



**(19) 대한민국특허청(KR)**  
**(12) 등록특허공보(B1)**

(45) 공고일자 2011년09월07일  
(11) 등록번호 10-1063720  
(24) 등록일자 2011년09월01일

- (51) Int. Cl.  
G06F 11/14 (2006.01) G06F 9/445 (2006.01)
- (21) 출원번호 10-2009-7014337
- (22) 출원일자(국제출원일자) 2008년04월11일  
심사청구일자 2009년07월28일
- (85) 번역문제출일자 2009년07월09일
- (65) 공개번호 10-2009-0101921
- (43) 공개일자 2009년09월29일
- (86) 국제출원번호 PCT/EP2008/054434
- (87) 국제공개번호 WO 2008/125634  
국제공개일자 2008년10월23일
- (30) 우선권주장  
11/735,295 2007년04월13일 미국(US)  
11/735,305 2007년04월13일 미국(US)
- (56) 선행기술조사문헌  
US06675258 B1\*  
US20060200707 A1\*  
\*는 심사관에 의하여 인용된 문헌

- (73) 특허권자  
인터내셔널 비지네스 머신즈 코퍼레이션  
미국 10504 뉴욕주 아몬크 뉴오차드 로드
- (72) 발명자  
그래비스, 제이슨 제임스  
미국 27560 노쓰 캐롤라이나주 모리스빌 웨일 하이 블러바드 4234  
모르톤, 데이비드 마이클  
미국 85747 아리조나주 투싼 이스트 카미노 도로 테아 12650  
(뒷면에 계속)
- (74) 대리인  
양영준, 장수길

전체 청구항 수 : 총 10 항

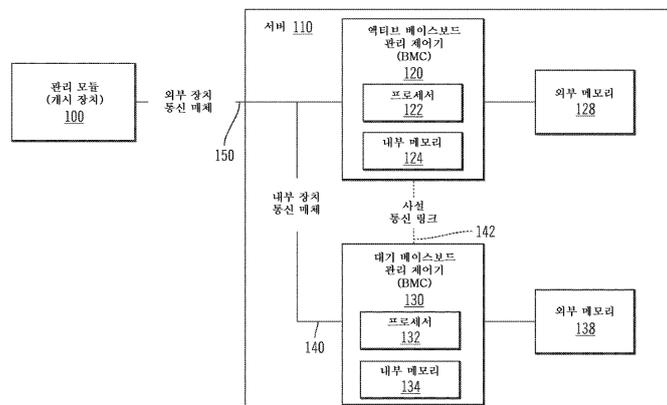
심사관 : 김종택

**(54) 피어 프로그램 가능 하드웨어 장치에 대한 자동화 펌웨어 복구**

**(57) 요약**

펌웨어를 복구하기 위한 방법 및 컴퓨터 프로그램 제품이 제공된다. 제1 프로그램 가능 하드웨어 장치는 제2 프로그램 가능 하드웨어 장치가 유효한 펌웨어 이미지를 필요로 한다는 것을 판정하고, 외부 메모리로부터 상기 유효한 펌웨어 이미지의 사본을 검색하고, 사설 통신 링크를 통해 상기 유효한 펌웨어 이미지를 상기 제2 프로그램 가능 장치로 전송하며, 상기 사설 통신 링크는 상기 제1 프로그램 가능 하드웨어 장치와 상기 제2 프로그램 가능 하드웨어 장치 간의 사설 통신을 가능하게 한다. 상기 제2 프로그램 가능 하드웨어 장치는 상기 유효한 펌웨어 이미지를 이용하여 기존의 펌웨어를 복구한다.

**대표도**



(72) 발명자

**응우옌, 누 타이**

미국 85713 아리조나주 투싼 웨스트 플라씨타 이스  
트렐라 아줄 900

**올긴, 이반 로날드 2세**

미국 85747-5938 아리조나주 투싼 이스트 하이마켓  
스트리트 10377

**엘스워스, 얼**

미국 85602 아리조나주 벤슨 사우스 매버릭 레인  
811

**홀더웨이, 케반**

미국 85747 아리조나주 투싼 사우스 카미노 세르페  
8182

**지, 루르드 매갈리**

미국 85757 아리조나주 투싼 웨스트 칼레 모헤다  
8395

**특허청구의 범위**

**청구항 1**

펌웨어(firmware)를 복구하는 컴퓨터 구현 방법(computer-implemented method)으로서,  
 서버 내의 제1 프로그램 가능 하드웨어 장치(programmable hardware device)의 제어 하에,  
 상기 서버 내의 제2 프로그램 가능 하드웨어 장치가 유효한 펌웨어 이미지를 필요로 하는지를 주기적으로 판정하는 단계; 및  
 상기 제2 프로그램 가능 하드웨어 장치가 상기 유효한 펌웨어 이미지를 필요로 한다고 판정한 것에 응답하여,  
 상기 제2 프로그램 가능 하드웨어 장치를 위한 상기 유효한 펌웨어 이미지의 사본을 검색(retrieving)하고,  
 상기 제1 프로그램 가능 하드웨어 장치와 상기 제2 프로그램 가능 하드웨어 장치 사이의 사설 통신(private communication)을 가능하게 하는 사설 통신 링크(private communication link)를 통해 상기 유효한 펌웨어 이미지를 상기 제2 프로그램 가능 하드웨어 장치로 송신하는 단계; 및  
 상기 제2 프로그램 가능 하드웨어 장치의 제어 하에,  
 상기 제1 프로그램 가능 하드웨어 장치로부터 상기 유효한 펌웨어 이미지를 수신한 것에 응답하여, 상기 유효한 펌웨어 이미지를 이용하여 기존 펌웨어를 복구하는 단계;  
 상기 제1 프로그램 가능 하드웨어 장치가 유효한 펌웨어 이미지를 필요로 하는지를 주기적으로 판정하는 단계; 및  
 상기 제1 프로그램 가능 하드웨어 장치가 상기 유효한 펌웨어 이미지를 필요로 한다고 판정한 것에 응답하여,  
 상기 제1 프로그램 가능 하드웨어 장치를 위한 상기 유효한 펌웨어 이미지의 사본을 검색하고,  
 상기 사설 통신 링크를 통해 상기 유효한 펌웨어 이미지를 상기 제1 프로그램 가능 하드웨어 장치로 송신하는 단계

를 포함하는 펌웨어를 복구하는 컴퓨터 구현 방법.

**청구항 2**

제1항에 있어서,  
 i) 상기 제1 프로그램 가능 하드웨어 장치는 상기 제1 프로그램 가능 하드웨어 장치에 전용되는(dedicated) 외부 메모리로부터 상기 유효한 펌웨어 이미지를 검색함;  
 ii) 상기 제1 프로그램 가능 하드웨어 장치 및 상기 제2 프로그램 가능 하드웨어 장치는 상기 유효한 펌웨어 이미지를 저장하는 외부 메모리를 공유함;  
 iii) 상기 제2 프로그램 가능 하드웨어 장치는 상기 제2 프로그램 가능 하드웨어 장치에 전용되는 외부 메모리로부터 상기 유효한 펌웨어 이미지를 검색함; 또는  
 iv) 외부 메모리는 상기 유효한 펌웨어 이미지의 복수 개의 버전을 저장함

중 하나 이상인, 펌웨어를 복구하는 컴퓨터 구현 방법.

**청구항 3**

제1항에 있어서,  
 i) 상기 제1 프로그램 가능 하드웨어 장치 및 상기 제2 프로그램 가능 하드웨어 장치는 중복되고(redundant) 동일한 펌웨어를 사용함, 또는  
 ii) 상기 제1 프로그램 가능 하드웨어 장치 및 상기 제2 프로그램 가능 하드웨어 장치는 중복되지 않고 상이한 펌웨어를 사용함

중 하나이거나 양자 모두인, 펌웨어를 복구하는 컴퓨터 구현 방법.

**청구항 4**

제1항에 있어서,

상기 제2 프로그램 가능 하드웨어 장치가 유효한 펌웨어 이미지를 필요로 하는지를 판정하는 단계는,

심박 기능(heartbeat function)을 사용하여 상기 제2 프로그램 가능 하드웨어 장치가 기능하고 있는지를 판정하도록 심박 기능 메시지(heartbeat function message)를 상기 제2 프로그램 가능 하드웨어 장치로 주기적으로 송신하는 단계

를 더 포함하고,

상기 제2 프로그램 가능 하드웨어 장치가 소정의 시간 내에 상기 제1 프로그램 가능 하드웨어 장치로부터의 상기 심박 기능 메시지에 응답하지 않는 경우, 상기 제1 프로그램 가능 하드웨어 장치는 상기 제2 프로그램 가능 하드웨어 장치가 상기 유효한 펌웨어 이미지를 필요로 한다고 판정하는, 펌웨어를 복구하는 컴퓨터 구현 방법.

