오브 3, 중앙값, 2 아웃오브 4, 3 아웃오브 4, FTA, FTM...)이 사용될

지가 제시될 수 있다. 또한 어떤 출력을 도출할지를 구성할 수 있다. 또한 어떠한 데이터에 대한 어떤 부품이 상기 구성에 영향을 미칠 수 있는지에 대해 실시예를 형성할 수 있다.

[0060] 관련 처리 유닛들의 출력 신호들은 전환 유닛 내에서 서로 비교된다. 신호들이 필수적으로 정확한 클럭으로 처리되지는 않기 때문에, 데이터들의 일시 저장이 필요하다. 이 경우 전환 유닛에서 데이터들도 비교될 수 있으며, 이들은 다양한 처리 유닛들에 의해 큰 시간차를 갖고 전환 유닛에 주어진다. 버퍼 메모리를 사용함으로써(예컨대 FIFO-메모리로서 형성됨: 선입선출(first-in, first-out) 또는 다른 버퍼 형태), 먼저 하나의 처리 유닛으로부터 복수의 데이터들이 수신될 수 있는 반면, 다른 처리 유닛들은 아직 데이터들을 제공하지 않는다. 이 경우 2개의 처리 유닛들의 동기화에 대한 수치는 FIFO-메모리의 점유율이다. 사전 설정된 특정의 점유율을 초과하면, 처리 시 최대로 선행된 처리 유닛은 제공된 WAIT-신호를 통해서 또는, 적합한 인터럽트-루틴을 통해서 임시로 정지되는데, 이는 처리 시 선행된 보다 느린 처리 유닛들을 대기하기 위해서이다. 이 경우 모니터링은 외부에 제공될 수 있는 모든 처리 유닛의 신호들에까지 연장되어야 하며; 이는 아날로그 신호 또는 PWM-신호도 포함한다. 전환 유닛에는 상기 신호들의 비교를 허용하는 구조들이 제공된다. 추가로, 비교할 데이터들 사이의 최대 시간 편차가 사전 설정되어 적어도 하나의 타이머에 의해서 모니터링되는 것이 제시된다.

[0061] 일반적인 경우 2개 이상의 처리 유닛들이 하나의 공통 전환 유닛에 의해서 서로 연결되면, 각각의 처리 유닛들에 대해서는 하나의 컨트롤 레지스터가 필요하다. 상기 컨트롤 레지스터의 특수한 실시예는 도6에 설명된다. 각각의 컨트롤 레지스터(Cx)의 (n+1) 하부 비트(B500x 내지 B50nx)들은 n+1 프로세서/처리 유닛들에 분명히 할당된다. 컨트롤 레지스터(Cx)의 비트(B514x)는 한편으로 비교/보팅과 다른 한편으로 병렬 작업 사이클을 전환하며, 도1의 B16의 값에 상응한다. 비트(B513x)는 해당 처리 유닛이 비교를 위해 준비되었는지(레디)를 제시하며, 비트(B512x)는 동기화 신호(WAIT 또는 INTERRUPT)를 제어하고, 비트(B511x)는 인터럽트를 통해 상응하는 처리 유닛(x)을 비교를 위해 준비하기 위해 사용될 수 있다. 상응하게, 비트(B5110x)는 처리 유닛을 병렬 모드로 되돌리는 인터럽트를 제어한다.

[0062] 컨트롤 레지스터(Ck)의 B50ik와 B50kk가 1로 세팅되면($0 \leq i, k \leq n$), 상기 실시예에서 이는, 처리 유닛(i)의 출력들이 처리 유닛(k)의 출력들과 비교되어야 하는 것을 의미한다. 추가로 B50jk도 동일하게 1이면, i, j 및 k 사이가 보팅되어야 하며 보팅 결과는 UVE의 출력(k)에 출력된다($0 \leq i, j, k \leq n$). 이를 위해 처리 유닛들의 각각의 그룹에 대해, 보팅 또는 단 하나의 다수결 비교의 특수한 형태가 정해질 수 있으며, 이는 이미지 M4에 대한 설명에서 열거했던 바와 같다. 일반적으로 비교될/보팅될 처리 유닛들(i)(컨트롤 레지스터(Ck) 내에서)을 위한 모든 비트들(B50ik)은, UVE의 출력(k)에서 보팅 결과가 출력되어야 할 때, 세팅되어야 한다. 다른 출력에서의 병렬 출력도 가능하다.

[0063] 컨트롤 레지스터(i)($0 \leq i \leq n$)의 B50ii 내의 1은, 비교 유닛의 출력(i)이 활성화되어야 하는 것을 나타낸다. 모든 컨트롤 레지스터(Ci)가 상응하는 메모리 지점(B50ii)에서만 1을 가지면($i = 0, 1, \dots, n$), 실행 모드 내에서 모든 처리 유닛들은 임의의 상이한 프로그램 및 고유의 출력 신호들으로써 작동한다. 모든 n+1 하부 비트들(B50ik)이 동일하게 1이고($i = 0, 1, \dots, n$) 또한 B514k가 세팅되면, 모든 처리 유닛들의 출력 신호들이 다수결(보팅)에 의해서 선택되어 UVE의 출력(k)에서 송출되며; n=1일 때 단 하나의 비교만이 발생한다.

