



등록특허 10-2706938



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2024년09월12일  
(11) 등록번호 10-2706938  
(24) 등록일자 2024년09월10일

- (51) 국제특허분류(Int. Cl.)  
*G06F 9/30* (2018.01) *G06F 9/38* (2024.01)
- (52) CPC특허분류  
*G06F 9/30043* (2013.01)  
*G06F 9/30181* (2013.01)
- (21) 출원번호 10-2018-7017864
- (22) 출원일자(국제) 2016년12월09일  
심사청구일자 2021년11월25일
- (85) 번역문제출일자 2018년06월22일
- (65) 공개번호 10-2018-0099680
- (43) 공개일자 2018년09월05일
- (86) 국제출원번호 PCT/US2016/065740
- (87) 국제공개번호 WO 2017/116652  
국제공개일자 2017년07월06일
- (30) 우선권주장  
62/271,475 2015년12월28일 미국(US)  
15/252,323 2016년08월31일 미국(US)
- (56) 선행기술조사문헌  
KR1020140069245 A\*  
US05832202 A\*  
US08826273 B1\*

\*는 심사관에 의하여 인용된 문헌

전체 청구항 수 : 총 30 항

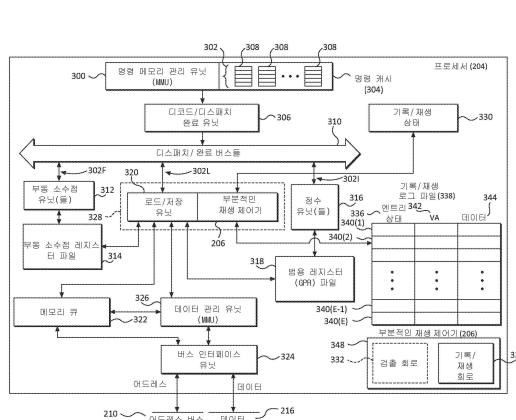
심사관 : 지정훈

(54) 발명의 명칭 **블록-원자 실행 모델을 사용하는 프로세서-기반 시스템에서 부분적으로 실행된 명령 블록들의 재생**

**(57) 요 약**

블록-원자 실행 모델을 사용하는 프로세서-기반 시스템에서 부분적으로 실행된 명령 블록들의 재생이 개시된다. 일 양상에서, 부분적인 재생 제어기가 CPU(central processing unit)의 프로세서(들)에 제공된다. 명령이 잠재적인 아키텍처 상태 수정과 연관된 명령 블록에서 검출되거나, 명령들의 실행 동안에 제외가 발생하면, 명령 블록(뒷면에 계속)

**대 표 도**



록이 재실행된다. 명령 블록의 재실행 동안에, 부분적인 재생 제어기는 로드/저장 명령들로부터 생성된 결과들을 기록하도록 구성된다. 따라서, 명령 블록의 재실행 동안에 제외가 발생하면, 제외가 발생하기 전에 실행된 로드/저장 명령들에 대해 이전에 기록된 생성된 결과들은, 제외가 해결된 후에 명령 블록의 재실행 동안에 재생된다. 따라서, 명령 블록에서 부작용 동작들로 이어지는 명령들의 실행은, 부작용을 반복하지 않고서, 이전에 생성된 결과들로 결정론적으로 반복될 수 있다.

(52) CPC특허분류

*G06F 9/3832* (2013.01)

*G06F 9/3861* (2013.01)

---

## 명세서

### 청구범위

#### 청구항 1

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기(partial replay controller)로서,

잠재적인 아키텍처 상태 수정(potential architectural state modification)과 연관된 명령의 검출 또는 상기 프로세서에서의 제외의 발생에 대한 응답으로, 명령 블록에 대해 기록/재생 상태를 활성 상태로 설정하도록 구성되는 검출 회로; 및

기록/재생 회로를 포함하고, 상기 기록/재생 회로는, 상기 기록/재생 상태가 상기 명령 블록에 대해 상기 활성 상태인 것에 대한 응답으로,

이전에 생성된 데이터가 다음 로드/저장 명령(next load/store instruction)에 대해 기록되는지를 결정하기 위해, 상기 명령 블록에서 실행될 상기 다음 로드/저장 명령에 대응하는 기록/재생 로그 파일 내의 엔트리 상태를 검사하고;

상기 이전에 생성된 데이터가 상기 다음 로드/저장 명령에 대해 상기 기록/재생 로그 파일에 기록되지 않는 것에 대한 응답으로, 실행되는 다음 로드/저장 명령의 생성된 데이터를 상기 기록/재생 로그 파일에 기록하고; 그리고

상기 이전에 생성된 데이터가 상기 다음 로드/저장 명령에 대해 상기 기록/재생 로그 파일에 기록되는 것에 대한 응답으로, 상기 다음 로드/저장 명령에 대해 상기 기록/재생 로그 파일에 기록된 상기 이전에 생성된 데이터를 사용하여 상기 다음 로드/저장 명령을 실행하도록 구성되는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

#### 청구항 2

제1 항에 있어서,

상기 검출 회로는 추가로, 상기 프로세서에 의한 상기 명령 블록의 실행 동안에, 상기 잠재적인 아키텍처 상태 수정과 연관된 상기 명령 블록 내의 명령을 검출하도록 구성되는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

#### 청구항 3

제2 항에 있어서,

상기 검출 회로는 부작용 검출 회로를 포함하고, 상기 부작용 검출 회로는:

상기 프로세서에 의한 상기 명령 블록의 실행 동안에, 잠재적인 부작용을 갖는 로드/저장 명령을 포함하는 상기 명령 블록 내의 명령을 검출하고; 그리고

상기 잠재적인 부작용을 갖는 상기 로드/저장 명령을 포함하는 명령의 검출에 대한 응답으로, 상기 명령 블록에 대해 상기 기록/재생 상태를 상기 활성 상태로 설정하도록 구성되는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

#### 청구항 4

제2 항에 있어서,

상기 검출 회로는 부작용 검출 회로를 포함하고, 상기 부작용 검출 회로는:

상기 프로세서에 의한 상기 명령 블록의 실행 동안에, 기록 시스템 구성 레지스터 명령을 포함하는 상기 명령 블록 내의 명령을 검출하고; 그리고

상기 기록 시스템 구성 레지스터 명령을 포함하는 명령의 검출에 대한 응답으로, 상기 명령 블록에 대해 상기 기록/재생 상태를 상기 활성 상태로 설정하도록 구성되는,  
프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

## 청구항 5

제2 항에 있어서,

상기 검출 회로는 부작용 검출 회로를 포함하고, 상기 부작용 검출 회로는:

상기 프로세서에 의한 상기 명령 블록의 실행 동안에, 예상되는 제외를 갖는 명령을 포함하는 상기 명령 블록 내의 명령을 검출하고; 그리고

상기 예상되는 제외를 갖는 명령을 포함하는 상기 명령의 검출에 대한 응답으로, 상기 명령 블록에 대해 상기 기록/재생 상태를 상기 활성 상태로 설정하도록 구성되는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

## 청구항 6

제5 항에 있어서,

상기 예상되는 제외를 갖는 명령은 정적 블록 제외(static block exception)를 발생시키는 동작과 연관된 명령을 포함하는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

## 청구항 7

제6 항에 있어서,

상기 정적 블록 제외는 명령 페이지 결함(instruction page fault); 무효 블록 헤더; 및 무효 명령 인코딩으로 구성된 그룹을 포함하는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

## 청구항 8

제5 항에 있어서,

상기 예상되는 제외를 갖는 명령은 워치포인트(watchpoint)와 연관된 명령을 포함하는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

## 청구항 9

제5 항에 있어서,

상기 예상되는 제외를 갖는 명령은 브레이크포인트(breakpoint)와 연관된 명령을 포함하는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

## 청구항 10

제5 항에 있어서,

상기 예상되는 제외를 갖는 명령은 디버그 스텝 포인트(debug step point)와 연관된 명령을 포함하는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

## 청구항 11

제1 항에 있어서,

상기 명령 블록 내의 모든 명령들이 실행되었는지를 결정하고; 그리고

상기 명령 블록 내의 모든 명령들이 실행된 것으로 결정한 것에 대한 응답으로, 상기 명령 블록 내의 실행된 명령들에 의해 외부 생성된 값들을 커밋(commit)하도록 추가로 구성되는,  
프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

#### 청구항 12

제1 항에 있어서,

상기 기록/재생 회로는 추가로, 상기 실행되는 다음 로드/저장 명령의 생성된 데이터에 대해 상기 기록/재생 상태를 상기 기록/재생 로그 파일에 기록하도록 구성되는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

#### 청구항 13

제1 항에 있어서,

상기 기록/재생 회로는 추가로, 상기 실행되는 다음 로드/저장 명령의 생성된 데이터와 연관하여 상기 실행되는 다음 로드/저장 명령에 의해 어드레싱된 가상 어드레스를 상기 기록/재생 로그 파일에 기록하도록 구성되는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

#### 청구항 14

제1 항에 있어서,

상기 기록/재생 상태가 상기 명령 블록에 대해 유휴 실행 상태일 때, 상기 검출 회로는, 상기 프로세서에 의한 상기 명령 블록의 실행 동안에, 상기 잠재적인 아키텍처 상태 수정과 연관된 상기 명령 블록 내의 명령을 검출하도록 구성되는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

#### 청구항 15

제14 항에 있어서,

상기 기록/재생 상태가 상기 유휴 실행 상태에 있을 때, 상기 명령 블록의 실행 동안에 제외가 발생하는 것에 대한 응답으로, 상기 명령 블록으로 하여금 재실행되게 하도록 구성되는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

#### 청구항 16

제1 항에 있어서,

상기 기록/재생 상태가 상기 활성 상태에 있을 때, 상기 명령 블록의 실행 동안에 제외가 발생하는 것에 대한 응답으로, 상기 기록/재생 회로는 추가로:

상기 명령 블록에 대해 상기 기록/재생 상태를 잠금 상태(locked state)로 설정하고; 그리고

상기 실행되는 다음 로드/저장 명령의 생성된 데이터를 상기 기록/재생 로그 파일에 기록하는 것을 중지하고, 상기 다음 로드/저장 명령에 대해 기록된 상기 이전에 생성된 데이터를 사용하여 상기 다음 로드/저장 명령을 실행하는 것을 중지하도록 구성되는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

#### 청구항 17

제16 항에 있어서,

상기 명령 블록에 대해 상기 기록/재생 상태가 이전 잠금 상태로부터 활성 상태로 복귀하는 것에 대한 응답으로, 상기 기록/재생 회로는 추가로, 상기 실행되는 다음 로드/저장 명령의 생성된 데이터를 상기 기록/재생 로그 파일에 기록하는 것, 및 상기 다음 로드/저장 명령에 대해 기록된 상기 이전에 생성된 데이터를 사용하

여 상기 다음 로드/저장 명령을 실행하는 것을 재개하도록 구성되는,  
프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

### 청구항 18

제17 항에 있어서,

상기 기록/재생 상태가 상기 활성 상태에 있을 때, 상기 명령 블록의 실행 동안에 상기 제외의 발생에 대한 응답으로, 상기 기록/재생 회로는 추가로, 상기 제외의 발생 전에 상기 명령 블록 내의 실행된 명령들에 의해 외부 생성된 값들을 커밋하도록 구성되는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

### 청구항 19

제18 항에 있어서,

상기 기록/재생 회로는 추가로, 상기 명령 블록에 대해 상기 기록/재생 상태가 상기 이전 잠금 상태로부터 상기 활성 상태로 복귀하는 것에 대한 응답으로, 상기 다음 로드/저장 명령에 대해 기록된 상기 이전에 생성된 데이터를 사용하여 상기 다음 로드/저장 명령의 실행을 재개한 후에 실행된 상기 명령 블록 내의 상기 실행된 명령들에 의한 상기 외부 생성된 값들을 커밋하도록 구성되는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

### 청구항 20

제1 항에 있어서,

SOC(system-on-a-chip)에 통합되는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

### 청구항 21

제1 항에 있어서,

셋톱 박스, 엔터테인먼트 유닛, 내비게이션 디바이스, 통신 디바이스, 고정형 위치 데이터 유닛, 모바일 위치 데이터 유닛, 모바일 폰, 셀룰러 폰, 스마트 폰, 태블릿, 패블릿, 컴퓨터, 휴대용 컴퓨터, 데스크탑 컴퓨터, PDA(personal digital assistant), 모니터, 컴퓨터 모니터, 텔레비전, 튜너, 라디오, 위성 라디오, 음악 플레이어, 디지털 음악 플레이어, 휴대용 음악 플레이어, 디지털 비디오 플레이어, 비디오 플레이어, DVD(digital video disc) 플레이어, 휴대용 디지털 비디오 플레이어 및 자동차로 구성된 그룹으로부터 선택된 디바이스에 통합되는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

### 청구항 22

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기로서,

잠재적인 아키텍처 상태 수정과 연관된 명령의 검출, 또는 상기 프로세서에서의 제외의 발생에 대한 응답으로, 명령 블록에 대해 기록/재생 상태를 저장하기 위한 수단을 활성 상태로 설정하기 위한 수단;

상기 명령 블록에 대해 상기 활성 상태로 상기 기록/재생 상태를 저장하기 위한 수단에 대한 응답으로, 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록되는지를 결정하기 위해, 상기 명령 블록에서 실행될 상기 다음 로드/저장 명령에 대응하는 기록/재생 로그 파일을 저장하기 위한 수단 내의 엔트리 상태를 검사하기 위한 수단;

상기 이전에 생성된 데이터가 상기 다음 로드/저장 명령에 대해 상기 기록/재생 로그 파일을 저장하기 위한 수단에 기록되지 않는 것에 대한 응답으로, 실행되는 다음 로드/저장 명령의 생성된 데이터를 상기 기록/재생 로그 파일을 저장하기 위한 수단에 기록하기 위한 수단; 및

상기 이전에 생성된 데이터가 상기 다음 로드/저장 명령에 대해 상기 기록/재생 로그 파일을 저장하기 위한 수

단에 기록되는 것에 대한 응답으로, 상기 다음 로드/저장 명령에 대해 상기 기록/재생 로그 파일을 저장하기 위한 수단에 기록된 상기 이전에 생성된 데이터를 사용하여 상기 다음 로드/저장 명령을 실행하기 위한 수단을 포함하는,

프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기.

### 청구항 23

프로세서에서 명령 블록을 재생하는 방법으로서,

잠재적인 아키텍처 상태 수정과 연관된 명령의 검출, 또는 상기 프로세서에서의 제외의 발생에 대한 응답으로, 명령 블록에 대해 기록/재생 상태를 활성 상태로 설정하는 단계; 및

상기 기록/재생 상태가 상기 명령 블록에 대해 상기 활성 상태인 것에 대한 응답으로,

이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록되는지를 결정하기 위해, 상기 명령 블록에서 실행될 상기 다음 로드/저장 명령에 대응하는 기록/재생 로그 파일 내의 엔트리 상태를 검사하는 단계;

상기 이전에 생성된 데이터가 상기 다음 로드/저장 명령에 대해 상기 기록/재생 로그 파일에 기록되지 않는 것에 대한 응답으로, 실행되는 다음 로드/저장 명령의 생성된 데이터를 상기 기록/재생 로그 파일에 기록하는 단계; 및

상기 이전에 생성된 데이터가 상기 다음 로드/저장 명령에 대해 상기 기록/재생 로그 파일에 기록되는 것에 대한 응답으로, 상기 다음 로드/저장 명령에 대해 상기 기록/재생 로그 파일에 기록된 상기 이전에 생성된 데이터를 사용하여 상기 다음 로드/저장 명령을 실행하는 단계를 포함하는,

프로세서에서 명령 블록을 재생하는 방법.