**청구항 5**

제1항에 있어서,

상기 제2 프로그램 가능 하드웨어 장치가 유효한 펌웨어 이미지를 필요로 하는지를 판정하는 단계는,

상기 유효한 펌웨어 이미지를 요청하는 상기 제2 프로그램 가능 하드웨어 장치로부터의 지시(indication)를 수신하는 단계

를 더 포함하고,

상기 제2 프로그램 가능 하드웨어 장치는 상기 유효한 펌웨어 이미지의 버전을 특정하는, 펌웨어를 복구하는 컴퓨터 구현 방법.

**청구항 6**

펌웨어를 복구하는 컴퓨터 구현 방법으로서,

서버 내의 제1 프로그램 가능 하드웨어 장치의 제어 하에,

상기 서버 내의 제2 프로그램 가능 하드웨어 장치가 유효한 펌웨어 이미지를 필요로 한다고 판정한 것에 응답하여,

제1 메모리로부터 상기 유효한 펌웨어 이미지의 사본을 검색하고,

상기 유효한 펌웨어 이미지로 상기 제2 프로그램 가능 하드웨어 장치의 제2 메모리를 직접 업데이트하는 단계; 및

상기 제2 프로그램 가능 하드웨어 장치의 제어 하에,

상기 제2 메모리가 업데이트된 것에 응답하여, 상기 유효한 펌웨어 이미지를 이용하여 기존 펌웨어를 복구하는 단계; 및

상기 제1 프로그램 가능 하드웨어 장치가 유효한 펌웨어 이미지를 필요로 한다고 판정한 것에 응답하여,

상기 제2 메모리로부터 상기 유효한 펌웨어 이미지의 사본을 검색하고,

상기 유효한 펌웨어 이미지로 상기 제1 프로그램 가능 하드웨어 장치의 상기 제1 메모리를 직접 업데이트하는 단계

를 포함하는 펌웨어를 복구하는 컴퓨터 구현 방법.

**청구항 7**

컴퓨터 판독가능 프로그램을 기록한 컴퓨터 판독가능 저장 매체로서, 상기 컴퓨터 판독가능 프로그램은, 서버의 프로세서에 의해 실행시,

상기 서버 내의 제1 프로그램 가능 하드웨어 장치의 제어 하에,

상기 서버 내의 제2 프로그램 가능 하드웨어 장치가 유효한 펌웨어 이미지를 필요로 하는지를 주기적으로 판정하는 단계; 및

상기 제2 프로그램 가능 하드웨어 장치가 상기 유효한 펌웨어 이미지를 필요로 한다고 판정한 것에 응답하여,

상기 제2 프로그램 가능 하드웨어 장치를 위한 상기 유효한 펌웨어 이미지의 사본을 검색하고,

상기 제1 프로그램 가능 하드웨어 장치와 상기 제2 프로그램 가능 하드웨어 장치 사이의 사설 통신을 가능하게 하는 사설 통신 링크를 통해 상기 유효한 펌웨어 이미지를 상기 제2 프로그램 가능 하드웨어 장치로 송신하는 단계; 및

상기 제2 프로그램 가능 하드웨어 장치의 제어 하에,

상기 제1 프로그램 가능 하드웨어 장치로부터 상기 유효한 펌웨어 이미지를 수신한 것에 응답하여, 상기 유효한 펌웨어 이미지를 이용하여 기존 펌웨어를 복구하는 단계;

상기 제1 프로그램 가능 하드웨어 장치가 유효한 펌웨어 이미지를 필요로 하는지를 주기적으로 판정하는 단계; 및

상기 제1 프로그램 가능 하드웨어 장치가 상기 유효한 펌웨어 이미지를 필요로 한다고 판정한 것에 응답하여,

상기 제1 프로그램 가능 하드웨어 장치를 위한 상기 유효한 펌웨어 이미지의 사본을 검색하고,

상기 사설 통신 링크를 통해 상기 유효한 펌웨어 이미지를 상기 제1 프로그램 가능 하드웨어 장치로 송신하는 단계

를 포함하는 동작을 수행하게 하는, 컴퓨터 판독가능 저장 매체.

#### 청구항 8

제7항에 있어서,

i) 상기 제1 프로그램 가능 하드웨어 장치는 상기 제1 프로그램 가능 하드웨어 장치에 전용되는 외부 메모리로부터 상기 유효한 펌웨어 이미지를 검색함;

ii) 상기 제1 프로그램 가능 하드웨어 장치 및 상기 제2 프로그램 가능 하드웨어 장치는 상기 유효한 펌웨어 이미지를 저장하는 외부 메모리를 공유함;

iii) 상기 제2 프로그램 가능 하드웨어 장치는 상기 제2 프로그램 가능 하드웨어 장치에 전용되는 외부 메모리로부터 상기 유효한 펌웨어 이미지를 검색함; 또는

iv) 외부 메모리는 상기 유효한 펌웨어 이미지의 복수 개의 버전을 저장함

중 하나 이상인, 컴퓨터 판독가능 저장 매체.

#### 청구항 9

제7항에 있어서,

i) 상기 제1 프로그램 가능 하드웨어 장치 및 상기 제2 프로그램 가능 하드웨어 장치는 중복되고 동일한 펌웨어를 사용함; 또는

ii) 상기 제1 프로그램 가능 하드웨어 장치 및 상기 제2 프로그램 가능 하드웨어 장치는 중복되지 않고 상이한 펌웨어를 사용함

중 하나이거나 양자 모두인, 컴퓨터 판독가능 저장 매체.

#### 청구항 10

컴퓨터 판독가능 프로그램을 기록한 컴퓨터 판독가능 저장 매체로서, 상기 컴퓨터 판독가능 프로그램은, 서버의 프로세서에 의해 실행시,

상기 서버 내의 제1 프로그램 가능 하드웨어 장치의 제어 하에,

상기 서버 내의 제2 프로그램 가능 하드웨어 장치가 유효한 펌웨어 이미지를 필요로 하는지를 판정한 것에 응답하여,

제1 메모리로부터 상기 유효한 펌웨어 이미지의 사본을 검색하고,

상기 유효한 펌웨어 이미지로 상기 제2 프로그램 가능 하드웨어 장치의 제2 메모리를 직접 업데이트하는 단계; 및

상기 제2 프로그램 가능 하드웨어 장치의 제어 하에,

상기 제2 메모리가 업데이트된 것에 응답하여, 상기 유효한 펌웨어 이미지를 이용하여 기존 펌웨어를 복구하는 단계; 및

상기 제1 프로그램 가능 하드웨어 장치가 유효한 펌웨어 이미지를 필요로 한다고 판정한 것에 응답하여,

상기 제2 메모리로부터 상기 유효한 펌웨어 이미지의 사본을 검색하고,

상기 유효한 펌웨어 이미지로 상기 제1 프로그램 가능 하드웨어 장치의 상기 제1 메모리를 직접 업데이트하는 단계

를 포함하는 동작을 수행하게 하는, 컴퓨터 판독가능 저장 매체.

## 명세서

### 기술분야

[0001] 본 발명의 실시예들은 피어 프로그램 가능 하드웨어 장치에 대한 자동화된 펌웨어 복구에 관련된 것이다.

### 배경기술

[0002] 프로그램 가능 하드웨어 장치(예컨대, 스토리지 서버의 SES(Small Computer System Interface(SCSI) Enclosure Services) 프로세서 또는 USB 장치를 위한 USB 컨트롤러)는 많은 상이한 유형의 시스템에서 찾아볼 수 있다. 일부의 경우, 프로그램 가능 하드웨어 장치의 목적은 신뢰성(reliability), 유효성(availability) 또는 실용성(serviceability)(RAS) 특징들을 제공하는 것이다. 그러나, 때때로 프로그램 가능 하드웨어 장치는 그 동작을 구동하고 있는 펌웨어에 대한 업데이트를 필요로 할 수 있다. 펌웨어는 (예컨대, PROM(Programmable Read-Only Memory)으로 삽입됨으로써) 장치의 상설 부분인 프로그래밍으로 기술될 수 있다. 또한, 펌웨어이미지가 프로그램 가능 하드웨어 장치상에 로딩되는 펌웨어를 포함하는 데이터 세트로 기술될 수 있다는 점에서, 펌웨어는 프로그램 가능 하드웨어 장치상에서 작동하는 프로그래밍으로 기술될 수 있다. 많은 경우, 프로그램 가능 하드웨어 장치에 기록된 펌웨어는 이전의 구동 펌웨어를 오버라이트할(overwrite) 것이다. 따라서, 오류가 있는 (corrupt) 펌웨어(즉, 펌웨어 이미지 형태의 것)가 프로그램 가능 하드웨어 장치에 기록된다면, 프로그램 가능 하드웨어 장치는 동작하지 않을 것이고, 따라서 더 이상 정상적인 기능을 제공하지 못할 것이다. 오류가 있는 펌웨어(즉, 오류가 있는 펌웨어 이미지)를 갖는 프로그램 가능 하드웨어 장치는 오류가 있는 프로그램 가능 하드웨어 장치로 지칭될 수 있다. 오류가 있는 펌웨어는 오류 펌웨어(corrupt firmware) 또는 부정 펌웨어 (invalid firmware)로 지칭될 수 있다.