[0064] 이하의 실시예들에서는 예컨대, 복수의 처리 유닛들을 갖는 하나의 시스템 내에서 비교/보팅으로의 전환 시 시퀀스가 어떻게 나타날 수 있는지가 설명된다. 컨트롤 레지스터(Ci) 내의 비트(B514i)는 비교 또는 보팅을 활성화하기 위해 세팅된다. 상기 비트는 처리 유닛 자체를 통해서 또한, 특정의 시스템 상태들, 시간 조건들 또는 다른 조건들(예컨대 특정의 메모리 영역, 에러 또는 비타당성에 대한 액세스)에 따라 전환 유닛 및 비교 유닛에 의해서도 세팅될 수 있다. 비트(B50ii, B50ki)가 B514i로 세팅되면, UVE를 통해서 자동으로 비트(B511i, B511k)가 세팅되므로, 처리 유닛들(i, k)에서 인터럽트가 개시된다. 상기 인터럽트에 의해, 처리 유닛들은 특정의 프로그램 지점으로 점프하며, 비교 모드로의 전환을 위한 특정의 초기화 단계가 실행된 다음, 전환 유닛 및 비교 유닛에 응답(레디)이 출력된다. 상기 레디-신호는 처리 유닛의 해당 컨트롤 레지스터(Ci)에서 인터럽트-비트(B511i)를 자동으로 리세팅하며 동시에 웨이트-비트(B512i)를 세팅한다. 연관된 처리 유닛들의 모든 웨이트-비트가 세팅되면, 이들은 전환 유닛 및 비교 유닛에 의해 동시에 리세팅된다. 그 후 처리 유닛들은 모니터링될 프로그램 부분들의 처리에 의해서 시작된다. 바람직한 일 실시예에서, 세팅된 비트(B514i)에 의한 컨트롤 레지스터(Ci) 상의 기록은 로킹(HW 또는 SW)에 의해서 방지된다. 이는, 비교의 구성이, 처리중이 아닐 때 변경될 수 있도록 한다. 컨트롤 레지스터(Ci) 내의 변경은 비트(B514i)의 리세팅 후에야 가능하다. 상기 리세팅은, 정상 모드(병렬 작동)로의 전환을 위한 연관된 모든 처리 유닛들의 컨트롤 레지스터 내의 비트(B510x)의 세팅을 통해 해당 처리 유닛들에 인터럽트를 일으킨다. 모든 컨트롤 레지스터 서로간의 일관성은 유저 요구에 상응하게 모니터링되며, 에러 상황인 경우 상태 정보의 부분인 에러 신호가 생성된다. 따라서 예컨대, 독립적

인 복수의 비교 프로세스 또는 보팅 프로세스를 위해서 하나의 처리 유닛이 동시에 사용되지 않을 수 있는데, 그럴 경우 동기화가 보장되지 않기 때문이다. 그러나 데이터 신호들의 출력 없이도, 단지 동일하지 않을 때 하나의 에러 신호를 생성할 목적으로 복수의 처리 유닛들의 비교도 고려할 수 있다. 추가의 실시예에서, 하나의 비교 또는 하나의 보팅에 참여한 처리 유닛들의 복수의 또는 모든 컨트롤 레지스터 내의 엔트리는 동일한 유형으로 실행되며 즉, 상기 처리 유닛들의 상응하는 비트들은 여기서 동일한 유형으로 세팅되고 경우에 따라서는, 출력을 제어하는 고유의 비트 i 를 제외하고 세팅된다.

[0065] 도7에는 중심 보팅을 위한 보팅-유닛(Q100)이 도시된다. 보팅은 적합한 하드웨어 및 소프트웨어에 의해 실행될 수 있다. 이를 위해 보팅-알고리즘(예컨대 정확한 비트의 보팅)이 제공된다. 보팅-유닛(Q100)은 복수의 신호들(Q110, Q111, Q112)을 수신하며, 이로부터 출력 신호(Q120)를 형성하고, 상기 출력 신호는 보팅(예컨대 n 에서 m -선택)을 통해 생성된다. 비교 시 에러가 발생하면, 해당 컨트롤 레지스터에는 에러-비트가 세팅된다. 보팅 시 해당 처리 유닛의 데이터가 무시되며; 단순한 비교 시 출력이 중단된다. 프로그래밍된 시간의 종료 전 적절한 때에 사용 가능하지 않은 모든 데이터들은 에러와 마찬가지로 취급된다. 에러-비트의 리세팅은 시스템에 따라 실행되며, 경우에 따라서는 해당 처리 유닛의 복구를 가능하게 한다. 처리 유닛 및/또는 보터(voter)가 공간적으로 동심이 아니게 배치되는 경우, 도8에 따라 적합한 버스 시스템에 연결된, 분산된 보팅도 가능하다. 도8에서 분산된 보팅 유닛(Q200)은 제어 유닛(Q210)에 의해서 제어된다. 상기 보팅 유닛은 버스 시스템(Q221, Q222)을 통해서 연결되며, 상기 버스 시스템을 통해서 데이터들을 수신하고, 또한 상기 지점에서 데이터들을 다시 출력한다. 활성 출력 비트를 갖는 컨트롤 레지스터 내에서의 비교 비트 및 보팅 비트의 리세팅은, 다시 병렬 작동으로 되돌아가는 관련 처리 유닛들 내에 인터럽트를 일으킨다. 이 경우 각각의 처리 유닛은 별도로 관리되는 상이한 벡터 어드레스를 가질 수 있다. 프로그램 처리는 동일한 프로그램 메모리에 의해서도 실행될 수 있다. 그러나 액세스는 별도로이며 일반적으로 상이한 어드레스로 간다. 병렬 모드에 대한 비교 시 안전성과 관련된 부분이 작은 한, 복사된 안전성 부분을 갖는 고유의 프로그램 메모리가 경우에 따라 비용이 적게 드는지가 검출된다. 데이터 메모리도 실행-모드 시에 공통으로 사용될 수 있다. 그 후 액세스는 예컨대 AHB/ABP-버스에 의해서 차례대로 실행된다. 특수하게는, 에러-비트가 시스템에 의해서 평가되어야 하는 것이 언급된다. 에러 상황일 때 확실한 스위치오프를 보장하기 위해, 안전성과 연관된 신호가 적절한 형태로 중복 구현된다(예컨대 1 아웃 오브 2코드). 도1, 도2, 도3, 도4 및 도5에 따른 종래의 UVEs에서는 우선, 처리 유닛들이 동일한 또는 서로에게서 유도된 클록으로 작동하는 것이 제시되었으며, 상기 클록들은 서로 일정한 위상 관계에 있다. 위상 관계가 변하는 상이한 발진기 또는 제너레이터로부터의 클록이 처리 유닛들을 위해 사용되면, 이로써 생성된 신호들이 클록 도메인을 변경할 때 상기 신호들을 동기화해야 한다. 동기화 요소(M800)은 도9에 도시된다. 특히 디지털 데이터들을 확실하게 저장하고 비교하기 위해, 신호 흐름 내에서 임의의 지점에 설치될 수 있는 동기화 장치(M800)이 필요하다. 상기 장치는, 데이터들을 제공하는 처리 유닛의 클록(M830)을 이용한 데이터(M820)의 저장을 한 번 보장한다. 관독을 위해, 데이터(M840)를 계속해서 처리하는 클록이 사용된다. 상기과 같은 동기화 단(M800)은 복수의 데이터들을 저장할 수 있기 위해(도9 참조) FIFO로서 구성될 수 있다. 일반적인 경우 데이터들의 동기화만으로는 충분하지 않으며, 데이터들의 제공 신호도 수신 클록과 함께 동기화되어야 한다. 이를 위해 또한 요구 신호(M850) 및 수신 알람 신호(M880)에 의해 전달을 보장하는 핸드셰이크-인터페이스가 필요하다(도10). 상기과 같은 인터페이스는 하나의 클록 도메인으로부터 다른 클록 도메인으로 데이터들의 확실한 전달을 보장하기 위해 클록 도메인들이 변경될 때마다 항상 필요하다. 기록 시, 영역(Q305)의 데이터(M820)는 클록(M830)을 이용하여 레지스터 셀(M800)에서 동기화되어 제공되며 기록 요구 신호(M850)는 데이터의 제공을 나타낸다. 상기 기록 요구 신호는 영역(Q306)으로부터 클록(M860)을 이용하여 메모리 요소(M801) 내로 전달되어, 동기화된 신호(M870)로서 데이터의 제공을 나타낸다. 클록(M860)의 다음 번 활성 클록 에지를 이용하여, 동기화된 데이터(M840)가 전달되며 확인 신호(M880)가 반송된다. 상기 확인 신호는 다른 메모리 요소(M801) 내의 클록(M830)에 의해서 신호(M890)로 동기화되며 이로써 데이터 제공이 종료된다. 그 후 새로운 데이터들이 해당 레지스터 내로 기록될 수 있다. 상기과 같은 인터페이스는 종래 기술에 공지되어 있으며 특수한 실시예에서 추가의 코딩에 의해서, 수신 알람 신호를 기다리지 않고서도 특히 신속하게 작동할 수 있다. 특수한 실시예에서 상기 메모리 요소(M800)는 FIFO-메모리(선입, 선출)로서 구성된다.