### 청구항 24

제23 항에 있어서,

상기 프로세서에 의한 상기 명령 블록의 실행 동안에, 상기 잠재적인 아키텍처 상태 수정과 연관된 상기 명령 블록 내의 명령을 검출하는 단계를 포함하는,

프로세서에서 명령 블록을 재생하는 방법.

### 청구항 25

제24 항에 있어서,

상기 프로세서에 의한 상기 명령 블록의 실행 동안에, 잠재적인 부작용을 갖는 로드/저장 명령을 포함하는 상기 명령 블록 내의 명령을 검출하는 단계; 및

상기 잠재적인 부작용을 갖는 상기 로드/저장 명령을 포함하는 상기 명령의 검출에 대한 응답으로, 상기 명령 블록에 대해 상기 기록/재생 상태를 상기 활성 상태로 설정하는 단계를 포함하는,

프로세서에서 명령 블록을 재생하는 방법.

### 청구항 26

제24 항에 있어서,

상기 프로세서에 의한 상기 명령 블록의 실행 동안에, 기록 시스템 구성 레지스터 명령을 포함하는 상기 명령 블록 내의 명령을 검출하는 단계; 및

상기 기록 시스템 구성 레지스터 명령을 포함하는 상기 명령의 검출에 대한 응답으로, 상기 명령 블록에 대해 상기 기록/재생 상태를 상기 활성 상태로 설정하는 단계를 포함하는,

프로세서에서 명령 블록을 재생하는 방법.

### 청구항 27

제24 항에 있어서,

상기 프로세서에 의한 상기 명령 블록의 실행 동안에, 예상되는 제외를 갖는 명령을 포함하는 상기 명령 블록 내의 명령을 검출하는 단계; 및

상기 예상되는 제외를 갖는 명령을 포함하는 상기 명령의 검출에 대한 응답으로, 상기 명령 블록에 대해 상기 기록/재생 상태를 상기 활성 상태로 설정하는 단계를 포함하는,

프로세서에서 명령 블록을 재생하는 방법.

### 청구항 28

제23 항에 있어서,

상기 기록/재생 상태가 상기 활성 상태에 있을 때, 상기 명령 블록의 실행 동안에 제외가 발생하는 것에 대한 응답으로,

상기 명령 블록에 대해 상기 기록/재생 상태를 잠금 상태로 설정하는 단계; 및

상기 실행되는 다음 로드/저장 명령의 생성된 데이터를 상기 기록/재생 로그 파일에 기록하는 것을 중지하고, 상기 다음 로드/저장 명령에 대해 기록된 상기 이전에 생성된 데이터를 사용하여 상기 다음 로드/저장 명령을 실행하는 것을 중지하는 단계를 더 포함하는,

프로세서에서 명령 블록을 재생하는 방법.

### 청구항 29

제23 항에 있어서,

상기 기록/재생 상태가 상기 활성 상태에 있을 때, 상기 명령 블록의 실행 동안에 상기 제외의 발생에 대한 응답으로, 상기 제외의 발생 전에 상기 명령 블록 내의 실행된 명령들에 의해 외부 생성된 값들을 커밋하는 단계를 더 포함하는,

프로세서에서 명령 블록을 재생하는 방법.

### 청구항 30

컴퓨터 실행 가능 명령들이 저장되는 비-일시적인 컴퓨터-판독 가능 저장 매체로서, 상기 컴퓨터 실행 가능 명령들은, 프로세서에 의해 실행될 때, 상기 프로세서로 하여금:

잠재적인 아키텍처 상태 수정과 연관된 명령의 검출, 또는 상기 프로세서에서의 제외의 발생에 대한 응답으로, 명령 블록에 대해 기록/재생 상태를 활성 상태로 설정하게 하고; 그리고

상기 기록/재생 상태가 상기 명령 블록에 대해 상기 활성 상태인 것에 대한 응답으로,

이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록되는지를 결정하기 위해, 상기 명령 블록에서 실행될 상기 다음 로드/저장 명령에 대응하는 기록/재생 로그 파일 내의 엔트리 상태를 검사하게 하고;

상기 이전에 생성된 데이터가 상기 다음 로드/저장 명령에 대해 상기 기록/재생 로그 파일에 기록되지 않는 것에 대한 응답으로, 실행되는 다음 로드/저장 명령의 생성된 데이터를 상기 기록/재생 로그 파일에 기록하게 하고; 그리고

상기 이전에 생성된 데이터가 상기 다음 로드/저장 명령에 대해 상기 기록/재생 로그 파일에 기록되는 것에 대한 응답으로, 상기 다음 로드/저장 명령에 대해 상기 기록/재생 로그 파일에 기록된 상기 이전에 생성된 데이터를 사용하여 상기 다음 로드/저장 명령을 실행하게 하는,

비-일시적인 컴퓨터-판독 가능 저장 매체.

## 발명의 설명

### 기술 분야

[0001] 본 출원은 2015년 12월 28일자로 출원되고 명칭이 "REPLAY OF PARTIALLY-EXECUTED INSTRUCTION BLOCKS IN A PROCESSOR-BASED SYSTEM EMPLOYING A BLOCK-ATOMIC EXECUTION MODEL"인 미국 가특허 출원 일련 번호 제

62/271,475 호를 우선권으로 주장하며, 상기 출원의 전체 내용들은 인용에 의해 본원에 통합된다.

[0002] 또한, 본 출원은 2016년 8월 31일자로 출원되고, 명칭이 "REPLAY OF PARTIALLY EXECUTED INSTRUCTION BLOCKS IN A PROCESSOR-BASED SYSTEM EMPLOYING A BLOCK-ATOMIC EXECUTION MODEL"인 미국 특허 출원 일련 번호 제15/252,323 호를 우선권으로 주장하고, 상기 출원의 전체 내용들은 인용에 의해 본원에 통합된다.

[0003] 본 개시내용의 기술은 일반적으로 프로세서-기반 시스템에서의 명령들의 실행에 관한 것이고, 더 상세하게는, 명령들이 명령 블록들로 그룹화되는 블록-원자 실행 모델을 사용하는 프로세서-기반 시스템들에 관한 것이며, 블록-자동 실행 모델에서 명령 블록 내의 모든 명령들이 커밋(commit)되거나 명령들 중 어떠한 것도 커밋되지 않는다.

## 배경 기술

[0004] 마이크로프로세서들은 매우 다양한 애플리케이션들에서 계산 작업들을 수행한다. 종래의 마이크로프로세서 애플리케이션은, 소프트웨어 명령들을 실행하는 "프로세서 코어들"로 또한 알려진 하나 또는 그 초과의 프로세서들을 포함하는 CPU(central processing unit)를 포함한다. 소프트웨어 명령들은 CPU에 데이터에 기반하여 동작들을 수행하도록 명령한다. 이러한 데이터의 예들은 명령 페치 데이터에 인코딩된 즉시 값, 레지스터에 저장된 데이터, 메모리 내의 위치로부터의 데이터, 및 입력/출력(I/O) 디바이스들과 같은 외부 디바이스들로부터의 데이터를 포함한다. CPU는 결과를 생성하기 위한 명령들에 따라 동작을 수행한다. 이어서, 결과는 레지스터 또는 메모리에 저장되거나, 출력으로서 I/O 디바이스에 제공될 수 있다.

[0005] 일부 CPU들은 "블록-원자" 실행 모델을 사용한다. 블록-원자 실행 모델에서, 명령들(예컨대, 128 개의 명령들)의 세트는 명령 블록들로 그룹화된다. 예컨대, 도 1은 블록-원자 실행 모델을 사용하는 프로세서에 의해 실행될 복수의 명령들(102(1)-102(N))로 구성된 예시적인 명령 블록(100)을 예시한다. 블록-원자 실행 모델에 따라 동작하는 프로세서는 단일 엔티티로서 명령 블록(100)을 논리적으로 페치, 실행 및 커밋(commit)한다. 블록-원자 실행 모델은 OoP(out-of-order processor)의 복잡성을 감소시키는 이점을 갖는다. 예컨대, 블록-원자 실행 모델을 사용하는 OoP는 각각의 명령의 실행 후에 프로세서의 정확한 상태를 다시 보고할 필요가 없다. 이와 관련하여, 도 1에 도시된 예시적인 명령 블록(100)에서, 실행되지 않을 경우 글로벌 레지스터들에 저장될 명령들(102(1), 102(2), 102(4), 102(N-2)-102(N))의 실행으로부터 중간 생성된 결과들(104(1), 104(2), 104(4), 104(N-2)-104(N))은 레지스터들에 저장될 필요가 없다. 예로서 명령(102(1))을 취하면, 그의 중간 생성 결과(104(1))는, 중간 생성 결과(104(1))를 저장하지 않고서, 피어-투-피어 방식으로 명령 블록(100) 내의 소비자 명령(102(2))에 제공될 수 있다. 이것은 더 적은 레지스터 판독 및 기록 동작들을 가능하게 한다. 그러나, 블록-원자 실행 모델에서, 중간 생성 결과들이 글로벌 레지스터들에 저장되지 않기 때문에, 명령 블록(100)의 외부 실행 결과들이(예컨대, 메모리, I/O 디바이스 등으로) 커밋되기 전에, 모든 명령들이 실행되어야 한다. 예컨대, 도 1에 도시된 바와 같이, 명령 블록(100) 내부의 명령(102(4))의 실행 동안에 제외(108)(예컨대, 정확한 제외, 이를테면, 페이지 결함 또는 디버그 브레이크포인트 또는 설정 포인트)가 발생하면, 남아있는 명령들(102(N-2)-102(N))이 실행되지 않는데, 왜냐하면 중간 생성 결과들(104(1), 104(2), 104(4))이 보존되지 않기 때문이며, 실행은 다음 명령(102(N-2))에서 시작될 수 있다. 명령 블록(100)은, 제외(108)가 해결된 후에 시작부터 다시-실행된다.

[0006] 따라서, 블록-원자 실행 모델을 사용하는 프로세서가 복잡성이 감소된다는 이점을 갖지만, 블록-원자 실행 모델을 사용하는 프로세서는, 외부 결과들이 커밋되기 전에 명령 블록 내의 모든 명령들을 실행해야 한다는 단점을 갖는다. 예로서, 이것은 멀티-스레드 CPU의 존재 시에 디버깅을 더 어렵게 만들 수 있는데, 왜냐하면 일반적으로 명령 블록의 실행 동안에 발생한 제외 또는 브레이크포인트를 유도한 어떠한 거동도 재생성하는 것이 불가능하기 때문이다. 다른 스레드의 이전 기록 동작으로부터 판독될 수 있는 명령 블록의 중간 생성 결과들은 저장되지 않는다. 이것은 또한, I/O 동작이 제외 발생 이전에 이미 실행된 경우에, I/O 디바이스 부작용들과 같은 부작용 동작들로 인한 어려움들을 야기한다. 예컨대, 도 1의 명령 블록(100) 내의 I/O 동작 명령(102(4))의 실행이 개시되었지만 제외(108)의 발생 전에 완료되지 않은 경우, 명령(102(4))의 I/O 동작으로부터 판독된 중간 데이터는 명령 블록(100)의 재실행 동안에 더 이상 이용 가능하지 않을 수 있다. 따라서, 이러한 예에서, 명령(102(4))의 제1 실행 동안 판독된 중간 데이터는 레지스터(r2)에 재생 가능하게 저장되지 않을 수 있다.

## 발명의 내용

[0007] 본원에 개시된 양상들은 블록-원자 실행 모델을 사용하는 프로세서-기반 시스템에서 부분적으로 실행된

명령 블록들의 재생을 포함한다. 이와 관련하여, 일 양상에서, 부분적인 재생 제어기(partial replay controller)가 블록-원자 실행 모델을 사용하는 CPU(central processing unit)의 프로세서(들)에 제공된다. 블록-원자 실행 모델에서, 명령들은, 외부 생성 결과들이 커밋되기 전에 프로세서(예컨대, OoP(out-of-order processor))에서 완전히 실행되는 명령 블록들로 그룹화된다. 이것은 외부 생성 결과들의 값들에 영향을 줄 수 있는 모든 로드/저장 종속성을 해결하고, 따라서 프로세서 및 다른 프로세스들의 아키텍처 상태는 그러한 외부 생성 결과들에 종속된다. 로드/저장 동작들로부터 생성된 결과들이 제외로 인해 로딩된 명령 블록의 재실행 동안에 재생될 수 없는 것을 회피하기 위해, 부분적인 재생 제어기는 명령 블록의 재실행 동안에 로드/저장 명령들의 결과들을 기록/재생하도록 구성된다. 이와 관련하여, 특정 양상들에서, 명령이 잠재적인 아키텍처 상태 수정(예컨대, 잠재적인 부작용)과 연관된 명령 블록에서 검출되거나, 제외가 발생하면, 명령 블록 내의 명령들의 실행 동안에, 명령 블록이 재실행된다. 명령 블록의 재실행 동안에, 부분적인 재생 제어기는 로드/저장 명령들로부터 생성된 결과들을 기록/재생하도록 구성된다. 따라서, 명령 블록의 재실행 동안에 제외가 발생하면, 제외가 발생하기 전에 실행된 로드/저장 명령들에 대해 이전에 기록된 생성된 결과들이 제외가 해결된 후에 명령 블록의 재실행 동안에 재생될 수 있다. 이러한 방식으로, 잠재적인 아키텍처 상태 수정과 연관된 동작들을 유도하거나 또는 제외가 발생하기 전의 명령들의 실행은, 부작용을 반복하지 않고서, 이전에 생성된 결과들로 결정론적으로 반복될 수 있다.

[0008] 그러나, 명령 블록을 재실행하는 동안에 제외가 발생하지 않으면, 이것은 생성된 결과들이 기록되면서 명령 블록 내의 모든 명령들이 완전히 실행되었다는 것을 의미한다는 것을 유의하라. 이러한 경우에 명령 블록이 재실행되지 않기 때문에, 이전에 실행된 로드/저장 명령들에 대해 기록된 생성된 결과들을 재생할 필요가 없다.

[0009] 이와 관련하여, 하나의 예시적인 양상에서, 프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기가 제공된다. 부분적인 재생 제어기는, 잠재적인 아키텍처 상태 수정(potential architectural state modification)과 연관된 명령의 검출 또는 프로세서에서의 제외의 발생에 대한 응답으로, 명령 블록에 대해 기록/재생 상태를 활성 상태로 설정하도록 구성된 검출 회로를 포함한다. 부분적인 재생 제어기는 또한 기록/재생 회로를 포함한다. 기록/재생 상태가 명령 블록에 대해 활성 상태인 것에 대한 응답으로, 기록/재생 회로는, 이전에 생성된 데이터가 다음 로드/저장 명령(next load/store instruction)에 대해 기록되는지를 결정하기 위해, 명령 블록에서 실행될 다음 로드/저장 명령에 대응하는 기록/재생 로그 파일 내의 엔트리 상태를 검사하도록 구성된다. 또한, 기록/재생 상태가 명령 블록에 대해 활성 상태인 것에 대한 응답으로, 기록/재생 회로는 또한, 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록/재생 로그 파일에 기록되지 않는 것에 대한 응답으로, 실행된 다음 로드/저장 명령의 생성된 데이터를 기록/재생 로그 파일에 기록하도록 구성된다. 또한, 기록/재생 상태가 명령 블록에 대해 활성 상태인 것에 대한 응답으로, 기록/재생 회로는 또한, 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록/재생 로그 파일에 기록되는 것에 대한 응답으로, 다음 로드/저장 명령에 대해 기록/재생 로그 파일에 기록된 이전에 생성된 데이터를 사용하여 다음 로드/저장 명령을 실행하도록 구성된다.