[0003] 때때로 발생하는 다른 조건은, 펌웨어가 정상적인 동작 중에 오류에 빠지고(예컨대, 펌웨어 이미지 다운로드가 일어나지 않을 때 또는 런타임 중) 그 오류가 펌웨어 이미지에 오류를 만들며, 따라서 프로그램 가능 하드웨어 장치가 정상적인 기능을 제공하지 못하게 하는 것이다.

[0004] 일반적으로, 몇몇 방식으로 장애를 일으키는 시스템 장치들은 시스템의 전체 성능에 부정적인 영향을 끼치며, 이는 대부분의 고객 환경에 있어서 허용될 수 없다. 일반적으로, 문제를 해결하기 위한 통상적인 수단은 프로그램 가능 하드웨어 장치를 교체하거나, 가능하다면 펌웨어 이미지를 재설치(reinstall)하는 것이다. 그러나, 이러한 해결법은 어떤 유형의 외부 지원의 개입을 필요로 하며, 이러한 유형의 개입은 자동적이지 않다. 고객이 어떤 지원에 의해 부정적인 영향을 받는 긴급한 동작 중이었다면, 지원을 요청하여 기다리는 기존의 해법은 적절하지 않을 것이다.

[0005] 따라서, 프로그램 가능 하드웨어 장치에 대한 펌웨어 장애의 자동적인 자가 복구를 가능하게 하는 기술이 필요하다.

**발명의 상세한 설명**

- [0006] 펌웨어를 복구하기 위한 방법 및 컴퓨터 프로그램 제품이 제공된다.
- [0007] 제1 프로그램 가능 하드웨어 장치는 제2 프로그램 가능 하드웨어 장치가 유효한 펌웨어 이미지를 필요로 한다는 것을 판정하고, 외부 메모리로부터 상기 유효한 펌웨어 이미지의 사본을 검색하고, 사설 통신 링크를 통해 상기 유효한 펌웨어 이미지를 상기 제2 프로그램 가능 장치로 전송하며, 상기 사설 통신 링크는 상기 제1 프로그램 가능 하드웨어 장치와 상기 제2 프로그램 가능 하드웨어 장치 간의 사설 통신을 가능하게 한다. 상기 제2 프로그램 가능 하드웨어 장치는 상기 유효한 펌웨어 이미지를 이용하여 기존의 펌웨어를 복구한다.

**실시예**

- [0015] 이하의 설명에서, 본 명세서의 일부이며 본 발명의 여러 실시예들을 도시하는 첨부 도면들에 대한 참조가 이루어진다. 기타의 실시예도 이용될 수 있으며 본 발명의 범주를 벗어나지 않고 구조 및 동작상의 변경이 이루어질 수 있다.
- [0016] 실시예들은 중복 파트너(redundant partner)를 갖거나, 또 다른 프로세서와의 공유 외부 메모리를 갖거나, 또는 프로그램 가능 하드웨어 장치에 대한 유효 펌웨어 이미지의 사본을 유지하는 또 다른 프로세서에 연결되는 프로그램 가능 하드웨어 장치에 대한 자동 펌웨어 복구를 제공한다. 특정 실시예에 있어서, 두 개의 중복 장치가 있고, 장애가 있는 장치가 업데이트를 수신할 수 있으며 두 장치 간에 통신 인터페이스가 작동하고 있다면, 장치들 중 하나의 임의의 펌웨어 업데이트가 다른 장치에 의해 교정될 수 있다. 펌웨어 업데이트는 펌웨어 이미지의 업데이트로 기술될 수 있다. 특정 실시예에 있어서, 중복되지 않는 두 개의 프로세서가 있으나, 오류가 있는 프로세서는 다른 프로세서로의 작동 통신 인터페이스를 가지고 있고, 이 다른 프로세서는 오류가 있는 프로세서의 펌웨어 이미지의 사본을 보유하거나 오류가 있는 프로세서의 펌웨어 이미지의 사본을 어떤 스토리지에 유지하는(예컨대, 보존 목적을 위해 펌웨어 업데이트 동안 펌웨어 이미지가 프로세서로 전송됨), 오류가 있는 프로세서의 외부 메모리에 대한 액세스를 가진다. 실시예들은 또한 프로세서가 정상적인 동작 중 펌웨어 이미지를 오류에 빠뜨리는 어떤 유형의 오류를 만나고 프로세서가 정상적인 동작 중 또는 리부팅 시에 장애를 감지하는, 펌웨어 업데이트와는 다른 경우에도 적용가능하다.
- [0017] 이해를 돕기 위해, 펌웨어 업데이트에 적용가능한 실시예들에 대한 설명이 제공된다. 그러나, 실시예들은 펌웨어 이미지가 정상적인 동작 중에 오류에 빠지는 경우에도 적용가능하다.
- [0018] 도 1은 본 발명의 특정 실시예에 따른 두 개의 중복된 장치들을 상세히 도시한 도면이다. 도 1에서, 관리 모듈(Management Module; MM, 100)은 개시 장치(initiating device)이고 외부 장치 통신 매체(150)를 통해 서버(110)에 연결된다. 서버(110)는 액티브(active) 베이스보드 관리 제어기(Baseboard Management Controller; BMC, 120) 및 대기(standby) 베이스보드 관리 제어기(130)를 포함한다. 액티브 및 대기 BMC(120, 130)는 로컬 자원들에 관리 능력을 제공하고 공유 자원들에게 중복된 관리 능력을 제공한다.
- [0019] 액티브 BMC(120)는 프로세서(122) 및 내부 메모리(124)를 포함한다. 대기 BMC는 프로세서(132) 및 내부 메모리(134)를 포함한다.
- [0020] 도 1에서, 액티브 및 대기 BMC(120, 130)는 중복되는 프로그램 가능 하드웨어 장치들이다. 액티브 BMC(120) 및 대기 BMC(130)는 각각 제1 및 제2 타겟 장치로 지칭되거나, 또는 파트너 BMC라고 지칭될 수 있다. 액티브 및 대기 BMC(120, 130)는 전체적인 서버(110) 신뢰성을 향상시키는 중복(redundancy) 목적을 위한 듀얼 BMC로 설명될 수 있지만, 중복에 의해 보호되지 않는 각 BMC(120, 130)이 개별적으로 제어하는 로컬 구성요소들이 있다(예컨대, 하나의 BMC(120, 130)는 파트너 BMC(120, 130)에 의해 직접 제어되는 구성요소들에 전력을 공급할 수 없다). BMC(120, 130)는 이들 사이의 내부 사설 통신 링크(private communication link; 142) 및 각 BMC(120, 130)에 전용되는 외부 메모리(128, 138)를 갖는다. 이 구성요소들(142, 128, 138)을 이용하여 실시예들은 자동 펌웨어 수정을 가능하게 한다.
- [0021] 액티브 BMC(120) 및 대기 BMC(130)는 사설 통신 링크(142)를 통해 통신을 할 수 있다. 사설 통신 링크(142)는 액티브 및 대기 BMC(120, 130) 간의 사설 통신을 가능하게 하는 사설 장치 통신 매체로 설명될 수 있다. 관리 모듈(100)은 사설 통신 링크(142) 상에서 직접 통신할 수 없다. 내부 장치 통신 매체(140)는 외부 장치 통신 매체(150)에 결합되어 있다. 특정 실시예에서, 외부 장치 통신 매체(150)는 버스(예컨대, RS485 시리얼 버스 인터-집적 회로(inter-integrated circuit; I2C) 버스, 듀얼 포트 RAM(DPRAM), 또는 기타 버스 기반 매체)일 수 있고, 내부 장치 통신 매체(140)는 외부 장치 통신 매체(150)에 연결된 버스(예컨대, 내부 RS485 시리얼 버

스 인터-집적 회로(inter-integrated circuit; I2C) 버스, 듀얼 포트 RAM(DPRAM), 또는 기타 버스 기반 매체) 일 수 있다.