[0066] 도11 내지 도14의 아날로그 신호들의 비교를 위한 회로들은, 비교될 아날로그 신호들을 제공하는 처리 유닛들이, 상기 비교가 바람직하도록 서로 동기화되는 것을 전제 조건으로 한다. 상기 동기화는 도1의 상응하는 신호들(B40 및 B41)을 통해서 도달될 수 있다.

[0067] 도11은 차동 증폭기를 도시한다. 상기 요소에 의해서, 2개의 전압들이 서로 비교될 수 있다. 여기서 B100은 연산 증폭기이며, 그 음의 입력부(B101)에는 값(R_{in})을 갖는 저항(B110)에 의해서 입력 신호(B111)에 연결된 신

호(B141)가 연결되고 상기 입력 신호에는 전압값(V_1)이 인가된다. 양의 입력부(B102)는 값(R_{in})을 갖는 저항(B120)에 의해서 입력부(B121)에 연결된 신호(B142)에 연결되며, 상기 입력부에는 전압값(V_2)이 인가된다. 상기 연산 증폭기의 출력부(B103)는 전압값(V_{out})을 갖는 출력 신호(B190)에 연결된다. 상기 신호(B190)는 값(R_f)을 갖는 저항(B140)에 의해 신호(B141)에 연결되며, 신호(B142)는 값(R_f)을 갖는 저항(B130)에 의해서 신호(B131)에 연결되고, 상기 신호는 아날로그 기준점(V_{agnd})의 전압값을 갖는다. 출력 전압은 이하의 식에 따라, 위에 제시된 전압값 및 저항값과 비교될 수 있다.

수학식 1

$$V_{out} = R_f / R_{in} (V_2 - V_1)$$

차동 증폭기가 일반적으로 CMOS에서와 같이, 하나의 양의 작동 전압으로만 작동되면, 아날로그-접지(V_{agnd})로서 작동 전압과 디지털-접지 사이의 하나의 전압이 선택되는데, 일반적으로는 평균 전위이다. 2개의 아날로그 입력 전압들(V_1 및 V_2)이 단지 미미하게만 차이가 나면, 출력 전압(V_{out})은 아날로그 접지에 대해서 단지 작은 차이(V_{diff})만을 포함한다(양 또는 네가티브). 2개의 비교기에 의해서, 출력 전압이 $V_{agnd} + V_{diff}$ (도12) 상부에, 또는 아날로그 기준점과 관련하여 $V_{agnd} - V_{diff}$ 하부에 위치하는지(도13)가 검사된다. 여기서 도12에서는 입력 신호(B221)가 값(R_1)을 갖는 저항(B150)에 의해서 신호(B242)에 연결되며, 상기 신호는 연산 증폭기(B200)의 양의 입력부(B202)에 연결된다. 또한 상기 신호(B242)는 값(R_2)을 갖는 저항(B160)에 의해서 신호(B231)에 연결되고 상기 신호는 디지털 기준 전위(V_{dngg})로서 사용된다. 연산 증폭기의 음의 입력부(B201)는 입력 신호(211)에 연결되며, 입력 신호는 기준 전압(V_{ref})의 전압값을 갖는다. 연산 증폭기(B200)의 출력부(B203)는 출력 신호(B290)에 연결되며, 상기 출력 신호는 전압값(V_{oben})을 갖는다. 도13에는 상응하게 입력 신호(B231)가 값(R_3)을 갖는 저항(B170)에 의해서 신호(B342)에 연결되며, 상기 신호는 연산 증폭기(B300)의 음의 입력(B301)에 연결된다. 상기 신호(B342)는 또한 값(R_4)을 갖는 저항(B180)에 의해 신호(B331)에 연결되며, 상기 신호는 또한 디지털 기준 전위(V_{dngd})를 갖는다. 연산 증폭기(B300)의 양의 입력부(B302)는 입력 신호(B311)에 연결되며, 상기 입력 신호는 기준 전압(V_{ref})의 전압값을 갖는다. 연산 증폭기(B300)의 출력부(B303)는 전압값(V_{unten})을 갖는 출력 신호(B390)에 연결된다.