[0010] 다른 예시적인 양상에서, 프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기가 제공된다. 부분적인 재생 제어기는, 잠재적인 아키텍처 상태 수정과 연관된 명령의 검출, 또는 프로세서에서의 제외의 발생에 대한 응답으로, 명령 블록에 대해 기록/재생 로그 파일을 저장하기 위한 수단을 활성 상태로 설정하기 위한 수단을 포함한다. 부분적인 재생 제어기는 또한, 명령 블록에 대해 기록/재생 로그 파일을 저장하기 위한 수단을 활성 상태로 설정한 것에 대한 응답으로, 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록되는지를 결정하기 위해, 명령 블록에서 실행될 다음 로드/저장 명령에 대응하는 기록/재생 로그 파일을 저장하기 위한 수단 내의 엔트리 상태를 검사하기 위한 수단을 포함한다. 부분적인 재생 제어기는 또한, 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록/재생 로그 파일을 저장하기 위한 수단에 기록되지 않는 것에 대한 응답으로, 실행된 다음 로드/저장 명령의 생성된 데이터를 기록/재생 로그 파일을 저장하기 위한 수단에 기록하는 수단을 포함한다. 부분적인 재생 제어기는 또한, 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록/재생 로그 파일을 저장하기 위한 수단에 기록된 것에 대한 응답으로, 다음 로드/저장 명령에 대해 기록/재생 로그 파일을 저장하기 위한 수단에 기록된 이전에 생성된 데이터를 사용하여 다음 로드/저장 명령을 실행하기 위한 수단을 포함한다.

[0011] 다른 예시적인 양상에서, 프로세서에서 명령 블록을 재생하는 방법이 제공된다. 방법은, 잠재적인 아키텍처 상태 수정과 연관된 명령의 검출, 또는 프로세서에서의 제외의 발생에 대한 응답으로, 명령 블록에 대해 기록/재생 상태를 활성 상태로 설정하는 단계를 포함한다. 기록/재생 상태가 명령 블록에 대해 활성 상태인 것

에 대한 응답으로, 방법은 또한, 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록되는지를 결정하기 위해, 명령 블록에서 실행될 다음 로드/저장 명령에 대응하는 기록/재생 로그 파일 내의 엔트리 상태를 검사하는 단계, 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록/재생 로그 파일에 기록되지 않는 것에 대한 응답으로, 실행된 다음 로드/저장 명령의 생성된 데이터를 기록/재생 로그 파일에 기록하는 단계, 및 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록/재생 로그 파일에 기록되는 것에 대한 응답으로, 다음 로드/저장 명령에 대해 기록/재생 로그 파일에 기록된 이전에 생성된 데이터를 사용하여 다음 로드/저장 명령을 실행하는 단계를 포함한다.

[0012] 다른 예시적인 양상에서, 컴퓨터 실행 가능 명령들이 저장된 비일시적인 컴퓨터-판독 가능 매체가 제공된다. 컴퓨터 실행 가능 명령들은, 프로세서에 의해 실행될 때, 프로세서로 하여금, 잠재적인 아키텍처 상태 수정과 연관된 명령의 검출, 또는 프로세서에서의 제외의 발생에 대한 응답으로, 명령 블록에 대해 기록/재생 상태를 활성 상태로 설정하게 한다. 기록/재생 상태가 명령 블록에 대해 활성 상태인 것에 대한 응답으로, 컴퓨터 실행 가능 명령들은 또한, 프로세서에 의해 실행될 때, 프로세서로 하여금, 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록되는지를 결정하기 위해, 명령 블록에서 실행될 다음 로드/저장 명령에 대응하는 기록/재생 로그 파일 내의 엔트리 상태를 검사하게 한다. 기록/재생 상태가 명령 블록에 대해 활성 상태인 것에 대한 응답으로, 컴퓨터 실행 가능 명령들은 또한, 프로세서에 의해 실행될 때, 프로세서로 하여금, 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록/재생 로그 파일에 기록되지 않는 것에 대한 응답으로, 실행된 다음 로드/저장 명령의 생성된 데이터를 기록/재생 로그 파일에 기록하게 한다. 기록/재생 상태가 명령 블록에 대해 활성 상태인 것에 대한 응답으로, 컴퓨터 실행 가능 명령들은 또한, 프로세서에 의해 실행될 때, 프로세서로 하여금, 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록/재생 로그 파일에 기록되는 것에 대한 응답으로, 다음 로드/저장 명령에 대해 기록/재생 로그 파일에 기록된 이전에 생성된 데이터를 사용하여 다음 로드/저장 명령을 실행하게 한다.

### 도면의 간단한 설명

[0013] 도 1은 블록-원자 실행 모델을 사용하는 CPU(central processing unit)에 의해 실행되도록 구성된 예시적인 명령 블록을 예시하며, 입력/출력(I/O) 동작이 명령 블록에서 실행된 후에 명령 블록 내의 명령들의 실행 동안에 제외가 발생했다.

[0014] 도 2는 예시적인 다중 프로세서("멀티 프로세서") CPU의 블록도이고, 각각의 프로세서는 외부 메모리 및 I/O 디바이스들에 대한 액세스들을 포함하는 기능들을 수행하기 위한 소프트웨어 명령들을 실행하도록 구성된다.

[0015] 도 3은 도 2의 멀티-프로세서 CPU에 포함될 수 있는 프로세서의 예시적인 세부사항을 예시한 블록도이고, 프로세서는, 명령 블록 내의 명령들의 실행 동안에, 잠재적인 아키텍처 상태 수정과 연관된 명령 블록 내의 명령을 검출하는 것에 대한 응답으로, 및/또는 제외의 발생에 대한 응답으로, 명령 블록의 재실행 동안에 로드/저장 명령들의 결과들을 기록 및/또는 재생하도록 구성된 부분적인 재생 제어기를 포함한다.

[0016] 도 4는, 잠재적인 아키텍처 상태 수정과 연관된 명령의 검출에 대한 응답으로 그리고/또는 프로세서에서의 제외의 발생에 대한 응답으로, 명령 블록의 재실행 동안에 로드/저장 명령들의 결과들을 기록 및/또는 재생하기 위해 기록/재생 활성 상태에 진입하기 위한, 도 3의 프로세서 내의 부분적인 재생 제어기에 의해 수행될 수 있는 예시적인 프로세스를 예시하는 흐름도이다.

[0017] 도 5는, 명령 블록의 실행 및 재실행 동안에 도 3의 부분적인 재생 제어기의 추가의 예시적인 논의를 용이하게 하기 위해, 잠재적인 아키텍처 상태 수정과 연관된 명령들을 포함하는, 도 3의 프로세서에 의해 실행되는 예시적인 명령 블록을 예시한다.

[0018] 도 6은 잠재적인 아키텍처 상태 수정과 연관된 명령 블록 내의 명령을 검출하는 것에 대한 응답으로, 명령 블록의 재실행 동안에 로드/저장 명령들의 결과들을 기록 및 재생하기 위한, 도 3의 부분적인 재생 제어기에 의해 수행될 수 있는 예시적인 프로세스를 예시하는 흐름도이다.

[0019] 도 7은 프로세서에서 명령 블록의 실행 동안에 도 3의 프로세서 내의 부분적인 재생 제어기의 예시적인 동작을 예시하는 예시적인 상태 머신이다.

[0020] 도 8은, 본원에 개시된 예들에 따라, 잠재적인 아키텍처 상태 수정과 연관된 명령 블록 내의 명령을 검출하는 것 및/또는 명령 블록의 유휴 실행 상태 동안에 제외의 발생에 대한 응답으로, 명령 블록의 재실행 동안

에 로드/저장 명령들의 결과들을 기록 및/또는 재생하도록 구성된 부분적인 재생 제어기를 포함하는 멀티-프로세서 CPU를 포함하는 예시적인 프로세서-기반 시스템의 블록도이다.

### 발명을 실시하기 위한 구체적인 내용

[0014]

[0021] 이제 도시된 도면들을 참조하면, 본 개시내용의 몇몇 예시적 양상들이 설명된다. "예시적인"이라는 단어는 본원에서 "예, 사례 또는 예시로서 기능하는 것"을 의미하도록 사용된다. 본원에서 "예시적인" 것으로 설명되는 임의의 양상은 반드시 다른 양상들에 비해 선호되거나 유리한 것으로 해석될 필요는 없다.

[0015]

[0022] 본원에 개시된 양상들은 블록-원자 실행 모델을 사용하는 프로세서-기반 시스템에서 부분적으로 실행된 명령 블록들의 재생을 포함한다. 이와 관련하여, 일 양상에서, 부분적인 재생 제어기가 블록-원자 실행 모델을 사용하는 CPU(central processing unit)의 프로세서(들)에 제공된다. 블록-원자 실행 모델에서, 명령들은, 외부 생성 결과들이 커밋되기 전에 프로세서(예컨대, OoP(out-of-order processor))에서 완전히 실행되는 명령 블록들로 그룹화된다. 이것은 외부 생성 결과들의 값들에 영향을 줄 수 있는 모든 로드/저장 종속성을 해결하고, 따라서 프로세서 및 다른 프로세스들의 아키텍처 상태는 그러한 외부 생성 결과들에 종속된다. 로드/저장 동작들로부터 생성된 결과들이 제외로 인해 로딩된 명령 블록의 재실행 동안에 재생될 수 없는 것을 회피하기 위해, 부분적인 재생 제어기는 명령 블록의 재실행 동안에 로드/저장 명령들의 결과들을 기록/재생하도록 구성된다. 이와 관련하여, 특정 양상들에서, 명령이 잠재적인 아키텍처 상태 수정(예컨대, 잠재적인 부작용)과 연관된 명령 블록에서 검출되거나, 제외가 발생하면, 명령 블록 내의 명령들의 실행 동안에, 명령 블록이 재실행된다. 명령 블록의 재실행 동안에, 부분적인 재생 제어기는 로드/저장 명령들로부터 생성된 결과들을 기록하도록 구성된다. 따라서, 명령 블록의 재실행 동안에 제외가 발생하면, 제외가 발생하기 전에 실행된 로드/저장 명령들에 대해 이전에 기록된 생성된 결과들이 제외가 해결된 후에 명령 블록의 재실행 동안에 재생될 수 있다. 이러한 방식으로, 잠재적인 아키텍처 상태 수정과 연관된 동작들을 유도하거나 또는 제외가 발생하기 전의 명령들의 실행은, 부작용을 반복하지 않고서, 이전에 생성된 결과들로 결정론적으로 반복될 수 있다.

[0016]

[0023] 이와 관련하여, 도 2는, 본원에서 프로세서(204(1)-204(P))로서 참조되는 다수의 프로세서 코어들(204(1)-204(P))을 갖는 CPU(202)를 포함하는 예시적인 프로세서-기반 시스템(200)의 블록도이고, 여기서 'P'는 CPU(202)에 포함되는 프로세서들(204)의 수이다. 비제한적인 예로서, 프로세서들(204(1)-204(P))은 입력 데이터의 이용 가능성에 기반하여 명령들의 비순차적인 실행을 수행하도록 구성된 OoP들(out-of-order processors)일 수 있다. 이것은, 프로그램에서 다음 명령에 대한 데이터를 리트리브(retrieve)하기 위해 선행 명령이 완료 되기를 대기하는 동안에 유휴 상태가 되는 것을 회피하기 위해, 프로그램에서 자신들의 오리지널 순서에 따라 명령들을 실행해야 하는 순차적인 프로세서와 반대이다. 이러한 예에서 CPU(202) 내의 하나 또는 그 초과의 프로세서들(204(1)-204(P))은 부분적인 재생 제어기(206(1)-206(P))를 포함한다. 이러한 예에서, CPU(202) 내의 각각의 프로세서(204(1)-204(P))는 부분적인 재생 제어기(206(1)-206(P))를 포함하지만, 그러한 것이 요구되지 않는 않는다.

[0017]

[0024] 도 3에서 시작하여 아래에 더 상세히 논의될 바와 같이, 프로세서들(204(1)-204(P))에 포함되는 부분적인 재생 제어기들(206(1)-206(P)) 각각은, 명령 블록 내의 명령들의 실행 동안에 잠재적인 아키텍처 상태 수정(예컨대, 잠재적인 부작용)과 관련된 명령을 검출하는 것에 대한 응답으로, 명령 블록의 재실행 동안에 로드/저장 명령들의 결과들을 기록 및/또는 재생하도록 구성된다. 부분적인 재생 제어기들(206(1)-206(P))의 양상들을 논의하기 전에, 도 2의 CPU(202)의 다른 컴포넌트들에 대한 설명이 먼저 아래에 논의된다.

[0018]

[0025] 이와 관련하여, 도 2를 계속 참조하면, 이러한 예에서 각각의 프로세서(204(1)-204(P))는, 오프-프로세서(off-processor)의 ("어드레스"에서) 메모리 액세스 요청을 어드레스 버스(210)에 제공할 필요없이, 온-프로세서(on-processor)의 캐싱된 데이터에 대한 액세스를 제공하기 위한 캐시 메모리("캐시")(208(1)-208(P))(예컨대, 레벨 2(L2) 캐시)를 포함할 수 있다. 오프-프로세서 메모리 액세스 요청들에 대해, 프로세서들(204(1)-204(P))은 시스템 메모리(214)에 액세스하기 위해 어드레스 버스(210) 상의 "어드레스"를 메모리 제어기(212)에 제공하도록 구성된다. 요청하는 프로세서(204(1)-204(P))에 의해 기록될 "데이터"는 데이터 버스(216)를 통해 시스템 메모리(214)에 제공된다. 시스템 메모리(214)로부터 판독된 "데이터"는 요청하는 프로세서(204(1)-204(P))에 제공되며, 시스템 메모리(214)에 의해, 데이터 버스(216) 상에 제공된다. 입력/출력(I/O) 디바이스(들)(218)는 또한, I/O 디바이스(들)(218)에 대한 프로세서들(204(1)-204(P))의 액세스를 제공하기 위해, 도 2의 프로세서-기반 시스템(200) 내의 어드레스 버스(210) 및 데이터 버스(216)에 커플링된다. 프로세서-기반 시스템(200)은 SoC(system-on-a-chip)(220)로서 제공될 수 있고, 프로세서들(204(1)-204(P)) 및 도 2에 예시되고 위에 설명된 관련 컴포넌트들이 칩 상에 함께 통합된다.