[0022] 정상 동작에서, 관리 모듈(100)은 단일 외부 장치 통신 매체(150)를 통해 펌웨어 이미지를 받함으로써 (복수의 패킷을 통해) 단일 펌웨어 이미지를 액티브 및 대기 BMC(120, 130)로 송신한다. 정상 업데이트 프로세스 중인 각 BMC(120, 103)는 프로세서(122, 132)를 구동하는 BMC 명령어 세트를 저장하는 내부 메모리(124, 134)(예컨대, 내부 플래시 영역)에 펌웨어 이미지를 직접 기입한다. 펌웨어 이미지가 오류가 있는(corrupt) 경우, 각 BMC(120, 130)는 펌웨어 업데이트 프로세스에서 업데이트되지 않은 코드(즉, 부트 블록(boot block)으로 불리는 곳에 저장되는 코드)로 제공되는 감소된 기능(functionality) 세트를 통해 새로운 펌웨어 업데이트를 기다리는 제한된 성능을 갖는다. 그러나 BMC(120, 130)의 정상 기능은 펌웨어 이미지가 오류가 있는 경우 더 이상 제공될 수 없다. 서버(110)에 있어서, 각 BMC(120, 130)가 서버(110)의 시스템 전력을 제어하기 때문에 이것은 문제가 될 수 있다. 따라서, 만약 서버(110)가 펌웨어 업데이트에 대해 파워다운되어 업데이트가 실패하면(즉, 펌웨어 이미지가 오류가 있으면), 서버(110)는 유효 펌웨어 업데이트가 제공될 때까지 전원을 켤 수 없고, 이는 사용자가 블레이드 서버(blade server)의 사용성을 잃게 되는 것을 의미한다. "유효(valid)" 펌웨어 업데이트는 오류가 없고 BMC(120, 130)로 하여금 자신의 기능을 정확히 수행하도록 하는 것으로 설명될 수 있다. 오류가 있는 펌웨어의 결과의 한 예로, 펌웨어 업데이트가 성공적으로 완료되고 BMC(120, 130)가 정상적으로 시작되지만 정상 기능의 일부가 새로운 오류가 있는 펌웨어에 의해 제한되는 경우가 있다. 제2 BMC(120, 130)가 있는 경우에도, 펌웨어 실패의 경우, 이 제2 BMC(120, 130)는 파트너 BMC(120, 130)에 의해 직접 제어되는 구성요소들에 전력을 공급할 수 없는데, 이는 동작 상태에 있는 두 BMC(120, 130)의 중요성을 강조한다.

[0023] 실시예들에서, 액티브 BMC(120)는 액티브 BMC(120)가 펌웨어 이미지의 사본을 저장하는 외부 메모리(128)에 결합된다. 대기 BMC(130)는 대기 BMC(130)가 펌웨어 이미지의 사본을 저장하는 외부 메모리(138)에 결합된다. 중복된 장치들의 경우, 각 중복된 장치는 동일한 펌웨어를 실행시킨다. 따라서, 외부 메모리(128, 138)의 펌웨어 이미지들의 사본들은 동일하다. 이러한 사본들에서, BMC들(120, 130) 중 하나가 오류가 있는 펌웨어를 갖는다면, 다른 BMC(120, 130)가 외부 메모리(128, 138)로부터 대응하는 펌웨어 이미지를 제공할 수 있다.

[0024] 특정 실시예에서, 서버(110)는 (International Business Machines Corporation으로부터 얻을 수 있는) IBM® BladeCenter®

새시(chassis)의 블레이드 서버이고, 이 블레이드 서버는 듀얼 베이스보드 관리 컨트롤러들(BMCs)을 갖는다. 블레이드 서버는 중형컴퓨터(midrange) 서버 클래스 저장 시스템으로 설명될 수 있다. 그러나, 실시예들은 임의의 타겟 장치 세트(예컨대, 액티브 및 대기 BMC(120, 130)와 같은 중복된 장치들)에 적용될 수 있고, 데이터 스니핑(sniffing) 모드에서 "스누핑(snooping)"을 허용하는 타겟 장치들 간에 임의의 공유된 통신 매체(예컨대, 내부 장치 통신 매체(140))를 사용할 수 있고, 중복된 타겟 장치들 간의 임의의 사설 장치 통신 매체(예컨대, 사설 통신 링크(142))를 사용할 수 있다. ("프로미스큐어스(promiscuous)" 모드로도 불리는) 데이터 스니핑 모드는 타겟 장치가, 통신이 타겟 장치로 지정되는지 여부에 관계 없이, 온전히 도착하는 각 통신(예컨대, 네트워크 패킷)을 인터셉트하고 판독하는 모드로 설명될 수 있다. 실시예들은 시리얼 또는 논-시리얼인 네트워크들에 사용될 수 있다. 본 명세서의 예시들이 펌웨어 업데이트를 언급하고 있지만, 실시예들은 소프트웨어 업데이트에도 적용될 수 있다. 또한, 동일한 장치 통신 매체를 통해 업데이트를 수신하는 임의의 수의 장치들이 있을 수 있다. 액티브 및 대기 BMC들(120, 130)은 IBM®

BladeCenter®

관리 모듈(Management Module; MM)에 의해 사용되는 블레이드 서버 내의 제어 객체들로 설명될 수 있다.

[0025] 정상 동작 동안, 하나의 BMC(예컨대, 액티브 BMC(120))는 외부 장치 통신 매체(150)를 "소유(own)"하는 것으로 되고, 따라서 한번에 하나의 BMC가 관리 모듈(100)과 통신할 수 있다. 관리 모듈(100)은 액티브 BMC(120)로 명령들을 송신하는 것을 알고 있지만, 관리 모듈(100)은 액티브 BMC(120)로 직접 통신을 하지 않는다. 대신, 관리 모듈(100)은 메시지들을 서버(110) 슬롯과 연관된 외부 장치 통신 매체(150) 상의 주소로 메시지들을 송신하고, 액티브 및 대기 BMC들(120, 130)이 이 주소에 응답 및/또는 대기(listening)할 수 있다. 따라서, 관리 모듈(100)로부터 봤을 때, 관리 모듈은 어느 시점에서든 하나의 BMC로 통신을 하고 있다. 서버(110) 내의 듀얼 BMC들(120, 130)의 특정 실시예들에서, 두 BMC들이 능동적으로 외부 장치 통신 매체(150)를 사용하지 못하게 하는 하드웨어 억제기(inhibitor)가 존재하지 않는다(여기서 대기 BMC(130)는 내부 장치 통신 매체(140)를 통해

외부 장치 통신 매체(150)에 액세스할 수 있다). 그러나, 특정 실시예에서, 외부 장치 통신 매체(150)는 액티브 BMC(120)로 정의되는 디폴트 BMC에 의해 능동적으로 사용되고, 다른 BMC(130)는 외부 장치 통신 매체(150)를 능동적으로 사용하고 있는 액티브 BMC(120)가 실패할 때까지 내부 장치 통신 매체(140)와 함께 비활성 상태로 남는다.

[0026] 도 2는 본 발명의 특정 실시예에 따른 두 개의 비-중복된(non-redundant) 프로그램 가능 하드웨어 장치들을 자세히 설명한 도면이다. 도 2에서, 관리 모듈(MM, 200)은 개시 장치이고 외부 장치 통신 매체(250)를 통해 서버(210)에 연결된다. 서버(210)는 프로그램 가능 하드웨어 장치 A(220) 및 프로그램 가능 하드웨어 장치 B(230)를 포함한다. 프로그램 가능 하드웨어 장치 A(220)는 프로세서(222) 및 내부 메모리(224)를 포함한다. 프로그램 가능 하드웨어 장치 B는 프로세서(232) 및 내부 메모리(234)를 포함한다.

[0027] 도 2에서, 프로그램 가능 하드웨어 장치들(220, 230)은 또한 파트너라고 지칭될 수 있다. 프로그램 가능 하드웨어 장치들(220, 230)은 그들 간에 내부 사실 통신 링크(242)를 갖고, 각 프로그램 가능 하드웨어 장치(220, 230)는 외부 메모리(228, 238)를 갖는다. 점선(290)은 프로그램 가능 하드웨어 장치 A(220)가 선택적으로 외부 메모리(238)에 결합되는 것을 나타내는 반면, 점선(292)은 프로그램 가능 하드웨어 장치 B(230)가 선택적으로 외부 메모리(228)에 결합되는 것을 나타낸다. 이러한 방식으로 외부 메모리(228, 238) 중 하나 혹은 두 외부 메모리(228, 238) 모두가 공유 외부 메모리로서 기능할 수 있다. 특정 실시예에서, 외부 메모리(228, 238)는 하나의 프로그램 가능 하드웨어 장치(220, 230)에 전용되고, 다른 프로그램 가능 하드웨어 장치(220, 230)에 의해 액세스될 수 있다. 도 2에 도시된 실시예에서, 프로그램 가능 하드웨어 장치들(220, 230)은 동일한 펌웨어를 사용하지 않지만, 각각은 (예컨대, 자신의 고유 외부 메모리(228, 238)로부터 또는 다른 프로그램 가능 하드웨어 장치의 외부 메모리(228, 238)로부터의) 다른 프로그램 가능 하드웨어 장치(220, 230) 펌웨어 이미지의 사본에 액세스할 수 있다. 그러면, 하나의 프로그램 가능 하드웨어 장치(220, 230)가 오류가 있는 펌웨어로 인해 실패되면, 다른 프로그램 가능 하드웨어 장치(220, 230)가 유효한 펌웨어 이미지를 제공할 수 있다.