이는, 그 값(R_1 , R_2 , R_3 및 R_4)을 갖는 저항들(B150, B160, B170 및 B180)의 크기가, 신호(B211, B311)에 인가되는 고정된 기준 전압(V_{ref})과 관련해서 이하와 같이 설정됨으로써 도달된다:

수학식 2

$$V_{ref} = (V_{agnd} + V_{diff}) * R_2 / (R_1 + R_2)$$

수학식 3

$$V_{ref} = (V_{agnd} - V_{diff}) * R_4 / (R_3 + R_4)$$

수학식 4

$$V_{diff} = ((V_{2max} - V_{1min}) * R_f / R_{in}) - V_{agnd}$$

이 경우 신호(B121)에서 최대로 허용되는 V_2 의 전압값은 V_{2max} 로, 신호(B111)에서 최소로 허용되는 V_1 의 전압값은 V_{1min} 으로 표현된다. 기준 전압원은 외부로부터 제공될 수 있지만, 또는 내부에서 구현된 밴드갭(온도를 보상하며 작동 전압과 무관한 기준 전압)을 통해서 구현될 수 있다. 수학식 4에서, 최대로 허용되는 차이(V_{diff})는 최대 양의 편차(V_{2max}) 및 이에 속하는 최소 음의 편차(V_{1min})로부터 결정되며 즉, ($V_{2max} - V_{1min}$)는 서로 비교되어야 하는 중복 아날로그 신호들 서로간의 최대 허용 전압 편차이다. 2개의 신호들(B290 또는 B390)에서의 전압값들 중 하나가(V_{oben} 또는 V_{unten}) 양이면, 허용되어야 하는 것보다 더 큰 아날로그 신호의 편차가 제공된다. 상기 아

날로그 신호들을 제공하는 프로세서가 동기화되는 한, 저장되어야 하고 경우에 따라서는 출력 신호의 스위치오프를 유도하는 하나의 에러가 제공된다. 동기화는, 상응하는 처리 유닛들의 컨트롤 레지스터 내에서 예컨대 레디-신호가 활성화될 때, 또는 관련 아날로그 신호의 특정 상태 및 이로써 식별의 의미로 비교될 값을 신호화하는 UVE에 특정 디지털 신호가 전송될 때 주어진다. 에러를 저장하는 회로가 도14에 도시된다. 상기 회로에서 2개의 입력 신호들(B390, B290)이 NOR-회로(후속하는 인버터를 갖는 로직 OR-회로)(B410)에 의해 출력 신호(B411)에 연결된다. 상기 신호(B411)는 추가의 NOR-요소(B420) 내의 입력 신호(B421)에 연결되어 출력 신호(B421)를 생성한다. 상기 신호(B421)는 하나의 OR-회로(B430) 내에서 신호(B401)에 연결되어 신호(B431)를 생성하며, 상기 신호(B431)는 메모리 요소(D-플립-플롭)(B400)를 위한 입력 신호로서 사용된다. 상기 요소(B400)의 출력 신호(B401)는 값 1로써 에러를 나타낸다. D-플립-플롭(B400)은 신호들(B390 또는 B290)에서 2개의 전압값들(V_{unten} 또는 V_{oben}) 중 하나가 양이고 즉, 디지털 신호가 값을 높게 가지고 신호(B421)가 활성이 아니며 리셋-신호(B402)가 인가되지 않을 때, 클록(B403)을 이용하여 1을 저장한다. 에러는, 신호 리셋이 적어도 한번 활성화되었을 때까지는 저장되어 있다. 도11 내지 도13의 회로들의 설계 시, 저항들이 서로 매칭되는 것이 관찰되는데 즉, R_f 및 R_{in} , R_1 및 R_2 , R_3 및 R_4 의 저항 비율은 가급적 제조 공차와 무관하게 일정하다. 회로가 활성화되어야 하는지 또는, 비교가 실행되지 않아도 되는 처리 유닛들의 동기화가 발생하는지가 신호(B421)로써 제어될 수 있다. 신호(B402)는 이전의 에러를 리세팅하며 이로써 새로운 비교를 가능하게 한다.

[0075]