[0019]

[0026] 예시적인 부분적인 재생 제어기 및 지원하는 데이터 구조들을 포함하는, CPU(202) 내의 각각의 프로세서(204(1)-204(P))의 예시적인 세부사항들이 도 3에서 일반적으로 "프로세서(204)"로 도시된다. 도 3에 도시된 바와 같이, 프로세서(204)는, 디코드/디스패치 완료 유닛(306)에 제공하기 위한, 명령 캐시(304)에 저장된 명령들(302)을 폐칭할 수 있는 명령 메모리 관리 유닛(MMU)(300)을 포함한다. 명령 캐시(304)는 도 2의 시스템 메모리(214)와 같은 메인 메모리로부터 실행될 명령들(302)을 로딩할 수 있다. 이러한 예에서, 명령들(302)은 실행될 명령 블록(308)으로 함께 그룹화된다. 디코드/디스패치 완료 유닛(306)은 디코딩된 명령 블록(308)에 대한 실행될 명령들(302)을 디스패치/완료 버스(들)(310)에 제공한다. 이전에 논의된 바와 같이, 프로세서(204)가 블록-원자 실행 모델을 사용하기 때문에, 주어진 명령 블록(308) 내의 모든 명령들(302)은 실행된 명령들(302)의 결과들이 커밋되기 전에 완전히 실행된다. 이러한 예에서, 실행될 명령 블록(308) 내의 부동 소수점 명령들(302F)을 수신하기 위한 부동 소수점 유닛(들)(312)이 제공되며, 생성된 결과들은 부동 소수점 레지스터 파일(314)에 저장된다. 유사하게, 실행될, 실행되는 명령 블록(308) 내의 정수 명령들(302I)을 수신하기 위해 정수 유닛(들)(316)이 제공되며, 생성된 결과들은 GPR(general purpose register) 파일(318)에 저장된다. 명령 블록(308)으로부터 로드/저장 명령들(302L)을 수신하기 위한 로드/저장 유닛(320)이 제공된다. 특정 로드/저장 명령(302L)에 따라, 실행된 로드/저장 명령들(302L)로부터 생성된 결과들은, 어드레스 버스(210) 및 데이터 버스(216)에 커플링된 버스 인터페이스 유닛(324)을 통해 외부 메모리에 액세스하기 위해, 범용 레지스터 파일(318) 또는 메모리 큐(322)에 저장될 수 있다. 외부 메모리에 대한 액세스를 요구하는 실행된 로드/저장 명령들(302L)은 또한 실행된 로드/저장 명령들(302L)에 기반하여 외부 메모리에 입력되거나 외부 메모리에 입력될 데이터를 관리하기 위해 데이터 MMU(326)에 제공된다.

[0020]

[0027] 도 3을 계속 참조하면, 도 1의 예시적인 명령 블록(100)의 명령들(102(1)-102(N))과 같은 명령들이 부분적인 재생 제어기(206)가 제공되지 않고서 명령 블록(308)으로서 프로세서(204)에 의해 실행되었다면, 그리고 명령 블록(308) 내부의 명령(302)의 실행 동안에 제외가 발생하면, 명령 블록(308) 내의 남아있는 명령들(302)은 실행되지 않는다. 중간 생성된 결과들은 보존되지 않고, 실행은, 제외가 해결된 후에, 명령 블록(308) 내의 다음 명령(302)에서 시작될 수 있다. 이와 관련하여, 명령 블록(308) 내의 명령들(302)은, 제외가 해결된 후에 시작부터 재실행될 것이다. 이것은 프로세서(204)에 의해 실행되는 디버깅 명령들(302)을 어렵게 만들 수 있는데, 왜냐하면 명령 블록(308) 내의 명령들(302)의 실행 동안 발생된 제외를 유도한 어떠한 거동도 재생하는 것이 일반적으로 불가능하기 때문이다. 다른 스파드에서의 이전의 기록 동작으로부터 판독될 수 있는, 명령 블록(308)으로부터의 실행된 명령들(302)의 중간 생성된 결과들은 저장되지 않는다. 이것은 또한 I/O 디바이스 부작용들과 같은 부작용을 가질 수 있거나 갖는 로드/저장 명령들(302L)에 어려움들을 야기하고, 여기서 I/O 디바이스(들)(218)(도 2 참조)에 대한 I/O 동작은 제외 전에 이미 실행되었다. 이와 관련하여, 명령 블록(308)의 로드/저장 명령(302L)이 개시되었지만 제외 이전에 완료되지 않은 경우에, I/O 동작으로부터 판독된 중간 데이터는 명령 블록(308) 내의 명령들(302)을 재실행하기 위해 더 이상 이용 가능하지 않을 수 있는데, 왜냐하면 중간 데이터가 저장되지 않았기 때문이다.

[0021]

[0028] 이와 관련하여, 부분적인 재생 제어기(206)는, 잠재적인 아키텍처 상태 수정(예컨대, 잠재적인 부작용)과 연관된 명령 블록(308) 내의 명령(302)의 검출에 대한 응답으로, 명령 블록(308) 내의 명령들(302)의 재실행 동안에, 로드/저장 명령들(302L)의 결과들을 기록 및/또는 재생하도록 프로세서(204)에 제공된다. 부분적인 재생 제어기(206)는 로드/저장 유닛(320)을 포함하는 로드/저장 시스템(328)의 부분으로서 또는 로드/저장 유닛(320)과 별개로 제공될 수 있다. 명령(302)은, 자신이 프로세서들(204(1)-204(P))을 포함하는 CPU(202)의 아키텍처 상태를 수정하는 동작을 수반하는 경우에 잠재적인 아키텍처 상태 수정과 연관되고, 명령 블록(308) 외부의 기능들 또는 프로세스들에의 관측 가능한 상호작용을 가질 수 있다. 예컨대, 잠재적인 아키텍처 상태 수정과 연관된 명령(302)은 명령 블록(308) 및/또는 다른 프로세서들(204(1)-204(P)) 외부의 다른 프로세스들 및 스파드들에 영향을 줄 수 있는 외부 생성 결과들에 영향을 줄 수 있다. 그러한 명령들(302)은 외부 생성 결과들의 값들에 영향을 줄 수 있는 로드/저장 종속성을 가질 수 있고, 따라서 프로세서들(204(1)-204(P)) 및 다른 프로세스들의 아키텍처 상태는 그러한 외부 생성 결과들에 종속된다.

[0022]

[0029] 예컨대, 명령 블록(308)은 잠재적으로 아키텍처 상태를 수정하고 따라서 부작용을 발생시킬 수 있는 명령(302)을 가질 수 있다. 비제한적 예들은 글로벌 변수를 수정하는 것, 시스템 구성 레지스터에 기록하는 것, 예상된 제외(예컨대, 브레이크포인트, 설정 포인트 또는 워치포인트(watchpoint))를 상승시키는 것, 및 데이터를 메모리 또는 I/O 디바이스에 기록하는 것을 포함한다. 아키텍처 상태는 또한, 데이터가 큐에서 팝 오프되고 따라서 후속 판독 동작에서 재생되지 않는 경우 I/O 디바이스와 같은 디바이스로부터 데이터를 판독하는 것과 같은 부작용을 갖는 동작으로부터 수정될 수 있다. 명령(302)이 잠재적인 아키텍처 상태 수정과 연관되면, 아

키텍처 상태 수정에 의해 영향을 받는 거동 또는 다른 프로세스들이 평가 순서에 의존한다.

[0023] [0030] 부작용들을 갖거나 이들을 발생시키는 잠재적인 아키텍처 상태 수정들과 연관된 명령들(302)에 대해, 명령들(302)이 명령들(302)의 임의의 재실행 동안에 동작하는 조건들을 재생성하거나 재생하는 것이 어려울 수 있다. 따라서, 명령 블록(308)이 완전히 실행되기 전에 제외가 발생하면, 잠재적인 아키텍처 상태 수정들과 연관된 명령들(302)의 동작을 수행하기 위해 사용된 중간 생성 결과들은, 제외가 해결된 후에, 더 이상 이용 가능하지 않거나 동일한 값을 가질 수 없다. 그러나, 부분적인 재생 제어기(206)가 잠재적인 아키텍처 상태 수정들과 연관된 명령 블록(308) 내의 명령(302)의 검출에 대한 응답으로 명령 블록(308)의 재실행 동안에 로드/저장 명령들(302L)의 결과들을 기록 및/또는 재생하도록 구성되는 경우에, 중간 생성 결과들은 명령 블록(308)의 재실행 동안에 기록될 수 있다. 따라서, 제외가 발생하면, 프로세서(204)의 아키텍처 상태에 영향을 주는, 명령 블록(308)으로부터 외부적으로 생성된 값들의 무결성을 보장하기 위해 제외가 해결된 후에, 명령 블록(308)의 다른 재실행 동안에 그러한 중간 결과들이 재생될 수 있다.

[0024] [0031] 예컨대, 제외는 명령 블록(308) 실행 경계에서 발생하는 블록 제외 및 명령 블록(308) 내의 명령(302) 경계에서 발생하는 명령 제외를 포함하는 정확한 제외일 수 있다. 제외들의 예들은 산술 제외들: 제로로 나누기, 나눗셈 오버플로우(division overflow), IEEE 부동 소수점, 권한 위반들(비-MMU), 데이터 페이지 결함들(페이지가 매핑되지 않음, 권한 위반, 기록 금지), 정정할 수 없는 정확한 ECC(error correcting code) 에러, 다른 정확한 하드웨어 결함들(TBD)을 포함한다. 제외는 또한 디버깅 모드에서 디버거 브레이크포인트 또는 다른 예들로서 위치포인트일 수 있다. 디버깅 모드 동안에 또는 위치포인트에서 명령 블록(308)으로부터 이전에 생성된 데이터를 기록 및 재생하는 능력은 실행 동안에 모든 로딩되거나 저장된 데이터의 상태를 볼 수 있는 데 특히 유용할 수 있다.

[0025] 그러나, 도 3의 프로세서(204)는, 이를테면, 명령 블록(308)의 실행 동안에 발생하는 제외로 인해, 명령 블록(308)이 프로세서(204)에 의해 재실행되어야 하는 경우, 명령 블록(308) 내의 로드/저장 명령(302L)으로부터 생성된 데이터를 재생할 수 없는 상황을 회피하기 위한 부분적인 재생 제어기(206)를 포함한다. 아래에 더 상세히 논의되는 바와 같이, 부분적인 재생 제어기(206)는 블록-원자 실행 모델을 사용하는 프로세서(204)에서 명령 제외가 재생 가능하게 하도록 구성된다. 부분적인 재생 제어기(206)는 부분적인 블록 재생(PBR) 모델이 사용되도록 허용하고, 제외를 유도하는 명령 블록(308) 실행이 결정론적으로 재생되도록 허용하기에 충분한 상태가 제외 핸들러 또는 디버거에 보고될 수 있다. 이러한 시나리오에서, 명령 블록(308)의 실행은 범용 레지스터들 및 메모리에 부작용들을 유발하지 않는다는 점에서 원자적으로 남아있다. 부분적인 블록 재생은 또한 부작용 동작들(강하게 순서화된 로드 및 저장 및 시스템 레지스터 기록들)이 명령 블록(308) 구조 내에 적합하도록 허용한다. 부분적인 재생 제어기(206)에 의해 가능해진 PBR 모델은 부작용을 갖는 명령 블록(308) 내의 명령들(302)에 대한 블록 원자성(atomicity)의 약화를 허용한다. 제외, 부작용 동작들을 유도하는 실행을 결정론적으로 반복한 후에, 그리고 이어서 부작용들 자체들을 반복하지 않고서, 명령 블록(308)이 재개되도록 허용하기에 충분한 상태가 저장된다.

[0026] 또한 이하에 더 상세히 논의될 바와 같이, 부분적인 재생 제어기(206)는 또한, 프로세서(204)에서 실제 제외의 발생에 대한 응답으로, 명령 블록(308)의 재실행 동안에 로드/저장 명령들(302L)의 결과들을 기록 및/또는 재생하도록 구성될 수 있다. 예컨대, 이러한 제외들은 정적 블록 제외, 이를테면, 명령 페이지 결함, 무효 블록 헤더 및 무효 명령 인코딩을 포함할 수 있다. 따라서, 프로세서(204)가 제외로부터 회복한 후에, 부분적인 재생 제어기(206)는 명령 블록(308)의 재실행 동안에 로드/저장 명령들(302L)의 결과들을 기록 및/또는 재생하도록 구성될 수 있다. 이는, 제외가 다시 발생하거나, 다른 제외가 발생하면, 명령 블록(308)의 재실행 동안에, 다른 스레드 내의 이전 기록 동작으로부터 판독될 수 있는 명령 블록(308)의 중간 생성 결과들이 저장되고, 명령 블록(308)이 다른 제외의 발생으로 인해 재실행되어야 하면, 재생될 수 있도록 하기 위해서이다. 그렇지 않으면, 로드/저장 동작들, 이를테면 I/O 동작들을 수행하기 위해 입력된 중간 데이터는 명령 블록(308)을 재실행하기 위해 더 이상 이용 가능하지 않을 수 있는데, 왜냐하면 중간 데이터가 저장되지 않았기 때문이다. 또한, 제외가 정확한 제외이면, 제외가 발생하기 전에 실행되는 명령 블록(308) 내의 명령들(302)로부터 생성된 결과들은 부분적인 커밋 동작에서 프로세서(204)에 의해 커밋될 수 있다. 따라서, 프로세서(204)가 제외로부터 회복한 후에, 부분적인 재생 제어기(206)는 로드/저장 명령들(302L)의 결과들을 기록 및/또는 재생하기 위해 명령 블록(308)의 시작으로부터 실행을 시작하도록 구성될 수 있다. 이전에 실행된 로드/저장 명령들(302L)의 이전에 기록된 결과들은 명령 블록(308)의 재실행 동안에 재생된다. 그러나, 일 예에서, 부분적 커밋 동작의 부분으로서 명령 블록(308)의 이전 실행 동안에 커밋된 로드/저장 명령들(302L)로부터의 임의의 생성된 데이터는, 명령 블록(308)이 완전히 실행될 때, 완전한 커밋의 후속의 부분적인 재커밋 동작 동안에 재커밋되지 않는다.

이것은, 이러한 이전에 실행된 로드/저장 명령들(302L)의 결과들이 이미 이전에 커밋되었기 때문이다. 따라서, 부분적인 커밋 기능을 사용하는 것이 실행 효율을 증가시킬 수 있는데, 왜냐하면 명령 블록(308)이 제외로부터 복구한 후에 시작 명령(302)으로부터 재실행될 필요가 없기 때문이다. 부분적인 재생 제어기(206)는, 로드/저장 명령(302L)의 재생으로부터 생성된 그러한 결과들이 재커밋되지 않도록, 커밋된 이전에 실행된 로드/저장 명령들(302L)로부터의 결과들을 추적하도록 구성될 수 있다.

[0027] 이와 관련하여, 도 4는, 명령 블록(308)의 재실행 동안에 로드/저장 명령들(302L)의 결과들을 기록 및/ 또는 재생하기 위해 기록/재생 상태를 활성 상태로 진입하기 위해, 도 3의 프로세서(204) 내의 부분적인 재생 제어기(206)에 의해 수행될 수 있는 예시적인 프로세스(400)를 예시하는 흐름도이다. 이와 관련하여, 도 4에 예시된 바와 같이, 프로세스(400)는, 부분적인 재생 제어기(206)가 잠재적인 아키텍처 상태 수정과 연관된 명령(302), 또는 명령 블록(308)의 실행 동안에 프로세서(204)에서의 제외의 발생을 검출함으로써 시작된다(도 4의 블록(402)). 도 3의 예에 도시된 바와 같이, 부분적인 재생 제어기(206)는 잠재적인 아키텍처 상태 수정과 연관된 명령(302), 또는 명령 블록(308)의 실행 동안에 프로세서(204)에서의 제외의 발생을 검출하도록 구성된 검출 회로(332)를 포함할 수 있다. 프로세서(204)에 의해 실행되는 명령들(302)은, 부분적인 재생 제어기(206)의 검출 회로(332)가 명령(302)이 잠재적인 아키텍처 상태 수정과 연관되는지를 검출하도록 허용하는 로드/저장 유닛(320)에 제공된다.