[0028] 실시예들은 외부 메모리(128, 138, 228, 238)를 인용했지만, 비휘발성이거나 (예컨대, 전원 오프 동안) 메모리 내의 펌웨어가 손실되지 않도록 콘텐츠를 보호하는 방식의 임의의 저장 공간이, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)에 대한 외부의 저장 공간으로 사용될 수 있다.

[0029] 도 2에서, 내부 장치 통신 매체(240)가 외부 장치 통신 매체(250)에 결합된다. 특정 실시예에서, 외부 장치 통신 매체(250)는 버스(예컨대, RS485 시리얼 버스 인터-집적 회로(inter-integrated circuit; I2C) 버스, 듀얼 포트 RAM(DPRAM), 또는 기타 버스 기반 매체)일 수 있고, 내부 장치 통신 매체(240)는 외부 장치 통신 매체(250)에 연결된 버스(예컨대, 내부 RS485 시리얼 버스 인터-집적 회로(inter-integrated circuit; I2C) 버스, 듀얼 포트 RAM(DPRAM), 또는 기타 버스 기반 매체)일 수 있다.

[0030] 실시예들에서, 각 프로그램 가능 하드웨어 장치(예컨대, BMC(120, 130) 또는 장치들(220, 230))는 이 프로그램 가능 하드웨어 장치에 연결된 새로운 하드웨어 구성요소, 즉 외부 메모리를 사용한다. 도 3은 특정 실시예에 따라 펌웨어 이미지를 수신하는 것에 응답하여 각 프로그램 가능 하드웨어 장치(120, 130, 220, 230)에 의해 수행되는 로직을 도시한다. 프로그램 가능 하드웨어 장치(120, 130, 220, 230)가 펌웨어 이미지(즉, 수신된 제1 펌웨어 이미지이거나 이전에 수신된 펌웨어 이미지에 대한 업데이트일 수 있는 새로운 펌웨어 이미지)를 수신하는 것으로 제어가 블록(300)에서 시작된다. 블록(302)에서, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)는 펌웨어 이미지를 외부 메모리에 저장한다. 각 외부 메모리(128, 138, 228, 238)는 펌웨어 이미지의 복수의 사본들(예컨대, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)가 펌웨어 이미지의 특정 버전을 얻는 것을 가능하게 하는 복수의 버전들)을 보유할 정도로 충분히 크다. 특정 실시예에서, 사본의 수는 메모리가 몇 개의 사본을 지원하느냐에 의해 결정되는 반면, 다른 실시예에서는 메모리 크기에 부가하여 또는 그 대신에 다른 인자들이 사용될 수 있다.

[0031] 특히, 정상 펌웨어 동작 동안, 각 BMC(120, 130)는 펌웨어 이미지를 외부 메모리(128, 138)의 영역에 복사한다. 각 BMC(120, 130)는 둘 이상의 사본을 보유한다. 하나 이상의 외부 메모리(228, 238)를 공유하는 비-중복된 프로그램 가능 하드웨어 장치들(220, 230)에 대해, 각 프로그램 가능 하드웨어 장치(220, 230)는 공유 외부 메모리(228, 238) 내에 펌웨어 이미지의 사본을 저장할 수 있다. 반면, 하나 이상의 외부 메모리(228, 238)를 공유하지 않는 비-중복된 프로그램 가능 하드웨어 장치들(220, 230)에 대하여, 각 프로그램 가능 하드웨어 장치(220, 230)는 다른 프로그램 가능 하드웨어 장치(220, 230)의 펌웨어 이미지의 사본을 수신하고 해당 사본을 자신의 외부 메모리(228, 238)에 저장한다.

[0032] 도 4는 특정 실시예에 따른, 장애를 일으키는 프로그램 가능 하드웨어 장치(120, 130, 220, 230)에 의해 수행되

는 로직을 도시한다. 프로그램 가능 하드웨어 장치(120, 130, 220, 230)가 유효 펌웨어 이미지가 필요하다고 선택적으로 판단하고 파트너에게 통지하는 것으로 제어가 블록(400)에서 시작한다. 특정 실시예에서, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)가 파트너에게 통지하기 보다는, 파트너가 자동으로 장애를 검출하고 유효 펌웨어 이미지를 송신한다. 블록(402)에서, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)가 유효 펌웨어 이미지의 사본을 수신한다. 특정 실시예에서, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)가 파트너로부터 사본을 수신한다. 특정한 대안적인 실시예에서, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)는 자신의 외부 메모리(128, 138, 228, 238)로부터 유효 펌웨어 이미지의 사본을 획득한다. 블록(406)에서, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)가 수신된 유효 펌웨어 이미지의 사본을 이용하여 복구된다. 블록(406)에서, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)는 (예컨대, 펌웨어 이미지가 자신의 외부 메모리(128, 138, 228, 238)로부터 검색되지 않으면) 유효 펌웨어 이미지의 사본을 자신의 외부 메모리(128, 138, 228, 238)에 선택적으로 저장한다.

[0033] 도 5는 특정 실시예에 따른, 파트너가 장애를 일으킨 경우 프로그램 가능 하드웨어 장치(120, 130, 220, 230)에 의해 수행되는 로직을 도시한다. 프로그램 가능 하드웨어 장치(120, 130, 220, 230)가, 파트너가 유효 펌웨어 이미지가 필요하다고 판단하는 것으로 제어가 블록(500)에서 시작된다. 판단은 프로그램 가능 하드웨어 장치(120, 130, 220, 230)에 의해 자동으로 이루어질 수 있거나 프로그램 가능 하드웨어 장치(120, 130, 220, 230)가 파트너로부터 장애의 표시를 수신할 수 있다. 블록(502)에서, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)는 외부 메모리(128, 138, 228, 238)로부터 유효 펌웨어 이미지의 사본을 검색한다. 다양한 실시예에서, 유효 펌웨어 이미지의 사본은 프로그램 가능 하드웨어 장치(120, 130, 220, 230)에 전용된 외부 메모리로부터 혹은 공유 외부 메모리로부터 검색될 수 있다. 블록(504)에서, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)는 유효 펌웨어 이미지의 사본을 사설 통신 링크(142, 242)를 통해 파트너로 송신한다.

[0034] 도 1 및 두 개의 중복된 장치가 있는 실시예와 관련하여, 오류가 있는 펌웨어 이미지가 관리 모듈(100)로부터 BMC(120, 130)로 전달되는 경우, 하나의 BMC(120, 130)가 한 번에 업데이트되기 때문에, 펌웨어 업데이트 프로세스는 본 실시예에 따라 수행되는 액션없이 실패될 수 있다. 특정 실시예에서, 오류가 있는 BMC(120, 130)는 업데이트하기 위한 펌웨어 이미지의 사본을 얻기 위해 자신의 외부 메모리(128, 138)에 직접 액세스한다. 특정 실시예에서, 만약 BMC(120, 130)가 외부 메모리(120, 130)에서 유효 펌웨어 이미지를 찾지 못하거나 오류가 있는 BMC(120, 130)가 파트너에게 표시(indication)를 제공하도록 설계된 경우, 오류가 있는 BMC(120, 130)는 유효 펌웨어 이미지가 필요하다는 표시를 파트너 BMC(120, 130)에게 제공한다. 파트너 BMC(120, 130)는 오류가 있는 BMC(120, 130)에게 자신의 외부 메모리(128, 138) 내의 최종 유효 펌웨어 이미지를 제공함으로써 또는 자신의 펌웨어 이미지를 직접 판독하고 이를 사설 통신 링크(142)를 통해 다른 BMC(120, 130)에 제공함으로써 오류가 있는 BMC(120, 130)의 펌웨어를 롤백(rollback)하는 위치에 있다. 동작하는 BMC(120, 130)는 펌웨어 업데이트 프로세스를 개시하고 오류가 있었던 파트너 BMC(120, 130)를 업데이트하는 관리 모듈로서 역할을 한다. BMC(120, 130)의 펌웨어 업데이트 개시의 제어는 심박 메커니즘(heartbeat mechanism)으로 유효 파트너의 도메인에서 이루어질 수 있다. 심박 메커니즘은 파트너가 계속 기능하고 있는지를 판단하기 위해 프로그램 가능 하드웨어 장치(120, 130, 220, 230)가 주기적으로 메시지를 파트너에게 송신하고 파트너로부터 메시지를 수신하는 것으로 설명될 수 있다. 예를 들어, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)가 소정의 시간 내에 심박 기능 메시지에 응답하지 않는 경우, 파트너 프로그램 가능 하드웨어 장치(120, 130, 220, 230)는 프로그램 가능 하드웨어 장치(120, 130, 220, 230)가 실패하였고 유효 펌웨어 이미지를 필요로 한다고 판단한다. 따라서, 완료와 동시에, 두 BMC는 중복된 방식으로 다시 동작할 수 있다.