도15는 ADC를 도시한다. 상기 ADC는 예컨대 변환 속도, 정확도, 해상도, 안정성, 선형도 및 주파수 스펙트럼과 같은 요구 조건들에 따라, 공지된 다양한 변환 방법들에 의해 구현될 수 있다. 따라서 예컨대 연속적인 근사법의 원리가 선택될 수 있으며, 여기서 아날로그 신호는 디지털 투 아날로그 변환기(DAC)로부터 생성된 신호와, 비교기에 의해서 비교되고, 이 경우 DAC의 디지털 입력 비트는 MSB(최대 유효 비트)로부터 LSB(최소 유효 비트)까지 시스템적으로 임의로 높게 세팅된 다음, DAC의 아날로그 출력 신호가 아날로그 입력 신호(변환될 신호)보다 더 높은 값을 가질 때 다시 리세팅된다. DAC는 LSB로부터 MSB까지의 그 디지털 비트로써, 다음 번으로 높은 비트의 세팅이 항상 아날로그값에 대해 이전의 세팅보다 두 배가 더 크게 작용하는 방식으로, 웨이팅 1, 2, 4, 8, 16, ...을 갖는 저항들 또는 커패시턴스를 제어한다. 모든 비트들이 임의로 세팅되고 경우에 따라 다시 리세팅된 후, 디지털 워드의 값은 아날로그 입력 신호의 디지털 표현에 상응한다. 더 높은 속도 요구에 대해, 연속적인 데이터 흐름의 경우, 연속적으로 아날로그 신호를 처리하며, 직렬 비트열을 통해 상기 아날로그 데이터 흐름에 근접하는 직렬 디지털 신호를 송출하는 변환기도 사용될 수 있다. 여기서 디지털 워드는 시프트 레지스터 내에 저장된 비트열을 통해서 표현된다. 그러나 상기와 같은 변환기는 변환 시간 내에 아날로그 신호의 일정한 변경을 전제 조건으로 하는데, 이는 상기 변환기가 일정한 값을 처리할 수 없기 때문이다. 더 낮은 속도 요구에 대해, 계수 원리에 따른 변환기도 사용될 수 있으며, 이는 예컨대 입력 전압 또는 입력 전류를 이용하여 하나의 적분기에 연결된 커패시터의 상응하게 일정한 충전 또는 방전을 일으킨다. 이를 위해 필요한 시간이 측정되어, 기준 전압원 또는 상응하는 기준 전류를 이용하여 동일한 커패시터(적분기)의 충전 또는 방전을 위해 반대 방향으로 요구되는 시간에 대한 비율로 세팅된다. 시간 유닛은 클록으로 측정되며 필요한 클록들의 수는 아날로그 입력값에 대한 수치이다. 상기와 같은 방법은 예컨대 듀얼-슬로프 방법이며, 여기서 제1 측면(슬로프)은 방전을 통해서 아날로그값에 상응하게 결정되고, 제2 측면은 재충전을 통해서 기준값에 상응하게 결정된다(<http://www.exstom.com/journal/adc/dsadc.html> 참조). 도15의 ADC(B600)는 일반적으로 프로세서의 출력 신호인 트리거 신호(B602)에 의해 제어되며, 상기 출력 신호는 아날로그 신호를 제공하고 선택적으로는, 복수의 아날로그 신호들을 구분할 수 있기 위해 바로 제공되는 아날로그 신호의 유형에 대한 정보를 주는 식별부(B603)를 제공한다. 트리거 신호(B602)에 의해, 메모리 영역(B640) 내에서 변환된 아날로그 워드는 디지털값으로서 레지스터(B610) 내로 전달되며 선택적으로는, B620 내에 저장된 식별부(B603) 및 경우에 따라서, 메모리(B630) 내에 저장된 추가의 신호(B604)(이는 아날로그값의 식별을 위해 1이다)와 함께이다. 메모리 영역(B640)은, 복수의 값들이 저장되어야 하고 먼저 저장된 값이 또한 다시 먼저 송출되어야 할 때, 바람직하게는 FIFO(선입, 선출)로서도 구현될 수 있다. 상기 메모리 영역(B640)이 디지털값 뿐만 아니라 디지털화된 아날로그값을 위해서도 사용되면, 바람직하게 모든 디지털값들은, A=1인 디지털화된 아날로그값(B630)으로부터의 구분을 위해(도16, 도17 참조), B630에 상응하게 MSB-지점에서 비트 A=0으로 보충된다. B602 및 B603은 프로세서 i의 디지털 출력 데이터(O_i)의 성분이다. 도16에는 메모리 영역에 저장된 바와 같이, 디지털화되어 저장된 아날로그값의 부분들이 별도로 도시된다. 여기서 B710은 디지털화된 아날로그값 자체이며, B720은 이에 속하는 식별부이고, B730은 이 경우 1로서 저장된 아날로그 비트이다. 도17에는 동일한 메모리 영역에 저장된 디지털값의 변형예가 제공된다. B810에는 디지털값 자체가 저장되고, B820에는, 디지털값이 주로 비교되어야 하는지 또는 비교를 위한 추가의 조건들이 포함될 수 있는지에 대한 정보를 제공하는 식별부가 선택적으로 저장된다. B830에는 디지

털값에 대한 것인지를 식별하기 위해 값 0이 저장된다.

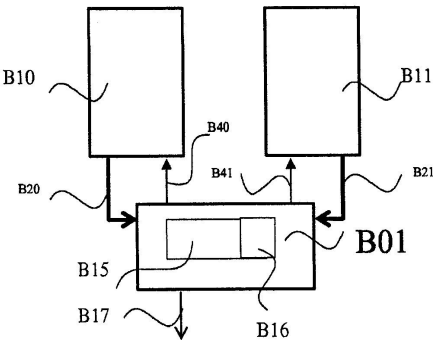
- [0076] 일시 저장된 디지털 및 아날로그 신호들을 비교하기 위해, 저장의 순서와, 경우에 따라서는 A-비트(B730 또는 B830)와, 식별부(B720 또는 B820)는 변환된 디지털값(B710) 또는 디지털값(B810)과 연관되어 검사된다. 또한 상이한 비트폭으로 인해, 아날로그 및 디지털 신호들을 분리된 메모리(2개의 FIFO) 내에 배치시킬 수도 있다. 비교는 하나의 프로세서의 하나의 값이 UVE로 전달되어, 다른 해당 프로세서들이 상기 값을 이미 제공했는지가 검사될 때마다, 이벤트 제어식으로 실행된다. 그러한 경우가 아닌 한, 값은 상응하는 FIFO 또는 메모리 내에 저장되고, 이와 다른 경우 비교가 바로 실행되며, 이 경우 FIFO도 메모리로서 사용될 수 있다. 비교는 해당 FIFO가 비어 있지 않을 때마다 완료된다. 2개 이상의 해당 프로세서들 또는 비교 유닛들의 경우, 모든 신호들이 배분을 위해 허용될 수 있는지(페일 사일런트 특성) 또는 경우에 따라 단지 하나의 에러-신호를 통해 에러 상태가 신호화되는지가 보팅을 통해 검출될 수 있다.