[0028] 계속해서 도 4를 참조하면, 잠재적인 아키텍처 상태 수정과 연관된 명령(302)의 검출 또는 명령 블록(308)의 실행 동안에 제외의 발생에 대한 응답으로, 부분적인 재생 제어기(206)는 명령 블록(308)에 대해 기록/재생 상태(330)를 활성 상태로 설정하도록 구성된다(도 4의 블록(404)). 예컨대, 잠재적인 아키텍처 상태 수정과 연관된 명령(302) 또는 프로세서(204)에서의 제외의 발생의 검출 전에, 기록/재생 상태(330)는 유휴 실행 상태로 설정될 수 있다. 기록/재생 상태(330)가 명령 블록(308)에 대해 활성 상태인 것에 대한 응답으로, 명령 블록(308)은 프로세서(204)에 의해 재실행된다. 활성 상태가 제외의 발생에 대한 응답으로 기록/재생 상태(330)로 설정되면, 제외가 CPU(202)에 의해(예컨대, CPU(202) 내의 운영 시스템에 의해)(도 2) 해결된 후에 명령 블록(308)이 재실행된다.

[0029] [0036] 기록/재생 상태(330)가 활성 상태인 것에 대한 응답으로 명령 블록(308)이 재실행될 때, 부분적인 재생 제어기(206) 내의 기록/재생 회로(334)는, 다음 로드/저장 명령(302L)이 실행되기 전에, 명령 블록(308)에서 실행될 각각의 다음 로드/저장 명령(302L)에 대응하는 기록/재생 로그 파일(338)에서 엔트리 상태(336)를 검사한다(도 4의 블록(406)). 기록/재생 회로(334)는 다음 로드/저장 명령(302L)에 대해 이전에 생성된 데이터가 기록되었는지를 결정하기 위해 엔트리 상태(336)를 검사하는데, 이는, 잠재적인 아키텍처 상태 수정이 검출되었거나 명령 블록(308)의 실행 동안에 제외가 발생되기 전에, 검사되는 다음 로드/저장 명령(302L)이 명령 블록(308)의 이전 실행 동안에 이미 실행되었음을 의미한다(도 4의 블록(406)). 실행될 다음 로드/저장 명령(302L)이 이미 이전에 실행되었음을 엔트리 상태(336)가 표시하면, 다음 로드/저장 명령(302L)에 대해 생성된 결과들은, 이것이 명령 블록(308) 내의 이러한 다음 로드/저장 명령(302L)의 실행의 첫 번째 인스턴스이기 때문에, 기록/재생 로그 파일(338)에 기록될 것이다. 그러나, 다음 로드/저장 명령(302L)이 명령 블록(308)의 이전 실행 인스턴스에서 이미 실행되지 않았음을 엔트리 상태(336)가 표시하면, 기록/재생 회로(334)는 기록/재생 상태(330)가 활성 상태인 것에 대한 응답으로 기록/재생 로그 파일(338) 내의 실행된 다음 로드/저장 명령(302L)의 생성된 데이터를 기록하도록 구성된다(도 4의 블록(408)). 이는, 명령 블록(308)이 다시 재실행되어야 하는 경우에, 실행된 다음 로드/저장 명령(302L)의 생성된 데이터가 재생될 기록/재생 로그 파일(338)에서 이용 가능하게 되도록 하기 위해서이다. 이와 관련하여, 기록/재생 회로(334)는 또한 기록/재생 로그 파일(338)의 다음 로드/저장 명령(302L)을 위해 기록된 이전에 생성된 데이터를 사용하여 다음 로드/저장 명령(302L)을 실행하도록 구성된다(도 4의 블록(410)).

[0030] [0037] 도 2의 CPU(202)의 프로세서들(204(1)-204(P))에 제공될 수 있는 부분적인 재생 제어기(206)에 대한 추가의 논의를 가능하게 하기 위해, 도 5 및 6은 명령 블록(308)의 실행 및 재실행 동안에 도 3의 부분적인 재생 제어기(206)의 추가의 예시적인 논의를 가능하게 하도록 제공된다. 이와 관련하여, 도 5는 잠재적인 아키텍처 상태 수정과 연관된 명령들을 포함하는, 도 3의 프로세서(204)에 의해 실행될 수 있는 예시적인 명령 블록(308(A))을 예시한다. 도 6은, 잠재적인 아키텍처 상태 수정과 연관된 명령 블록(308(A)) 내의 명령(502)을 검출하는 것에 대한 응답으로, 도 5의 명령 블록(308(A))의 재실행 동안에 로드/저장 명령들(302L)의 결과들을 기록 및 재생하기 위해 부분적인 재생 제어기(206)에 의해 수행될 수 있는 예시적인 프로세스(600)의 더 세부적인 사항을 예시하는 흐름도이다. 도 6의 프로세스(600)에 대한 참조는 도 5의 명령 블록(308(A))과 관련하여 이루어질 것이다.

[0031]

[0038] 이와 관련하여, 도 5를 참조하면, 명령 블록(308(A))은, 기록/재생 상태(330)가 유휴로 설정된 새로운 인스턴스에서 실행하기 위해 도 3의 프로세서(204)에 의해 폐침된다(도 6의 블록(602)). 이러한 예에서 I/O 디바이스 로드/저장 명령인 제1 명령(502(1))이 도 3의 로드/저장 유닛(320)에 제공될 때, 부분적인 재생 제어기(206)의 검출 회로(332)는, 제1 명령(502(1))이 잠재적인 아키텍처 상태 수정과 연관되는지를 검출한다(도 6의 블록(604)). 이러한 예에서, 로딩될 데이터의 소스가 I/O 디바이스로부터 오기 때문에, 제1 명령(502(1))은 잠재적인 아키텍처 상태 수정과 연관된 잠재적인 부작용을 갖는다. 따라서, 제1 명령(502(1))의 실행 후에 제외가 발생하였다면, I/O 디바이스로부터 팝핑된(popped) 데이터는 I/O 디바이스 큐에서 더 이상 유지되지 않을 수 있고, 따라서 제1 명령 블록(502(1))이 명령 블록(308(A))의 재실행 동안에 재실행되기 전에 재생 가능하지 않을 수 있다. 따라서, 이러한 경우에, 부분적인 재생 제어기(206)는 제1 명령(502(1))을 실행하지 않고서 제1 명령(502(1))에 대한 잠재적인 부작용의 검출에 대한 응답으로 명령 블록(308(A))에 대해 기록/재생 상태(330)를 활성으로 설정한다. 이는, 결과들의 부작용이 명령 블록(308(A))의 재실행 동안에 제1 명령(502(1))의 다음 실행 인스턴스에서 재생 불가하게 하는 제1 명령(502(1))의 실행으로부터의 결과들이 생성되지 않도록 하기 위해서이다(도 6의 블록(606)). 이후에, 프로세서(204)는 명령 블록(308(A))이 재실행되게 한다.

[0032]

[0039] 도 5의 명령 블록(308(A))의 재실행 동안에, 제1 명령(502(1))은 실행을 위해 도 3의 프로세서(204) 내의 로드/저장 유닛(320)에 다시 제공된다. 부분적인 재생 제어기(206) 내의 기록/재생 회로(334)는, 이전에 생성된 데이터가 제1 명령(502(1))에 대해 기록되었는지를 결정하기 위해 기록/재생 로그 파일(338) 내의 엔트리(340(1)-340(E))의 엔트리 상태(336)를 조사한다(도 6의 블록(608)). 이러한 예에서, 제1 명령(502(1))이 잠재적인 부작용의 검출로 인해 명령 블록(308(A))의 초기 실행 동안 실행되지 않았기 때문에, 제1 명령(502(1))에 대해 이전에 생성된 데이터가 기록되지 않았다. 따라서, 부분적인 재생 제어기(206)의 기록/재생 회로(334)는, 이를테면, 제외로 인해 명령 블록(308(A))이 다시 재실행되는 경우에, 생성된 결과들(504(1))이 재생될 수 있도록, 로드/저장 유닛(320)이 기록/재생 로그 파일(338) 내의 제1 명령(502(1))을 실행한 것의 결과로서 제1 명령(502(1))으로부터 생성된 결과들(504(1))을 기록한다(도 6의 블록(610)). 이후에, 프로세서(204)는 부작용을 고려하지 않고서 제2 명령(502(2))을 실행하며, 제2 명령(502(2))은 로드/저장 명령이 아니다. 그러나, 도 5의 예시적인 명령 블록(308(A))에서의 제3 명령(502(3))은 잠재적인 부작용을 갖는 로드/저장 동작이다. 그러나, 기록/재생 상태(330)가 이미 활성화되어 있기 때문에-이는 부분적인 재생 제어기(206)가 실행된 로드/저장 명령들(302L)로부터 생성된 결과들(504(1)-504(Q))을 기록하고 있다는 것을 의미함-, 명령 블록(308(A))이 재실행되지 않는다. 제3 명령(502(3))에서 글로벌 레지스터(r3)로부터 획득된 생성된 결과들(504(3))은 기록/재생 로그 파일(338)에 기록된다. 명령들(502(4)-502(Q))의 실행은, 로드/저장 명령들(302L)의 생성된 결과들(504(1)-504(Q))이 부분적인 재생 제어기(206)에 의해 기록/재생 로그 파일(338)에 기록되는 것을 계속한다(도 6의 블록들(608, 610)).

[0033]

[0040] 그러나, 도 5의 예에 도시된 바와 같이, 제외(508)는 명령(502(4))과 명령(502(Q-2)) 사이에서 발생하였다. 제외(508)가 해결된 후에, 명령 블록(308(A))은 프로세서(204)에서 재실행된다. 그러나, 명령들(502(1)-502(4))이 명령 블록(308(A))의 이전 실행 인스턴스에서 이미 실행되었기 때문에, 부분적인 재생 제어기(206) 내의 기록/재생 회로(334)는 생성된 결과들(504(1), 504(3)-504(4))이 기록/재생 로그 파일(338)에 기록되었는지를 결정하기 위해 기록/재생 로그 파일(338)을 검사하도록 구성된다(도 6의 블록(608)). 이러한 예에서, 생성된 결과들(504(1), 504(3)-504(4))이 기록/재생 로그 파일(338)에 기록되기 때문에, 기록/재생 회로(334)는, 이러한 예에서 명령 블록(308(A))의 이전 재실행 동안에 기록/재생 로그 파일(338)에 기록된 이전에 생성된 결과들(504(1), 504(3)-504(4))을 사용하여 이전에 생성된 결과들(504(1)-504(Q))을 실행 또는 "재생"하도록 구성된다(도 6의 블록(612)). 따라서, 제외(508)가 발생한 후에 명령들(502(1), 502(3)-502(4))에 관련하여 제외 또는 부작용이 발생하였다면, 이전에 생성된 결과들(504(1), 504(3)-504(4))은, 명령 블록(308(A))의 재실행이 정확한 결과들 및 생산된 값들을 생성하도록, 기록/재생 로그 파일(338)에 보존되었다.

[0034]

[0041] 이러한 예에서, 제외(508)가 발생하기 전에 명령 블록(308(A))의 이전 재실행 동안에 명령들(502(Q-2)-502(Q))이 실행되지 않았기 때문에, 부분적인 재생 제어기(206)는, 명령들(502(Q-2)-502(Q))에 대해 생성된 결과들(504(Q-2), 504(Q))이 기록/재생 로그 파일(338)에 기록되었는지를 결정하기 위해, 기록/재생 로그 파일(338)을 검사하도록 구성된다(도 6의 블록(608)). 이러한 예에서, 이들은 제외(508)가 발생하기 전에 기록되지 않았기 때문에, 기록/재생 회로(334)는, 명령들(502(Q-2)-502(Q))을 재생하는 것과는 대조적으로, 명령들(502(Q-2), 502(Q))에 대해 생성된 결과들(504(Q-2), 504(Q))을 기록/재생 로그 파일(338)에 기록하도록 구성된다(도 6의 블록(610)). 따라서, 다른 제외(508)가 발생하면, 명령들(502(Q-2), 502(Q))에 대해 생성된 결과들(504(Q-2), 504(Q))은 기록/재생 로그 파일(338)로부터 재생될 수 있다. 도 5의 명령 블록(308(A))에서 모

는 명령들(502(1)-502(Q))이 실행된 후에(도 6의 블록(614)), 실행된 명령들(502(1)-502(Q))로부터 발생한 외부 생성된 값들이 커밋된다(도 6의 블록(616)).

[0042] 이것은 어떤 타입들의 로드/저장 명령들(302L)이 검출 회로(332)에 의해 잠재적인 부작용을 갖는 것으로 검출되도록 구성되는지에 대한 설계 선택이라는 것이 유의된다. 예컨대, 검출 회로(332)는, 로드/저장 명령(302L)의 타입을 고려하지 않고서 실행된 명령 블록(308)의 모든 각각의 로드/저장 명령(302L)을 잠재적인 부작용을 갖는 것으로서 검출하도록 구성될 수 있다. 예컨대, 프로세서(204)가 다중-스레드 코드를 실행하는 경우, 잠재적으로 임의의 로드 명령이 생성 불가한 결과들을 발생시킬 수 있는데, 왜냐하면 제외가 발생한 후에, 다른 스레드가 그러한 로드 명령의 소스 위치에서 데이터를 변경할 수 있기 때문이다. 그러나, 실행된 명령 블록(308) 내의 모든 각각의 타입의 로드/저장 명령(302L)에서 잠재적인 부작용을 검출하는 것은 프로세서(204)의 성능을 감소시킬 수 있는데, 왜냐하면 실행된 명령 블록(308)이 더 자주 재실행될 수 있기 때문이다. 기록/재생 상태(330)는, 임의의 타입의 로드/저장 명령(302L)을 갖는 실행된 명령 블록(308)에 기반하여 활성으로 설정될 것이다. 이는, 부분적인 재생 제어기(206)가 디버깅 모드 동안에 로드/저장 명령들(302L)로부터 모든 데이터의 상태를 검토하기 위한 능력을 제공할 것이기 때문에, 디버깅 모드에서 또는 디버거 브레이크포인트에 대한 응답으로 유용할 수 있다. 그러나, 다른 예로서, 정상 또는 비-디버깅 모드 동안에, 검출 회로(332)는, 다른 재생 가능성 이슈들을 고려하지 않고서 I/O 동작들을 수반하는 잠재적인 부작용들을 갖는 것으로서, 명령 블록(308) 내의 특정 타입들의 로드/저장 명령들(302L)만을 검출하도록 구성될 수 있다. 예컨대, 검출 회로(332)는 프로세서(204) 외부의 위치들 또는 디바이스들로부터 또는 이들로 데이터를 로딩 또는 저장하는, 명령 블록(308) 내의 로드/저장 명령들(302L)만을 검출하도록 구성될 수 있다.

[0043] 부분적인 재생 제어기(206)의 동작은, 명령 블록(308)이 재실행 상태에 기반한 동작들에 따라 여러 번 재실행될 수 있기 때문에, 상태 머신으로서 구현하기에 매우 적합할 수 있다. 이와 관련하여, 도 7은 명령 블록의 재실행 동안에 로드/저장 명령들의 결과들을 기록 및/또는 재생하기 위해 도 3의 프로세서(204)의 부분적인 재생 제어기(206)에 의해 사용될 수 있는 상태 머신(700)이다. 이와 관련하여, 도 7을 참조하면, 프로세서(204)에서 명령 블록(308)의 새로운 실행 인스턴스(도 7의 702)에 대한 응답으로, 프로세서(204)에 의해 액세스 가능한 기록/재생 상태(330)는 유휴 상태("유휴")로 설정되는데, 이는 기록/재생 동작이 명령 블록(308)에 대해 발생하지 않음을 의미한다. 그러한 스레드에 대한 아키텍처 상태 또는 제외 또는 인터럽트는 명령 블록(308) 경계에 대응한다. 프로세서(204)는 유휴 상태에서 명령 블록(308) 내의 명령들(302)을 실행한다. 명령 블록(308) 내의 모든 명령들(302)이 명령 블록(308)의 초기 실행 동안에 잠재적인 아키텍처 상태 수정을 갖는 것으로 검출되지 않고, 어떠한 제외도 발생하지 않는다면, 명령들(302)의 실행은 결국 외부 생성된 결과들이 커밋되는 것으로 완전히 완료될 것이다(도 7에서 704).