[0035] 도 2 및 장치들이 중복이 아닌 실시예들을 참고하면, 공유 외부 메모리(228, 238)가 프로세서들(222, 232) 간에 이용가능할 수 있거나 두 프로세서(222, 232)가 별도의 전용 외부 메모리(228, 238)에 파트너의 펌웨어 이미지를 유지할 수 있다. 프로세서들(222, 232)은 중복될 필요는 없지만 기능 통신 인터페이스(예컨대, 사설 통신 링크(242))를 갖는다. 다양한 실시예에서, 각 프로세서(222, 232)는 정상 동작 동안 심박 기능을 수행하거나 오류가 있는 프로세서(222, 232)가 파트너 프로세서(222, 232)에게 실패를 표시한다. 펌웨어 업데이트에 대하여, 업데이트 중인 프로세서(222, 232)가 기능을 다루는 위치에 있지 않기 때문에(즉, 일반적으로 기능은 펌웨어 업데이트 동안 액세스될 수 없는 동작 코드로 되어 있기 때문에) 심박 메커니즘은 일시적으로 멈춘다. 그러나, 심박 메커니즘이 제때에 시작되지 않은 경우 만료되는 타임아웃(timeout)이 발생할 수 있다(즉, 타임아웃은 파트너 프로세서에 대한 펌웨어 업데이트에 대해 허용가능한 가장 긴 시간으로 정의될 수 있다). 타임아웃의 경우, 파트너 프로세서(222, 232)는 외부 메모리(228, 238)로부터 펌웨어 이미지를 자동으로 검색하여 오류가 있는 프로세서(222, 232)를 업데이트한다. 특정 실시예에서, 오류가 있는 프로세서(222, 232)는 파트너 프로세서(222, 232)로부터 이용가능한 펌웨어의 최종 버전의 펌웨어 이미지를 요청하고 수신한다. 펌웨어 이미지는

프로세서들(222, 232) 간의 사설 통신 링크(242)를 통해 제공될 수 있거나, 이와 달리, 만약 파트너 프로세서(222, 232)가 Joint Test Action Group (JTAG) 인터페이스와 같이 프로세서의 내부 메모리(224, 234)를 업데이트하는 유틸리티에 직접 기입하여 내부 메모리(224, 234)를 직접 업데이트하는 능력을 갖으면 오류가 있는 프로세서의 내부 메모리(224, 234)(예컨대, 내부 플래시 영역)가 직접 업데이트될 수 있다. 완료와 함께, 오류가 있는 프로세서는 유효 펌웨어 이미지를 갖는다.

[0036] 이러한 실시예들을 통해, 오류가 있는 펌웨어 이미지를 수신하는 프로그램 가능 하드웨어 장치(120, 130, 220, 230)는 자동으로 유효한 펌웨어 이미지로 복구된다. 실시예들에서, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)가 정상 동작 동안 자신의 펌웨어 또는 펌웨어 이미지를 손상시키는 오류를 발견하는 경우, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)는 자신의 외부 메모리(128, 138, 228, 238) 또는 파트너(120, 130, 220, 230)로부터의 유효 펌웨어 이미지를 이용하여 복구될 수 있다. 정상 동작 동안 펌웨어 이미지를 손상시킬 수 있는 조건의 예로, 펌웨어로 하여금 유효하지 않은 코드를 실행하도록 하거나 액세스 가비지(garbage) 데이터에 액세스하도록 하는 유효하지 않은 메모리 액세스가 있다. 이 경우, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)가 정상 동작 동안 자신이 오류가 있는 상태에 있다는 것을 식별하는 능력을 갖고 있으면, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)는 유효 펌웨어 이미지로 복구될 수 있다. 또한, 만약 프로그램 가능 하드웨어 장치(120, 130, 220, 230)가 타임아웃(예컨대, 감시 타이머(watchdog timer)의 타임아웃) 때문에 리부트가 되면, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)는 문제가 있다는 것을 식별하고, 프로그램 가능 하드웨어 장치(120, 130, 220, 230)는 유효 펌웨어 이미지를 얻기 위해 정상 부트 업(boot up)을 멈출 수 있다. 감시 타이머는 하드웨어에 의해 주기적으로 리셋되는 타이머로 설명될 수 있고, 만약 타이머가 리셋되지 않으면 시스템은 실패 상태로 된다.

[0037] 따라서, 이러한 실시예에 따라 오류가 있는 프로그램 가능 하드웨어 장치(120, 130, 220, 230)는 파트너 프로그램 가능 하드웨어 장치(120, 130, 220, 230)로부터 유효 펌웨어 이미지를 획득하고 유효 펌웨어 이미지로 동작할 수 있다.

[0038] 오류가 있는 펌웨어 이미지는 펌웨어 업데이트 동안 또는 정상 동작 동안 복구될 수 있다. 실시예들은 유효 펌웨어 이미지로 롤백되는 것을 가능하게 한다. 실시예들은 또한 심박 메커니즘을 제공하여 정상 동작 동안 파트너(120, 130, 220, 230)의 실패를 검출한다.

[0039] 추가 실시예에 대한 설명

[0040] 설명된 동작들은 소프트웨어, 펌웨어, 하드웨어 또는 이들의 임의의 조합을 제작하기 위한 표준의 프로그래밍 기술 및/또는 엔지니어링 기술을 사용하여 방법, 컴퓨터 프로그램 제품(computer program product) 또는 장치로서 구현될 수 있다.

[0041] 실시예들 각각은 전부 하드웨어인 실시예, 또는 전부 소프트웨어인 실시예, 또는 하드웨어 구성요소 및 소프트웨어 구성요소 양자를 모두 포함하는 실시예의 형태를 취할 수 있다. 실시예들은, 펌웨어, 상주 소프트웨어(resident software), 마이크로코드(microcode) 등을 포함하지만 이들로 한정되지 않는 소프트웨어로 구현될 수 있다.

[0042] 나아가, 실시예들은 컴퓨터 또는 임의의 명령 실행 시스템에 의해 사용되거나 그와 관련하여 사용되기 위한 프로그램 코드를 제공하는 컴퓨터 사용가능 매체(computer-usable medium) 또는 컴퓨터 판독가능 매체(computer-readable medium)로부터 액세스 가능한 컴퓨터 프로그램 제품의 형태를 취할 수 있다. 본 설명을 위하여, 컴퓨터 사용가능 매체 또는 컴퓨터 판독가능 매체는 명령 실행 시스템, 명령 실행 장치 또는 명령 실행 디바이스에 의해 사용되거나 그와 관련하여 사용되기 위한 프로그램을 포함할 수 있거나, 저장할 수 있거나, 전달할 수 있거나, 전파할 수 있거나, 수송할 수 있는 임의의 장치일 수 있다.

[0043] 설명된 동작들은 컴퓨터-사용가능 매체 또는 컴퓨터 판독가능 매체에 유지된 코드로서 구현될 수 있고, 프로세서는 컴퓨터 판독가능 매체로부터 그 코드를 판독하고 실행할 수 있다. 이러한 매체는 전자, 자기, 광학, 전자기, 적외선 또는 반도체 시스템(또는 장치나 디바이스)이거나 전파 매체(propagation medium)일 수 있다. 컴퓨터-판독가능 매체의 예에는, 반도체 또는 고체 상태 메모리(solid state memory), 자기 테이프, 이동식 컴퓨터 디스켓, 리지드 자기 디스크(rigid magnetic disk), 광 디스크, (예컨대, 하드 디스크 드라이브, 플로피 디스크, 테이프 등의) 자기 저장 매체, (예컨대, 랜덤 액세스 메모리(RAM), DRAM, SRAM, 판독 전용 메모리(ROM), PROM, EEPROM, 플래시 메모리, 펌웨어, 프로그램 가능 로직(programmable logic) 등의) 휘발성 및 비휘

발성 메모리 장치가 포함된다. 광 디스크의 예에는 현재 콤팩트 디스크-판독 전용 메모리(CD-ROM), 콤팩트 디스크-판독/기록(CD-R/W) 및 DVD가 포함된다.