도면의 간단한 설명

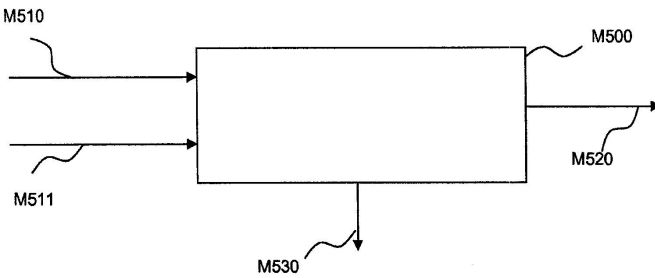
- [0022] 도1은 2개의 처리 유닛들을 위한 전환 및 비교 유닛의 기본 기능을 도시한 도면이다.
- [0023] 도1a는 비교기의 일반적인 도면이다.
- [0024] 도1c는 비교기의 확대도이다.
- [0025] 도1b는 전환 유닛 및 비교 유닛의 일반적인 도면이다.
- [0026] 도2는 2개의 처리 유닛들을 위한 전환 유닛 및 비교 유닛의 상세도이다.
- [0027] 도3은 2개의 처리 유닛들을 위한 전환 유닛 및 비교 유닛의 가능한 구현을 도시한 도면이다.
- [0028] 도4는 2개 이상의 처리 유닛들을 위한 전환 유닛 및 비교 유닛의 더욱 상세한 도면이다.
- [0029] 도5는 2개 이상의 처리 유닛들을 위한 전환 유닛 및 비교 유닛의 가능한 구현을 도시한 도면이다.
- [0030] 도6은 컨트롤 레지스터의 가능한 구현을 도시한 도면이다.
- [0031] 도7은 중앙 집중 보팅을 위한 보팅 유닛을 도시한 도면이다.
- [0032] 도8은 분산 보팅을 위한 보팅 유닛을 도시한 도면이다.
- [0033] 도9는 동기화 요소를 도시한 도면이다.
- [0034] 도10은 핸드셰이크-인터페이스를 도시한 도면이다.
- [0035] 도11은 차동 증폭기를 도시한 도면이다.
- [0036] 도12는 양의 전압 차이에 대한 비교기를 도시한 도면이다.
- [0037] 도13은 음의 전압 차이에 대한 비교기를 도시한 도면이다.
- [0038] 도14는 에러를 저장하기 위한 회로를 도시한 도면이다.
- [0039] 도15는 출력 레지스터를 갖는 아날로그 대 디지털 변환기를 도시한 도면이다.
- [0040] 도16은 식별부 및 아날로그 비트를 갖는, 디지털 변환된 아날로그값을 도시한 도면이다.
- [0041] 도17은 디지털 비트를 갖는 디지털 워드로 디지털값을 도시한 도면이다.