[0044] 계속해서 도 7을 참조하면, 일단 검출 회로(332)가 실행될 잠재적인 아키텍처 상태 수정과 연관된 명령(302)을 검출하면(도 7의 706), 부분적인 재생 제어기(206)는 기록/재생 상태(330)를 활성 상태("활성")로 변경한다. 이어서, 프로세서(204)는 명령 블록(308)이 시작 명령(302)으로부터 재실행되게 한다. 이것은, 명령 블록(308)이 재실행될 수 있고 잠재적인 부작용들을 갖는 로드/저장 명령들(302L)에 대해 생성된 결과들이 기록/재생 로그 파일(338)에 기록될 수 있도록 한다. 아래에서 논의되는 바와 같이, 스레드가 활성일 때, 이는 부분적인 재생 제어기(206)가 기록 및/또는 재생되고 있다는 것을 의미한다. PBR 상태를 기록한다는 것은 시스템 레지스터들의 로드들 및 저장들 + 판독들 및 기록들이 나중 재생을 위해 기록된다는 것을 의미한다.

[0045] 활성 상태에서, 부분적인 재생 제어기(206)는, 이전에 논의된 바와 같이, 기록/재생 로그 파일(338)에 액세스하도록 구성된다. 부분적인 재생 제어기(206)는, 명령 블록(308)의 재실행 동안에, 실행된 로드/저장 명령들(302L)로부터의 생성된 결과들로 기록/재생 로그 파일(338)을 업데이트 또는 기록하도록 구성된다. 이러한 방식으로, 잠재적인 아키텍처 상태 수정(예컨대, 잠재적인 부작용)과 연관된 로드/저장 명령(302L)의 실행 후에 제외가 발생하면(도 7의 708), 생성된 결과는 로드/저장 유닛(320)에 의해 기록/재생 로그 파일(338)에 저장된다. 명령 블록(308)의 재실행 동안에 제외가 발생하면, 부분적인 재생 제어기(206)는 CPU(202) 내의 운영 시스템에서의 제외 핸들링에 의해 제외가 해결되도록 기록/재생 상태(330)를 잠금 상태(locked state)(("잠금"))로 설정한다. 제외가 해결된 후에, 명령 블록(308)이 재실행되도록, 기록/재생 상태(330)는 다시 활성 상태("활성")로 설정된다(도 7의 710). 위에서 이전에 논의된 바와 같이, 생성된 결과들은 제외가 해결된 후에 명령 블록(308)의 재실행 동안에 재생될 수 있어서, 이전에 기록된 생성된 결과들이, 예들로서, 이를테면, 다른 스레드 또는 I/O 디바이스로부터 팝핑된 데이터로부터의 변화로 인해 상이해질 경우에, 명령 블록(308)은 그러한 기록된 생성된 결과들로 성공적으로 재실행될 수 있다.

[0046] 예컨대, 기록/재생 로그 파일(338) 내의 실행된 로드/저장 명령(302L)에 대해 기록된 엔트리 상태(330)

6)는, 비제한적인 예들로서, 실행되는 로드/저장 명령(302L)의 타입에 의존하여 널(null), 로드(정상), 저장(정상), 어널링(annulled), 로드-커밋(부작용), 저장-커밋(부작용), 팬독 시스템 레지스터 및 기록 시스템 레지스터-커밋일 수 있다. 실행된 로드/저장 명령(302L)에 대해 기록된 가상 어드레스(VA)(342)는, 프로그램을 디버깅하는데 유용한 로드/저장 명령(302L)에 의해 액세스되는 메모리 또는 I/O 디바이스의 어드레스이다. 실행된 로드/저장 명령(302L)에 대해 기록된 데이터(344)는 로드/저장 명령(302L)을 실행한 결과로서 생성된 결과이다.

[0040] 일 예에서, 기록/재생 상태(330)가 활성 상태("활성")에 있을 때 제외가 발생하는 경우에, 제외의 발생 전에 실행된 명령들(302)로부터의 어떠한 생성된 결과들도 커밋되지 않는다는 것이 유의된다. 명령 블록(308)은 일단 제외가 잠금 상태("잠금")로부터 해결될 때 재실행된다(도 7의 710). 그러나, 대안적인 양상에서, 부분적인 재생 제어기(206)는, 제외의 발생 전에 실행된 명령 블록(308) 내의 명령들(302)로부터 생성된 결과들을 커밋하도록 구성될 수 있다. 이러한 경우에, 제외가 해결되고 기록/재생 상태(330)가 활성 상태("활성")로 되돌아갈 때, 일단 남아있는 명령들(302)이 재실행되면, 이전에 실행하지 않은 명령들(302)로부터의 생성된 결과들만이 커밋된다.

[0041] 계속해서 도 7을 참조하면, 기록/재생 상태(330)는, 명령 블록(308) 내의 모든 명령들(302)이 실행될 때까지 활성 상태("활성")로 유지된다. 이후에, 명령 블록(308) 내의 실행된 명령들(302)로부터의 외부 생성된 결과들은 커밋된다(또는 부분적인 커밋 기능의 경우에, 이전에 커밋되지 않은 명령들(302)로부터 생성된 결과들이 커밋됨)(도 7의 712). 이전에 논의된 바와 같이, 부분적인 커밋 동작이 사용되는 일 예에서, 부분적인 커밋 동작의 부분으로서 명령 블록(308)의 이전 실행 동안에 커밋된 로드/저장 명령들(302L)로부터 생성된 데이터는, 명령 블록(308)이 완전히 실행될 때 완전 커밋의 후속하는 부분적인 재커밋 동작 동안에 재커밋되지 않는다. 부분적인 재생 제어기(206)는 다음 명령 블록(308)을 실행하기 위해 기록/재생 상태(330)를 유휴 상태("유휴")로 다시 설정한다. 기록/재생 상태(330)가 유휴 상태("유휴")일 때 제외가 발생하면, CPU(202) 내의 운영 시스템은 현재 아키텍처 상태를 저장하고, 어떠한 추가의 명령들(302)이 명령 블록(308)에서 실행되지 않도록 제외를 해결하기 위해 잠금 상태 "잠금"으로 스위칭한다(도 7의 714). 일단 잠금 상태 "잠금"에서 제외가 해결되면, 운영 시스템은 제외가 발생하기 전에 존재한 아키텍처 상태를 복원하고, 기록/재생 상태(330)를 유휴 상태("유휴")로 다시 스위칭한다(도 7의 716).

[0042] 요약하면, 도 3의 프로세서(204) 내의 부분적인 재생 제어기(206)에 의해 제공되는 PBR 모델은, 부작용들을 갖는 동작들을 수행하는 명령 블록(308)이 나중에 임의의 종류의 제외를 취할 수 있을 때만 필요로 된다. 그러나, 임의의 정확한 제외가 블록-원자 실행 모델에 대해 프로그램 종결 또는 디버거로의 전송을 발생시키는 것으로 예상될 때, PBR 모델은 또한 디버깅을 향상시키는 데 사용될 수 있다. 성능상의 이유들로, PBR 모델은 슈퍼바이저 서비스(supervisor service)가 인보크되게 하고 이어서 정상 실행이 재개되게 할 수 있는 비-에러 제외들(예컨대, 페이지 결함)에 대해 선택적으로 관여될 수 없다.

[0043] 부분적인 재생 제어기(206)를 사용함으로써 명령 블록(308)의 아키텍처 상태가 보존될 수 있는 예시적인 제외 카테고리들의 요약이 아래에 도시된다. 그러나, PBR 모델이 부분적인 재생 제어기(206)에 의해 인에이블되는 경우에 현재 명령 블록(308)이 실행되면, 인터럽트는 물론 정확한 부분적인 블록 상태로 취해질 수 있다. 이전에 논의되고 아래 표에 도시된 바와 같이, 제외가 발생하기 전에, 명령 블록에 부작용들이 발생하거나 생성된 데이터를 저장하였다면, 부분적인 재생 제어기는, 제외가 발생하기 전에, 명령 블록에서 실행된 명령들로부터 생성된 결과들을 폐기하도록 구성될 수 있다. 프로세서가 제외로부터 복구한 후, 이어서 부분적인 재생 제어기는, 로드/저장 명령들의 결과들을 기록 및/또는 재생하기 위해, 명령 블록의 시작 명령으로부터 다시 실행을 시작하도록 구성될 것이다. 대안적으로, 부분적인 재생 제어기는, 제외가 발생하기 전에 명령 블록에서 실행된 명령들로부터 생성된 결과들을 커밋하도록 구성될 수 있다. 프로세서가 제외로부터 복구한 후, 이어서 부분적인 재생 제어기는, 재실행 동안에 명령 블록에서 이전에 실행되지 않은 로드/저장 명령들의 결과들을 기록 및/또는 재생하기 위해 실행되지 않은 명령 블록 내의 다음 명령으로부터 시작하도록 구성될 것이다.

표 1

정확성	발생	부분적인 블록 재생	블록은 수행된 부작용들 또는 저장들을 가짐	아키텍처 상태	재생 가능성
정확함	블록	N/A	아니오	블록의 시작	예
	명령	아니오	아니오	블록의 시작	아니오
		예	아니오	블록의 시작	예

		예(폐기된 정상 저장들)	GPR/정상 메모리: 블록의 시작 부작용들: 부작용들 후	예
		예(실행 포인트로 부분적으로 커밋된 정상 저장들)	GPR: 블록의 시작 정상 메모리 또는 부작용들: 제외의 포인트에서	예

[0045]

[0051] 추가의 양상들에서, 잠재적인 아키텍처 상태 수정과 연관된 명령의 검출, 또는 프로세서에서의 제외의 발생에 대한 응답으로, 명령 블록에 대해 기록/재생 로그 파일을 저장하기 위한 수단을 활성 상태로 설정하기 위한 수단을 포함하는, 프로세서에서 실행되는 명령 블록의 실행 재생을 제어하기 위한 부분적인 재생 제어기가 제공될 수 있다. 예컨대, 도 3의 부분적인 재생 제어기(206), 검출 회로(332) 또는 기록/재생 회로(334)는 그러한 설정하기 위한 수단의 예들이고, 여기서 부분적인 재생 제어기(206)는, 잠재적인 아키텍처 상태 수정과 연관된 명령의 검출, 또는 프로세서(204)에서 제외의 발생에 대한 응답으로, 기록/재생 상태(330)를 활성 상태로 설정하도록 구성된다. 부분적인 재생 제어기(206)는 또한, 명령 블록에 대해 기록/재생 로그 파일을 저장하기 위한 수단을 활성 상태로 설정하는 것에 대한 응답으로, 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록되는지 결정하기 위해 명령 블록에서 실행될 다음 로드/저장 명령에 대응하는 기록/재생 로그 파일을 저장하기 위한 수단에서 엔트리 상태를 검사하는 수단을 포함할 수 있다. 예컨대, 기록/재생 로그 파일(338)에 액세스함으로써, 도 3의 부분적인 재생 제어기(206) 또는 기록/재생 회로(334)에 의해 엔트리 상태를 검사하기 위한 이러한 수단이 제공될 수 있다. 부분적인 재생 제어기(206)는 또한, 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록/재생 로그 파일을 저장하기 위한 수단에 기록되지 않는 것에 대한 응답으로, 기록/재생 로그 파일을 저장하는 수단에서 실행된 다음 로드/저장 명령의 생성된 데이터를 기록하는 수단을 포함할 수 있다. 예컨대, 기록/재생 로그 파일을 저장하기 위한 수단은, 다음 로드/저장 명령에 대해 기록/재생 로그 파일(338)에 기록되지 않는 이전에 생성된 데이터를 저장하기 위해, 도 3의 부분적인 재생 제어기(206) 또는 기록/재생 회로(334)에 의해 제공될 수 있다. 부분적인 재생 제어기(206)는 또한, 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록/재생 로그 파일을 저장하기 위한 수단에 기록되는 것에 대한 응답으로, 기록/재생 로그 파일을 저장하기 위한 수단에서 다음 로드/저장 명령에 대해 기록된 이전에 생성된 데이터를 사용하여 다음 로드/저장 명령을 실행하기 위한 수단을 포함할 수 있다. 이전에 생성된 데이터를 사용하여 다음 로드/저장 명령을 실행하기 위한 수단은 도 3의 부분적인 재생 제어기(206) 또는 기록/재생 회로(334)에 의해 제공될 수 있다.

[0046]

[0052] 예컨대, 도 3의 프로세서(204) 내의 부분적인 재생 제어기(206)와 같이, 그리고 본원에 개시된 예들 중 임의의 것에 따른, 잠재적인 아키텍처 상태 수정과 연관된 명령 블록 내의 명령을 검출하는 것에 대한 응답으로, 그리고/또는 명령 블록 내의 명령들의 실행 동안에, 제외의 발생에 대한 응답으로, 명령 블록의 재실행 동안에, 로드/저장 명령들의 결과들을 기록 및/또는 재생하도록 구성된 부분적인 재생 제어기를 포함하는 프로세서는 임의의 프로세서-기반 디바이스에 제공되거나 이에 통합될 수 있다. 예들은, 제한없이, 셋톱 박스, 엔터테인먼트 유닛, 내비게이션 디바이스, 통신 디바이스, 고정형 위치 데이터 유닛, 모바일 위치 데이터 유닛, 모바일 폰, 셀룰러 폰, 스마트 폰, 태블릿, 패블릿, 컴퓨터, 휴대용 컴퓨터, 데스크탑 컴퓨터, PDA(personal digital assistant), 모니터, 컴퓨터 모니터, 텔레비전, 퓨너, 라디오, 위성 라디오, 음악 플레이어, 디지털 음악 플레이어, 휴대용 음악 플레이어, 디지털 비디오 플레이어, 비디오 플레이어, DVD(digital video disc) 플레이어, 휴대용 디지털 비디오 플레이어 및 자동차를 포함한다.

[0047]

[0053] 이와 관련하여, 도 8은 하나 또는 그 초과의 프로세서들(804)을 포함하는 CPU(802)를 포함하는 프로세서-기반 시스템(800)의 예를 예시한다. 프로세서(들)(804) 각각은, 명령 블록 내의 명령들의 실행 동안에 잠재적인 부작용들을 검출하는 것에 대한 응답으로, 명령 블록의 재실행 동안에 로드/저장 명령들의 결과들을 기록 및/또는 재생하도록 구성된 부분적인 재생 제어기(806)를 포함할 수 있다. 부분적인 재생 제어기(806)는 예로서 도 3의 부분적인 재생 제어기(206)일 수 있다. CPU(802)는 일시적으로 저장된 데이터에 신속하게 액세스하기 위해 프로세서(들)(804)에 커플링된 캐시 메모리(808)를 가질 수 있다. CPU(802)는 시스템 버스(810)에 커플링되고, 프로세서-기반 시스템(800)에 포함되는 주변 디바이스들을 상호커플링할 수 있다. CPU(802) 내의 프로세서(들)(804)는 시스템 버스(810)를 통해 어드레스, 제어, 및 데이터 정보를 교환함으로써 이러한 다른 디바이스들 간의 통신을 지원한다.