- [0044] 또한, 설명된 동작들을 구현하는 코드는 (예컨대, 집적 회로 칩, 프로그램 가능 게이트 어레이(PGA: Programmable Gate Array), 주문형 반도체(ASIC) 등의) 하드웨어 로직으로 구현될 수 있다. 나아가, 설명된 동작들을 구현하는 코드는 "전송 신호(transmission signal)"로 구현될 수 있는데, 전송 신호는 광섬유, 구리선 등과 같은 전송 매체를 통하거나 공간을 통해 전파될 수 있다. 코드 또는 로직이 인코딩된 전송 신호는 또한 무선 신호, 위성 전송(satellite transmission), 전파(radio wave), 적외선 신호, 블루투스 등을 포함할 수 있다. 코드 또는 로직이 인코딩된 전송 신호는 송신국에 의해 전송되고 수신국에 의해 수신될 수 있으며, 전송 신호에 인코딩된 코드 또는 로직은 수신국 및 송신국이나 수신 및 송신 장치의 하드웨어 또는 컴퓨터 판독가능 매체에서 디코딩되고 저장될 수 있다.
- [0045] 컴퓨터 프로그램 제품은, 코드가 구현될 수 있는 전송 신호, 하드웨어 로직 및/또는 컴퓨터 사용가능 매체나 컴퓨터 판독가능 매체를 포함할 수 있다. 물론, 당업자라면, 실시예들의 범위로부터 벗어나지 않고 이러한 구성에 다양한 변경이 이루어질 수 있고, 컴퓨터 프로그램 제품은 기술 분야에 알려진 임의의 정보 보유 매체(information bearing medium)를 포함할 수 있음을 인식할 것이다.
- [0046] 로직이라는 용어는, 예로서, 소프트웨어, 펌웨어 및/또는 소프트웨어와 하드웨어의 조합을 포함할 수 있다.
- [0047] 특정 실시예들은, 컴퓨터-판독가능 코드를 컴퓨팅 시스템에 결합시키는 사람 또는 자동화된 프로세싱에 의해 컴퓨팅 하부구조를 배치하는 방법에 관한 것일 수 있는데, 컴퓨팅 시스템과 결합된 코드는 설명된 구현예들의 동작을 수행할 수 있다.
- [0048] 도 3, 4 및 5의 로직은 특정 순서로 일어나는 구체적인 동작들을 설명한다. 다른 실시예에서, 로직 동작들 중 일부가 상이한 순서로 수행되거나, 수정되거나, 제거될 수 있다. 또한, 지금까지 설명된 로직에 동작들이 부가될 수 있고, 여전히 설명된 실시예에 부합할 수 있다. 나아가, 본 명세서에 설명된 동작들은 순차적으로 일어날 수도 있고, 특정 동작들이 병렬로 처리될 수도 있으며, 또는 단일 프로세스에 의해 수행되는 것으로 설명된 동작들이 분산된 프로세스들에 의해 수행될 수도 있다.
- [0049] 도 3, 4 및 5에 도시된 로직은 소프트웨어, 하드웨어, 프로그램 가능 게이트 어레이 로직(programmable gate array logic), 프로그램 불가능 게이트 어레이 로직(non-programmable gate array logic), 또는 하드웨어, 소프트웨어나 게이트 어레이 로직의 조합으로 구현될 수 있다.
- [0050] 도 6은 특정 실시예에 따라 사용될 수 있는 시스템 아키텍처(600)를 도시한다. 클라이언트 컴퓨터(100) 및/또는 서버 컴퓨터(120)가 시스템 아키텍처(600)를 구현할 수 있다. 시스템 아키텍처(600)는 프로그램 코드를 저장 및/또는 실행하는데 적합하고, 시스템 버스(620)를 통해 메모리 구성요소(604)에 직접적 또는 간접적으로 결합된 적어도 하나의 프로세서(602)를 포함한다. 메모리 구성요소(604)는 프로그램 코드의 실제 실행 중에 이용되는 로컬 메모리, 대량 저장소(bulk storage), 그리고 실행 중에 코드를 대량 저장소로부터 검색하여야 하는 횟수를 감소시키기 위해 적어도 일부의 프로그램 코드에 대한 일시적인 저장소를 제공하는 캐시 메모리를 포함할 수 있다. 메모리 구성요소(604)는 운영 체제(605) 및 하나 이상의 컴퓨터 프로그램(606)을 포함한다. 또한, 메모리 구성요소(604)는 본 발명의 실시예들이 교시하는 설명된 동작들의 일부 또는 전부를 구현하는 코드(630)를 포함한다. 코드(630)가 도시되어 있지만, 본 발명의 실시예들이 교시하는 설명된 동작들은, 이와 달리 하드웨어로 구현되거나, 하드웨어와 소프트웨어의 조합으로 구현될 수 있다.
- [0051] (키보드, 디스플레이, 포인팅 장치 등을 포함하지만 이들로 한정되지 않는) 입출력(I/O) 장치(612, 614)는 시스템에 직접 결합되거나, 중개 I/O 컨트롤러(intervening I/O controller; 610)를 통해 결합될 수 있다.
- [0052] 또한, 데이터 처리 시스템이 중개 사설망(intervening private network) 또는 중개 공중망(intervening public network)을 통해 다른 데이터 처리 시스템 또는 원격 프린터나 원격 저장 장치에 연결되도록 하기 위해 네트워크 어댑터(608)가 시스템에 결합될 수 있다. 모뎀, 케이블 모뎀 및 이더넷 카드는 현재 이용가능한 유형의 네트워크 어댑터(608) 중 일부이다.
- [0053] 시스템 아키텍처(600)는 (예컨대, 자기 디스크 드라이브, 광 디스크 드라이브, 테이프 드라이브 등과 같은 비휘발성 저장 영역 등의) 저장소(616)에 결합될 수 있다. 저장소(616)는 내부 저장 장치(internal storage device)나 부착형 또는 네트워크 액세스 가능형 저장소(attached or network accessible storage)를 포함할 수 있다. 저장소(616)의 컴퓨터 프로그램(606)은 기술 분야에 알려진 방식에 따라 메모리 구성요소(604)로 로딩되

어 프로세서(602)에 의해 실행될 수 있다.

[0054] 시스템 아키텍처(600)는 도시된 것보다 적은 수의 컴포넌트를 포함하거나, 도시되지 않은 추가적인 컴포넌트를 포함하거나, 도시된 컴포넌트와 추가적인 컴포넌트들의 일부 조합을 포함할 수 있다. 시스템 아키텍처(600)는 메인프레임, 서버, 퍼스널 컴퓨터, 워크스테이션, 랩톱, 핸드헬드 컴퓨터, 전화 장치(telephony device), 네트워크 가전(network appliance), 가상화 장치(virtualization device), 저장소 컨트롤러 등과 같이 기술분야에 알려진 임의의 컴퓨팅 장치를 포함할 수 있다.