도면

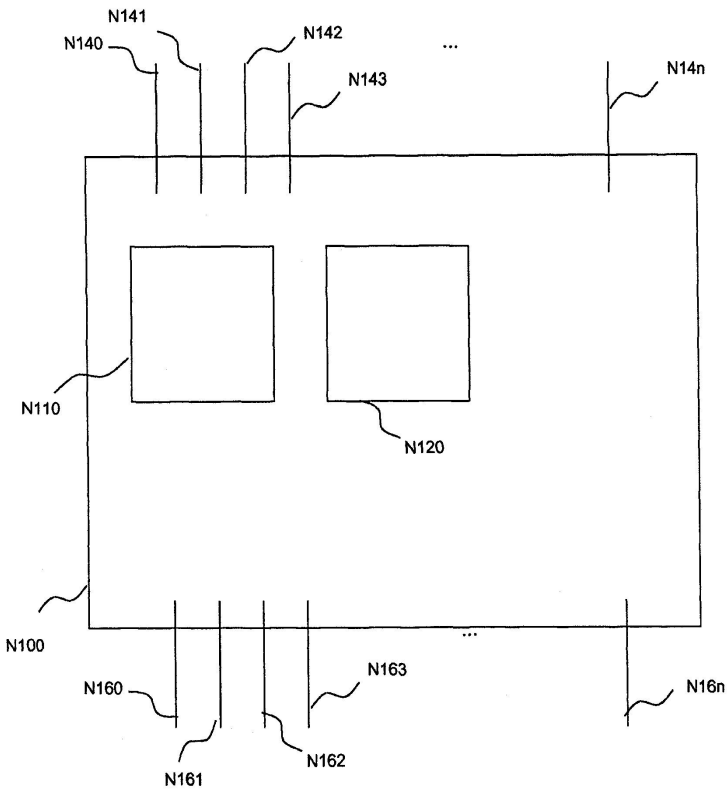
도면1



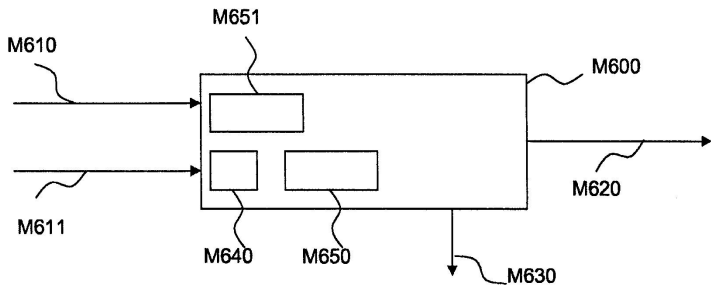
도면1a



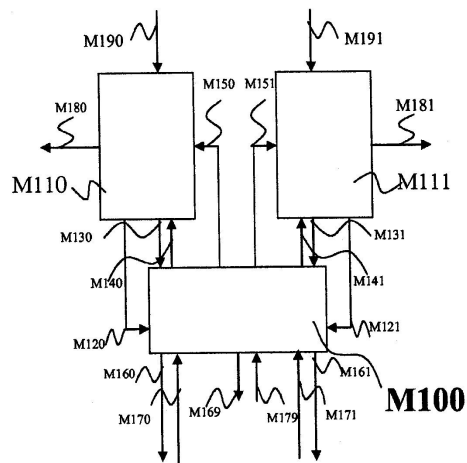
도면1b



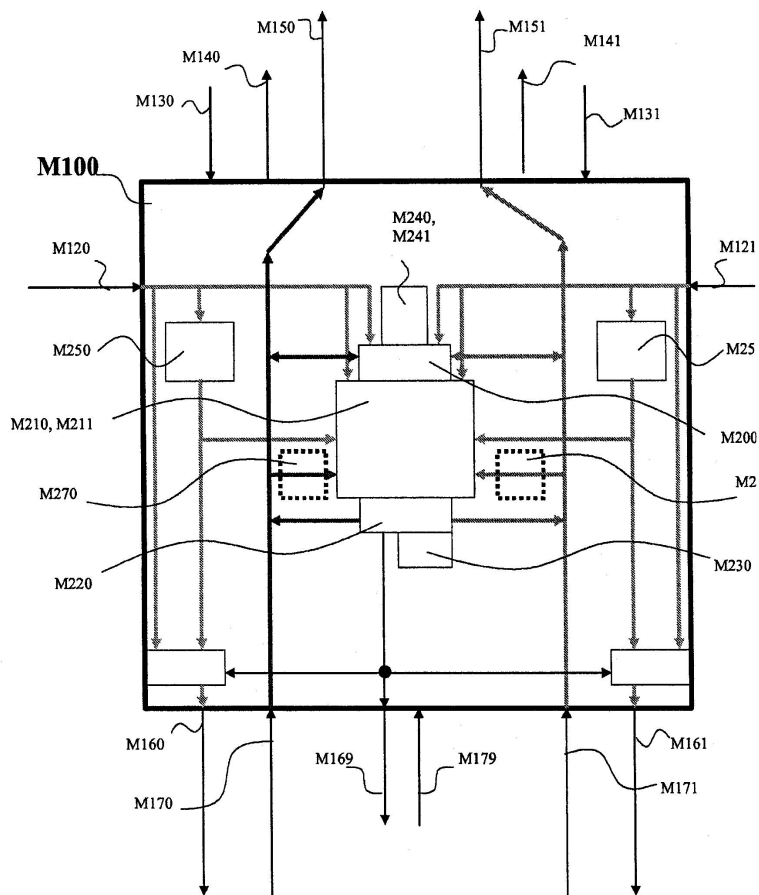
도면1c



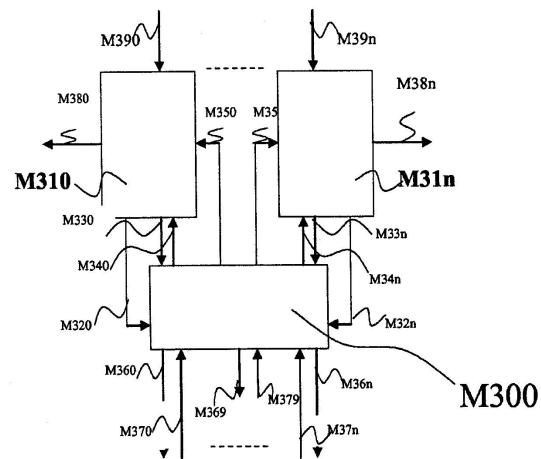
도면2



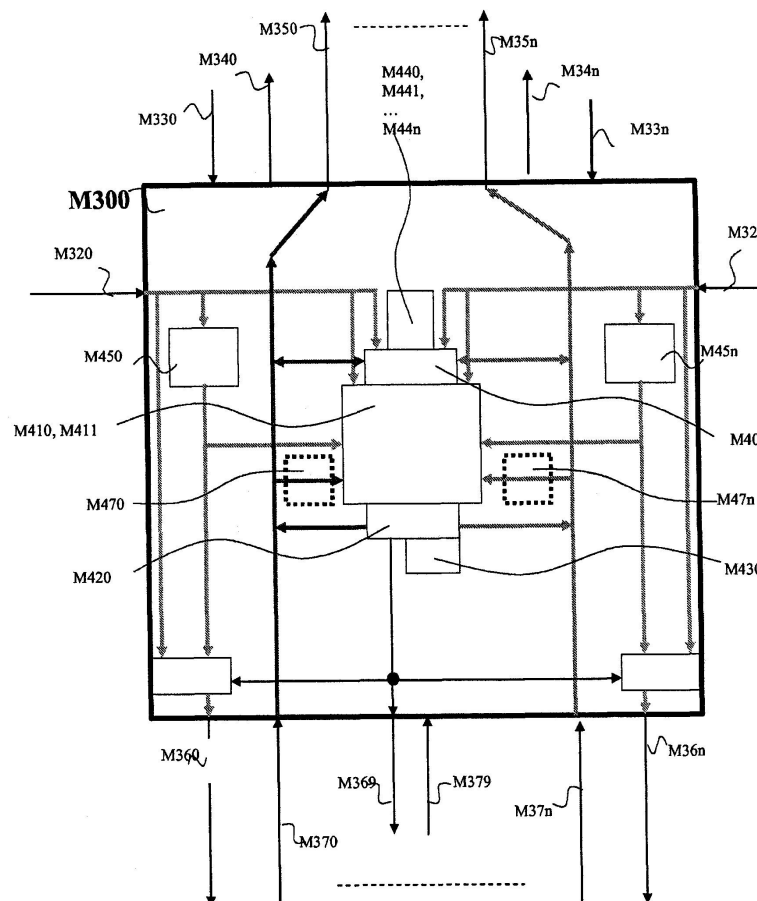
도면3



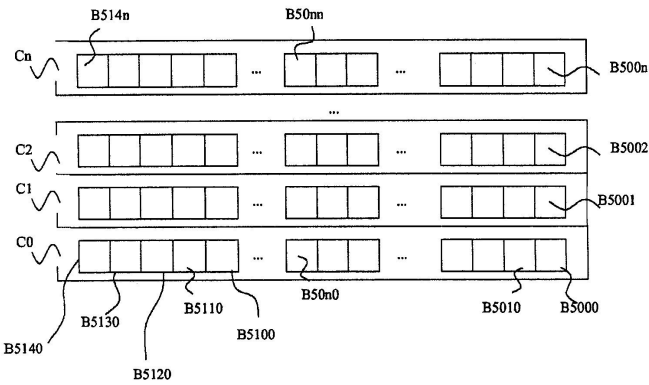
도면4



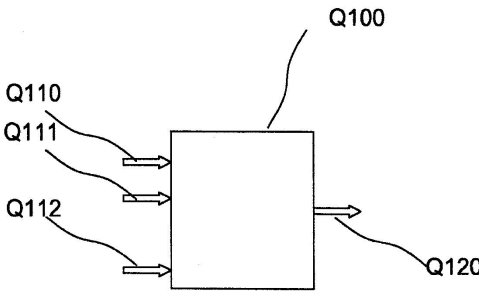
도면5



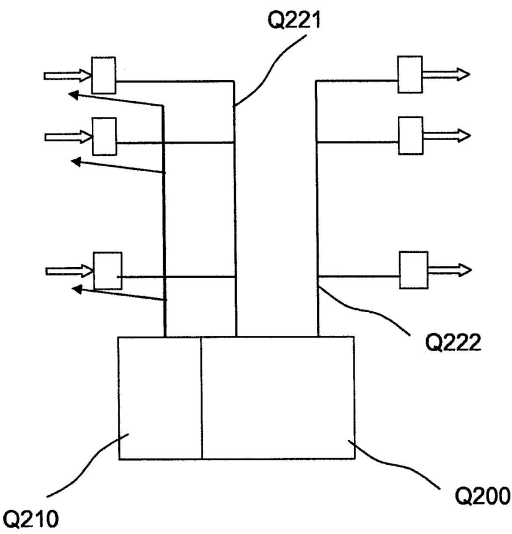