이스들과 통신할 수 있다. 도 8에 예시되지 않았지만, 다수의 시스템 버스들(810)이 제공될 수 있고, 각각의 시스템 버스(810)는 상이한 패브릭을 구성한다. 예컨대, CPU(802)는 버스 트랜잭션 요청들을 슬레이브 디바이스의 예로서 메모리 시스템(814) 내의 메모리 제어기(812)에 통신할 수 있다. 이러한 예에서, 메모리 제어기(812)는 메모리 시스템(814) 내의 메모리 어레이(816)에 메모리 액세스 동작들을 제공하도록 구성된다.

[0048] 다른 디바이스들이 시스템 버스(810)에 접속될 수 있다. 도 8에 예시된 바와 같이, 이러한 디바이스들은 메모리 시스템(814), 하나 또는 그 초과의 입력 디바이스들(818), 하나 또는 그 초과의 출력 디바이스들(820), 하나 또는 그 초과의 네트워크 인터페이스 디바이스들(822) 및 하나 또는 그 초과의 디스플레이 제어기들(824)을 예들로서 포함할 수 있다. 입력 디바이스(들)(818)는 입력 키들, 스위치들, 음성 프로세서들 등을 포함하는(그러나 이에 제한되는 것은 아님) 임의의 타입의 입력 디바이스를 포함할 수 있다. 출력 디바이스(들)(820)는 오디오, 비디오, 다른 시각적 표시자들 등을 포함하는(그러나 이에 제한되는 것은 아님) 임의의 타입의 출력 디바이스를 포함할 수 있다. 네트워크 인터페이스 디바이스(들)(822)는 네트워크(826)로 그리고 네트워크(826)로부터의 데이터의 교환을 가능하게 하도록 구성된 임의의 디바이스들일 수 있다. 네트워크(826)는 유선 또는 무선 네트워크, 사설 또는 공공 네트워크, 근거리 네트워크(LAN: local area network), 무선 근거리 네트워크(WLAN: wireless local area network), 광역 네트워크(WAN: wide area network), BLUETOOTH™ 네트워크 및 인터넷을 포함하는(그러나 이에 제한되는 것은 아님) 임의의 타입의 네트워크일 수 있다. 네트워크 인터페이스 디바이스(들)(822)는 원하는 임의의 타입의 통신 프로토콜을 지원하도록 구성될 수 있다.

[0049] CPU(802)는 또한 하나 또는 그 초과의 디스플레이들(828)에 전송되는 정보를 제어하기 위해 시스템 버스(810)를 통해 디스플레이 제어기(들)(824)에 액세스하도록 구성될 수 있다. 디스플레이(들)(828)는 음극선관(CRT), 액정 디스플레이(LCD), 플라즈마 디스플레이 등을 포함하는 임의의 타입의 디스플레이를 포함할 수 있지만, 이에 제한되지 않는다. 디스플레이 제어기(들)(824)는 하나 또는 그 초과의 비디오 프로세서들(830)을 통해 디스플레이될 정보를 디스플레이(들)(828)로 전송하고, 하나 또는 그 초과의 비디오 프로세서들(830)은 디스플레이될 정보를 디스플레이(들)(828)에 적합한 포맷으로 프로세싱한다.

[0050] 명령 블록 내의 명령들의 실행 동안에 잠재적인 부작용을 검출하는 것에 대한 응답으로, 명령 블록의 재실행 동안에 로드/저장 명령들의 결과들을 기록 및/또는 재생하도록 구성된 부분적인 재생 제어기는 또한 소프트웨어-기반 시스템에 제공될 수 있다. 부분적인 재생 제어기는, 소프트웨어 명령들 없이 부분적인 재생 제어기의 기능들을 제공하는 하드웨어-전용 회로에서 구현될 필요는 없다. 도 3의 부분적인 재생 제어기(206)와 같은 부분적인 재생 제어기는 컴퓨터 실행 가능 명령들이 저장된 비일시적인 컴퓨터 판독 가능 매체에 제공될 수 있고, 컴퓨터 실행 가능 명령들은, 도 2의 프로세서(204(1)-204(P)) 등과 같은 프로세서에 의해 실행될 때, 프로세서로 하여금, 잠재적인 아키텍처 상태 수정과 연관된 명령의 검출, 또는 프로세서에서의 제외의 발생에 대한 응답으로, 명령 블록에 대해 기록/재생 상태를 활성 상태로 설정하게 한다. 기록/재생 상태가 명령 블록에 대해 활성 상태인 것에 대한 응답으로, 프로세서에 의해 실행될 때, 프로세서로 하여금, 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록되는지를 결정하기 위해 명령 블록에서 실행될 다음 로드/저장 명령에 대응하는 기록/재생 로그 파일 내의 엔트리 상태를 검사하게 하는 컴퓨터 실행 가능 명령들은, 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록/재생 로그 파일에 기록되지 않은 것에 대한 응답으로, 실행된 다음 로드/저장 명령의 생성된 데이터를 기록/재생 로그 파일에 기록하고, 그리고 이전에 생성된 데이터가 다음 로드/저장 명령에 대해 기록된 것에 대한 응답으로, 기록/재생 로그 파일에 다음 로드/저장 명령을 실행한다.

[0051] "컴퓨터-판독 가능 매체"는 하나 또는 그 초과의 명령들의 세트들을 저장하는 단일 매체 또는 다수의 매체들(예컨대, 중앙집중형 또는 분산형 데이터베이스 및/또는 연관된 캐시들 및 서버들)을 포함하는 것으로 여겨져야 한다. "컴퓨터-판독 가능 매체"라는 용어는 또한, 프로세서에 의해 실행하기 위한 명령들의 세트를 저장, 인코딩 또는 반송할 수 있고 프로세서로 하여금 본원에 개시된 양상들의 방법들 중 임의의 하나 또는 그 초과의 방법들을 수행하게 하는 임의의 매체를 포함하는 것으로 여겨져야 한다. 따라서, "컴퓨터-판독 가능 매체"라는 용어는 고체-상태 메모리들, 광학 매체들 및 자기 매체들(그러나, 이들로 제한되지는 않음)을 포함하는 것으로 여겨져야 한다.

[0052] 당업자들은 추가로, 본원에 개시된 양상들과 관련하여 설명된 다양한 예시적인 로직 블록들, 모듈들, 회로들 및 알고리즘들이 메모리에 또는 다른 컴퓨터 판독 가능 매체에 저장되어 프로세서 또는 다른 처리 디바이스에 의해 실행되는 명령들, 전자 하드웨어, 또는 이 둘의 결합들로서 구현될 수 있다고 인식할 것이다. 본

원에서 설명된 마스터 및 슬레이브 디바이스들은 예들로서 임의의 회로, 하드웨어 컴포넌트, 집적 회로(IC: integrated circuit) 또는 IC 칩에 이용될 수 있다. 본원에 개시된 메모리는 임의의 타입 및 크기의 메모리일 수 있고, 원하는 임의의 타입의 정보를 저장하도록 구성될 수 있다. 이러한 상호 호환성을 명확히 설명하기 위해, 각종 예시적인 컴포넌트들, 블록들, 모듈들, 회로들 및 단계들은 일반적으로 이들의 기능과 관련하여 위에서 설명되었다. 이러한 기능이 어떻게 구현되는지는 전체 시스템에 부과된 설계 제약들, 특정 애플리케이션 및 /또는 설계 선택들에 좌우된다. 당업자들은 설명된 기능을 특정 애플리케이션마다 다양한 방식들로 구현할 수 있지만, 이러한 구현 결정들이 본 개시내용의 범위를 벗어나게 하는 것으로 해석되지는 않아야 한다.

[0053]

[0059] 본원에 개시된 양상들과 관련하여 설명된 다양한 예시적인 로직 블록들, 모듈들 및 회로들은 프로세서, 디지털 신호 프로세서(DSP: Digital Signal Processor), 주문형 집적 회로(ASIC: Application Specific Integrated Circuit), 필드 프로그래밍 가능 게이트 어레이(FPGA: Field Programmable Gate Array) 또는 다른 프로그래밍 가능한 로직 디바이스, 이산 게이트 또는 트랜지스터 로직, 이산 하드웨어 컴포넌트들, 또는 본원에서 설명된 기능들을 수행하도록 설계된 이들의 임의의 결합으로 구현되거나 이들에 의해 수행될 수 있다. 프로세서는 마이크로프로세서일 수 있지만, 대안으로 프로세서는 임의의 종래 프로세서, 제어기, 마이크로컨트롤러 또는 상태 머신일 수 있다. 프로세서는 또한 컴퓨팅 디바이스들의 결합, 예컨대 DSP와 마이크로프로세서의 결합, 복수의 마이크로프로세서들, DSP 코어와 결합된 하나 또는 그 초과의 마이크로프로세서들, 또는 임의의 다른 이러한 구성으로서 구현될 수 있다.

[0054]

[0060] 본원에 개시된 양상들은 하드웨어로 그리고 하드웨어에 저장되는 명령들로 구현될 수 있으며, 예컨대, 랜덤 액세스 메모리(RAM: Random Access Memory), 플래시 메모리, 판독 전용 메모리(ROM: Read Only Memory), 전기적으로 프로그래밍 가능한 ROM(EPROM: Electrically Programmable ROM), 전기적으로 소거 가능한 프로그래밍 가능 ROM(EEPROM: Electrically Erasable Programmable ROM), 레지스터들, 하드디스크, 착탈식 디스크, CD-ROM, 또는 당해 기술분야에 공지된 임의의 다른 형태의 컴퓨터 판독 가능 매체에 상주할 수 있다. 예시적인 저장 매체는 프로세서가 저장 매체로부터 정보를 읽고 저장 매체에 정보를 기록할 수 있도록 프로세서에 연결된다. 대안적으로, 저장 매체는 프로세서에 통합될 수 있다. 프로세서 및 저장 매체는 ASIC에 상주할 수 있다. ASIC는 원격국에 상주할 수 있다. 대안적으로, 프로세서 및 저장 매체는 원격국, 기지국 또는 서버에 개별 컴포넌트들로서 상주할 수 있다.

[0055]

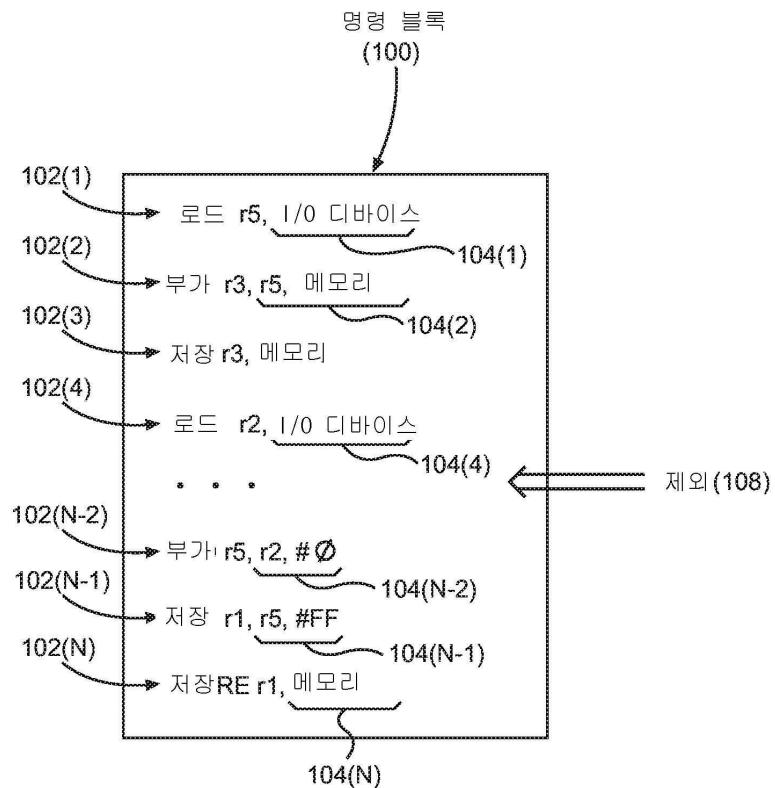
[0061] 본원의 예시적인 양상들 중 임의의 양상에서 설명된 동작 단계들은 예들 및 논의를 제공하기 위해 설명된다는 점이 또한 주목된다. 설명된 동작들은 예시된 순서들과는 다른 다수의 상이한 순서들로 수행될 수 있다. 게다가, 단일 동작 단계로 설명된 동작들은 실제로 다수의 서로 다른 단계들에서 수행될 수 있다. 추가로, 예시적인 양상들에서 논의된 하나 또는 그 초과의 동작 단계들이 결합될 수 있다. 흐름도 도면들에 예시된 동작 단계들은 당업자에게 쉽게 명백해지듯이 다수의 서로 다른 변형들이 가능할 수 있다고 이해되어야 한다. 당업자들은 또한 정보 및 신호들이 다양한 다른 기술들 및 기법들 중 임의의 것을 이용하여 표현될 수 있다고 이해할 것이다. 예컨대, 상기 설명 전반에 걸쳐 참조될 수 있는 데이터, 명령들, 커맨드들, 정보, 신호들, 비트들, 심벌들 및 칩들은 전압들, 전류들, 전자기파들, 자기 필드들 또는 자기 입자들, 광 필드들 또는 광 입자들, 또는 이들의 임의의 결합으로 표현될 수 있다.

[0056]

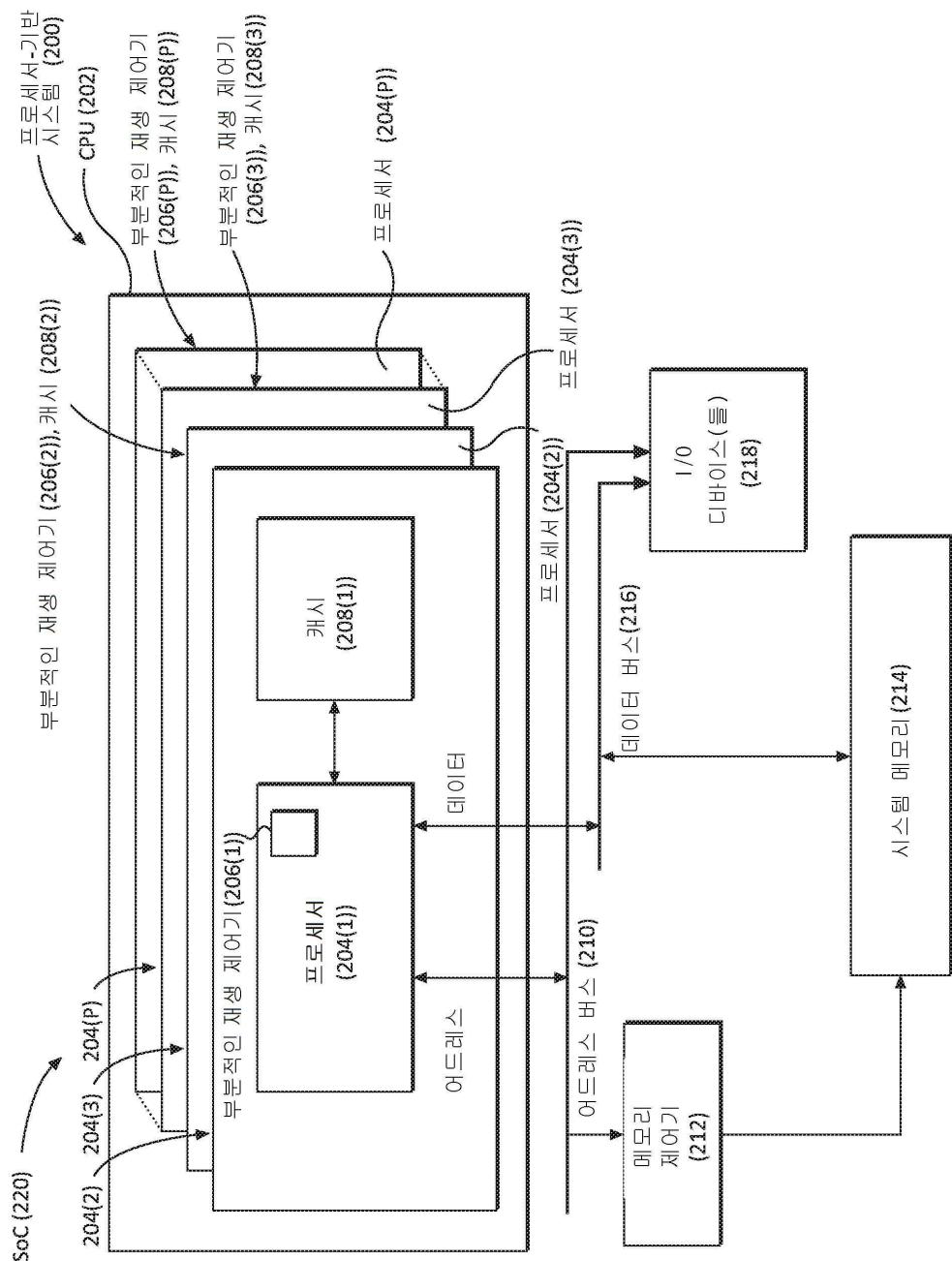
[0062] 본 개시내용의 상기의 설명은 임의의 당업자가 본 개시내용을 이용하거나 실시할 수 있게 하도록 제공된다. 본 개시내용에 대한 다양한 변형들이 당업자들에게 쉽게 명백할 것이며, 본원에 정의된 일반 원리들은 본 개시내용의 사상 또는 범위를 벗어나지 않으면서 다른 변형들에 적용될 수 있다. 그러므로 본 개시내용은 본원에서 설명된 예시들 및 설계들로 한정되는 것으로 의도되는 것이 아니라, 본원에 개시된 원리들 및 신규한 특징들에 부합하는 가장 넓은 범위에 따르는 것이다.