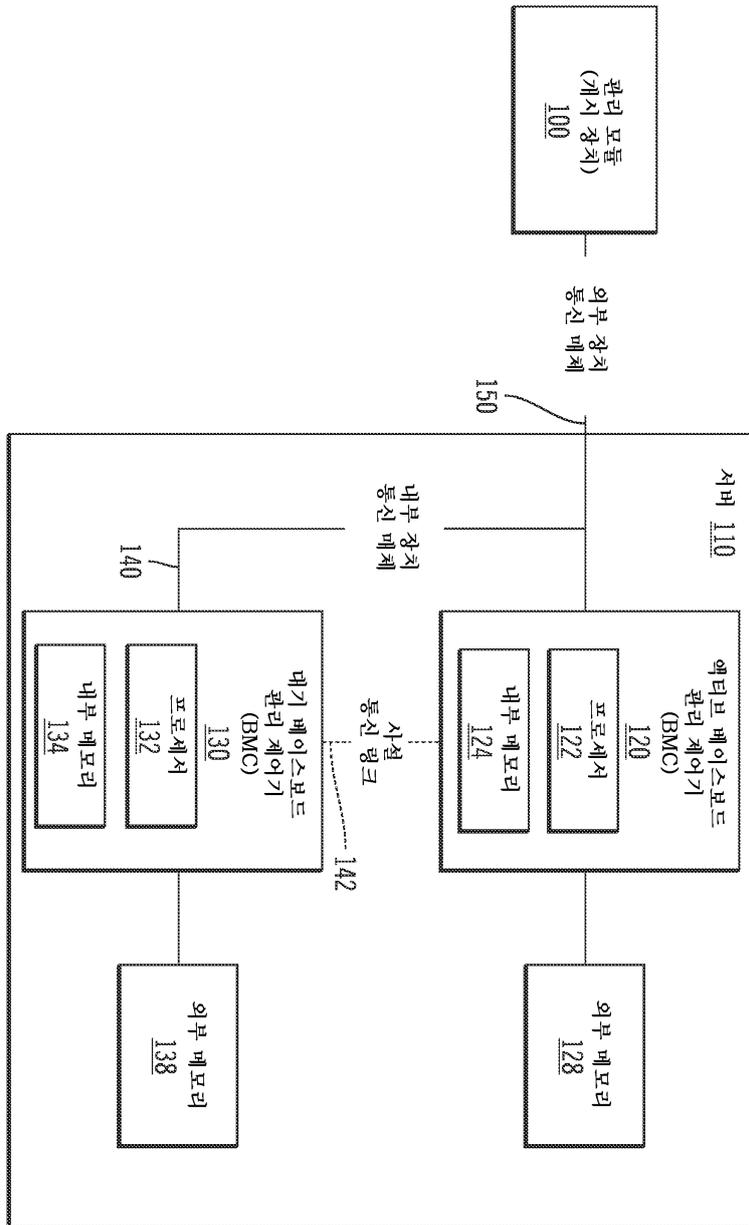
[0055] 본 발명의 실시예들에 대한 이상의 설명은 예시와 설명을 위해 제공된 것이다. 이는 제한적이거나, 실시예들을 정확히 개시된 형태로 한정하는 것으로 의도된 것이 아니다. 지금까지의 교시 내용에 비추어 다양한 수정 및 변경이 가능하다. 실시예들의 범위는 본 상세한 설명에 의해서가 아니라, 첨부된 특허청구의 범위에 의해 한정되는 것으로 의도된 것이다. 지금까지의 상세사항, 예 및 데이터는 실시예들의 구성의 제조 및 사용에 대한 완전한 설명을 제공한다. 실시예들의 기술적 사상 및 범위로부터 벗어나지 않고 다양한 실시예들이 이루어질 수 있으므로, 실시예들은 이하에 첨부되는 특허청구의 범위나 추후 제출될 임의의 특허청구의 범위 및 그 균등물에 속한다.

**도면의 간단한 설명**

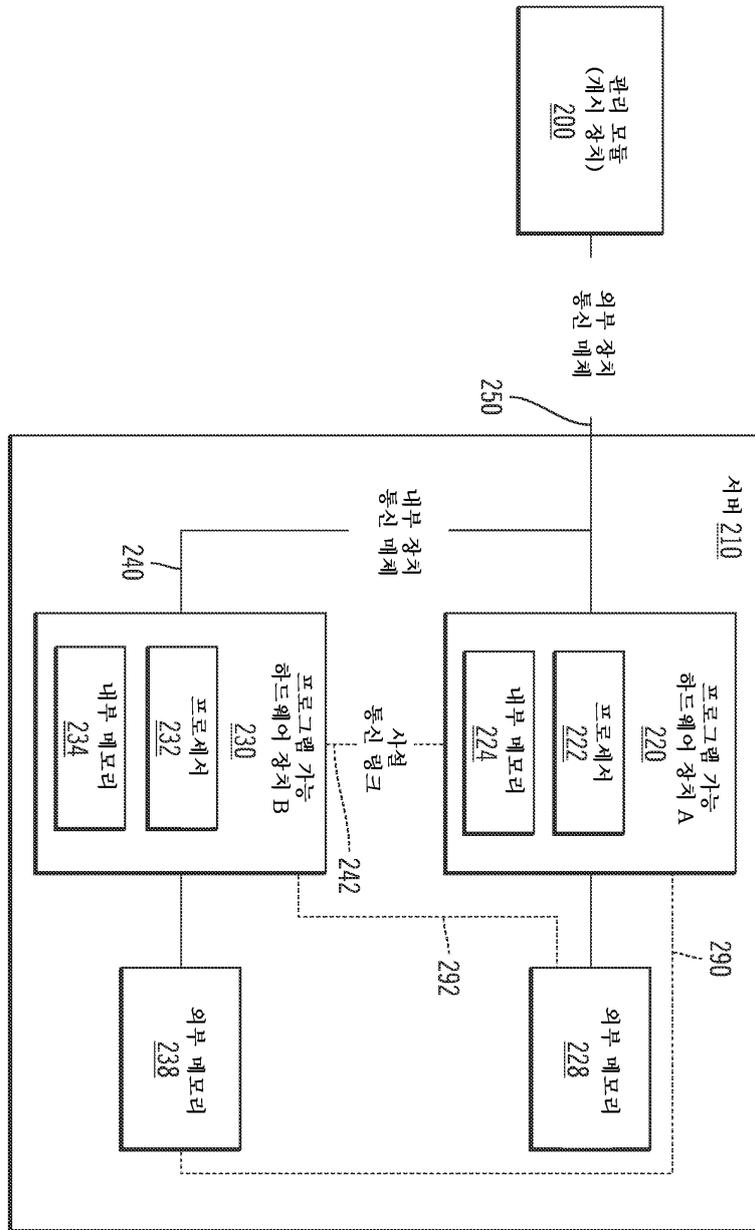
- [0008] 도면을 참조할 때, 유사한 참조부호는 도면 전체를 통해 대응되는 부분을 나타낸다.
- [0009] 도 1은 본 발명의 특정 실시예에 따른 두 개의 중복 장치를 상세히 도시한 도면.
- [0010] 도 2는 본 발명의 특정 실시예에 따른 중복되지 않은 두 개의 프로그램 가능 하드웨어 장치를 자세히 도시한 도면.
- [0011] 도 3은 특정 실시예에 따른, 펌웨어 이미지를 수신하는 것에 응답하여 각 프로그램 가능 하드웨어 장치에 의해 수행되는 로직을 도시하는 도면.
- [0012] 도 4는 특정 실시예에 따른, 장애를 일으키는 프로그램 가능 하드웨어 장치에 의해 수행되는 로직을 도시하는 도면.
- [0013] 도 5는 특정 실시예에 따른, 파트너가 실패한 경우 프로그램 가능 하드웨어 장치에 의해 수행되는 로직을 도시하는 도면.
- [0014] 도 6은 특정 실시예에 따라 사용될 수 있는 시스템 아키텍처를 도시하는 도면.

도면

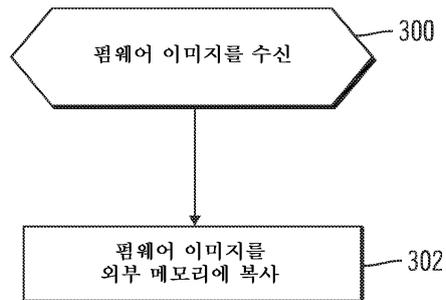
도면1



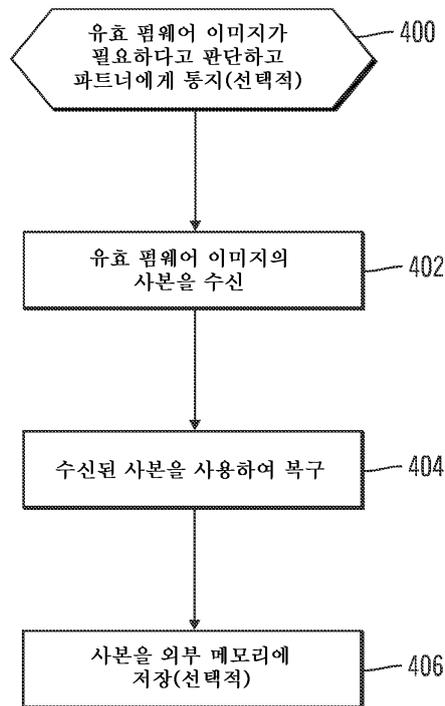
도면2



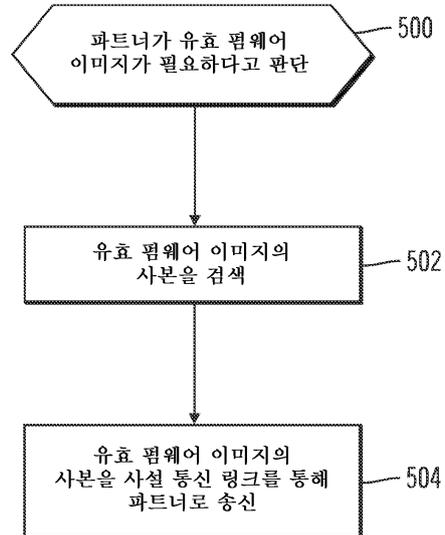
도면3



도면4



도면5



도면6