도면6



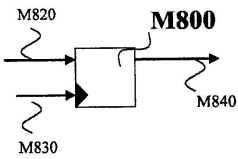
도면7



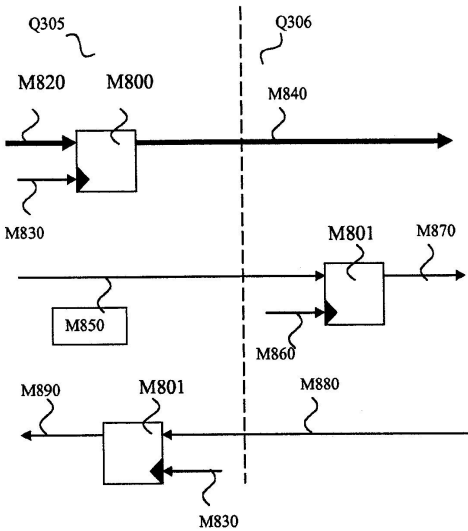
도면8



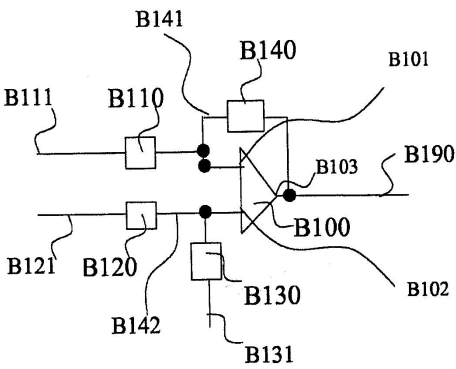
도면9



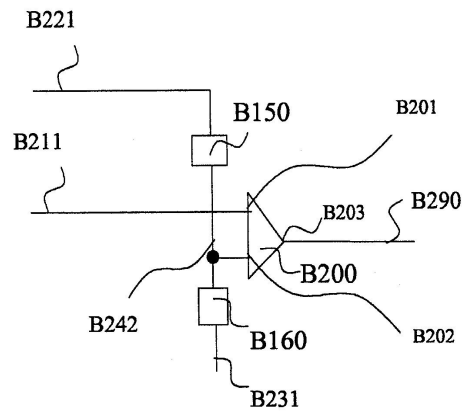
도면10



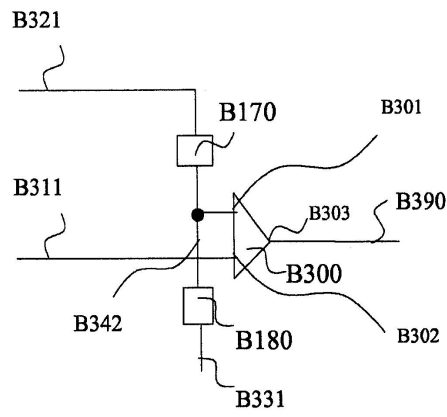
도면11



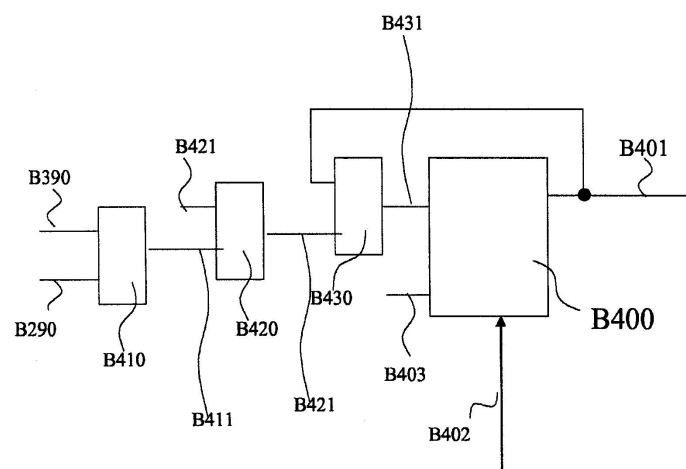
도면12



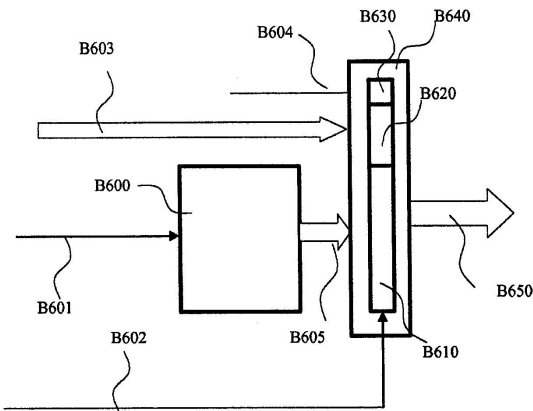
도면13



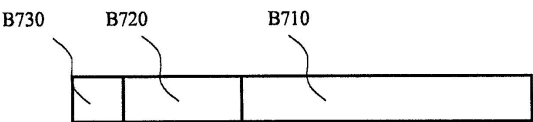
도면14



도면15



도면16



도면17