## 도면

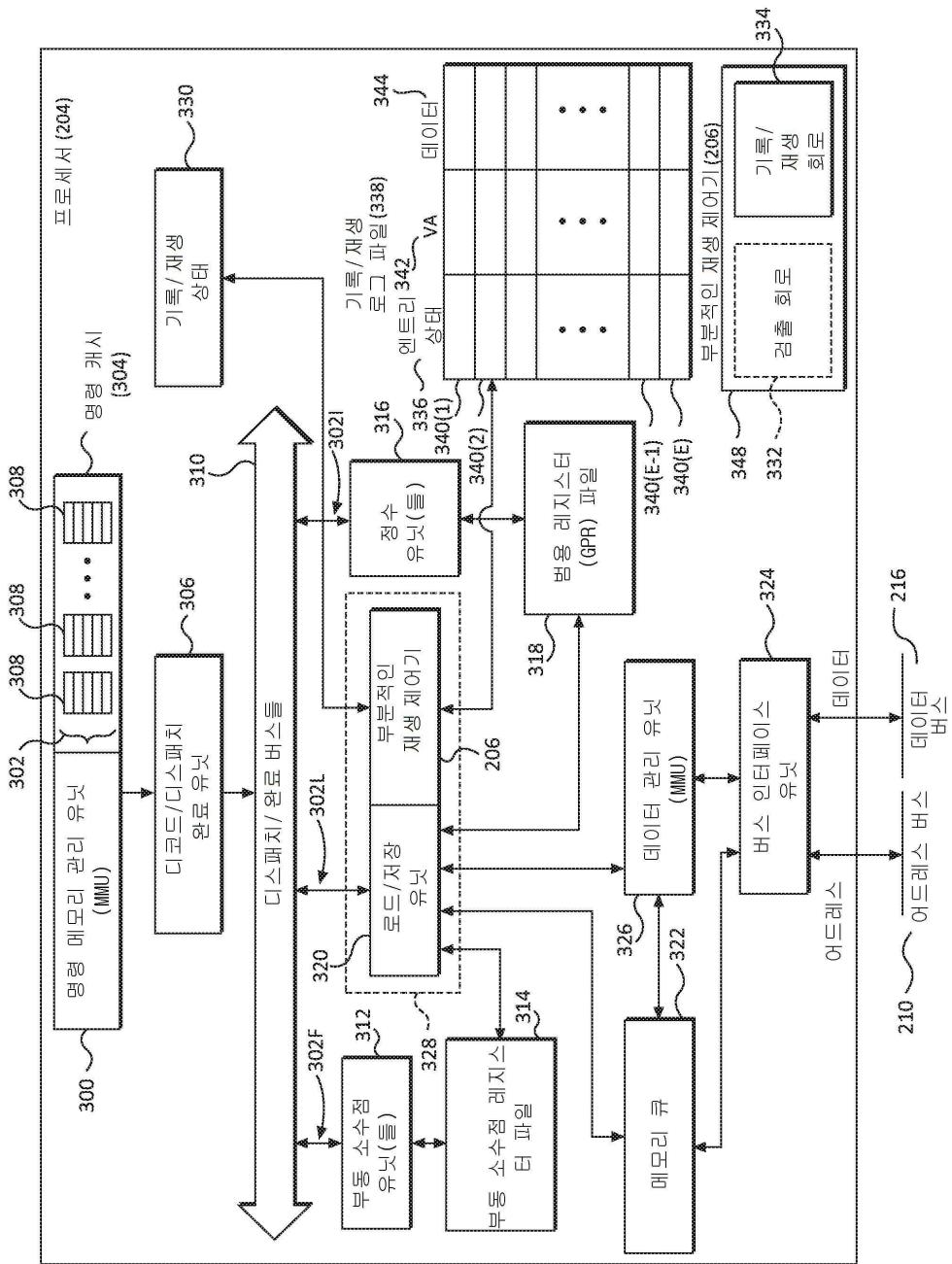
## 도면1



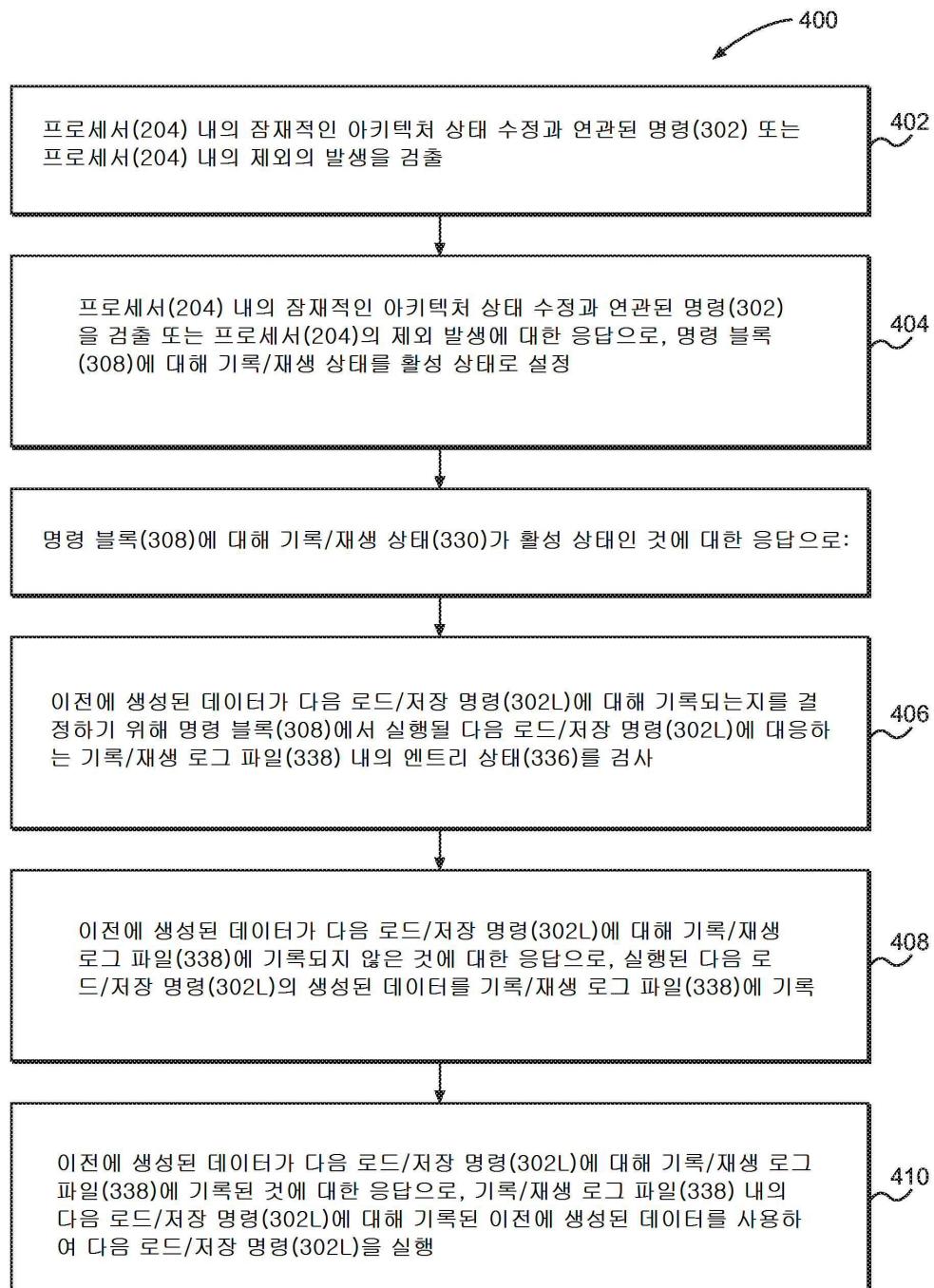
## 도면2



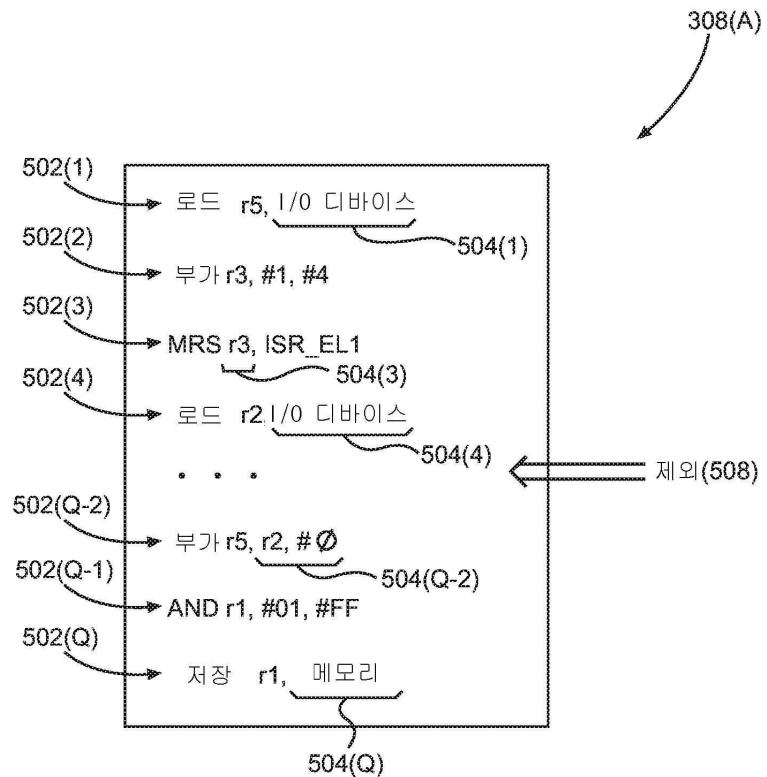
### 도면3



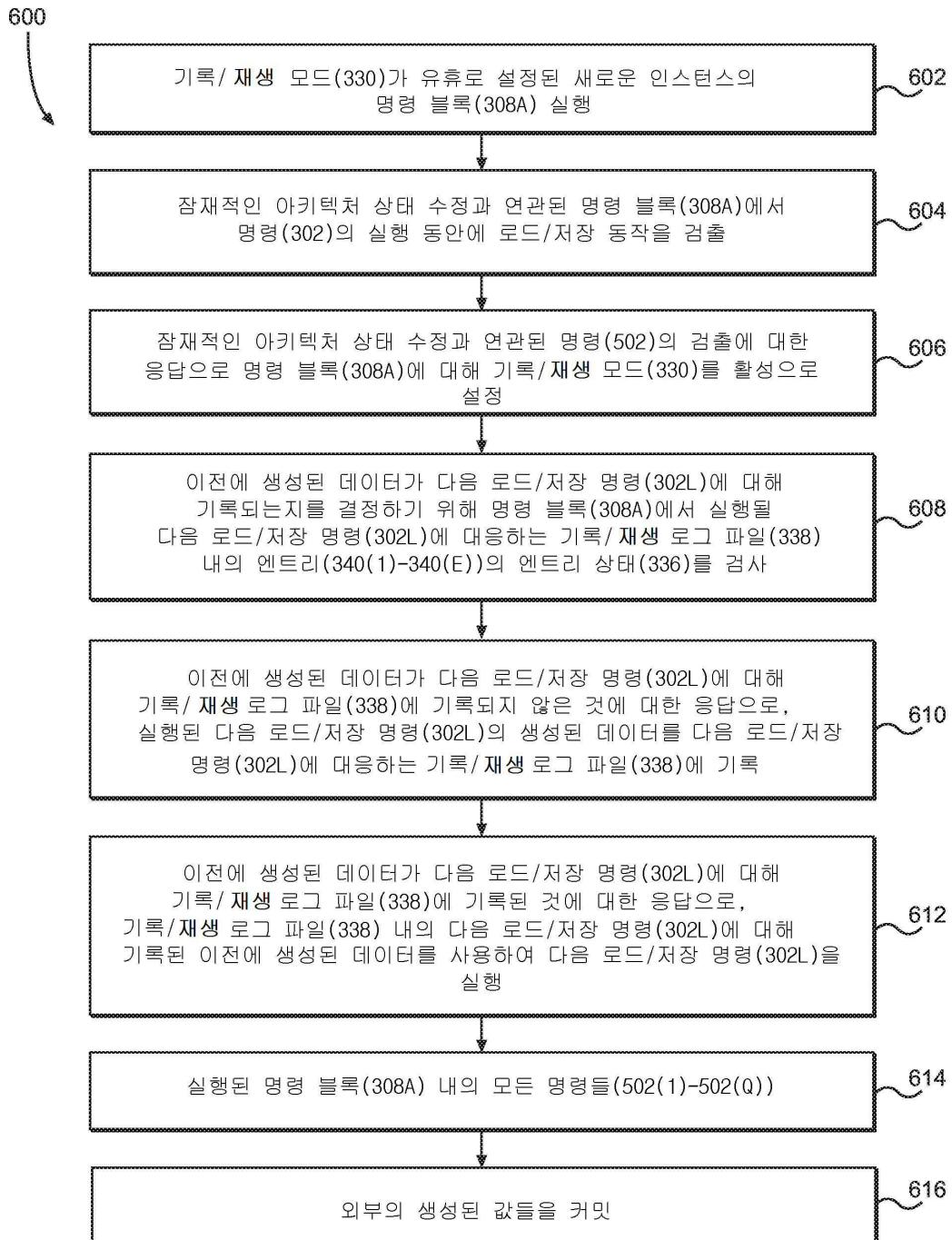
## 도면4



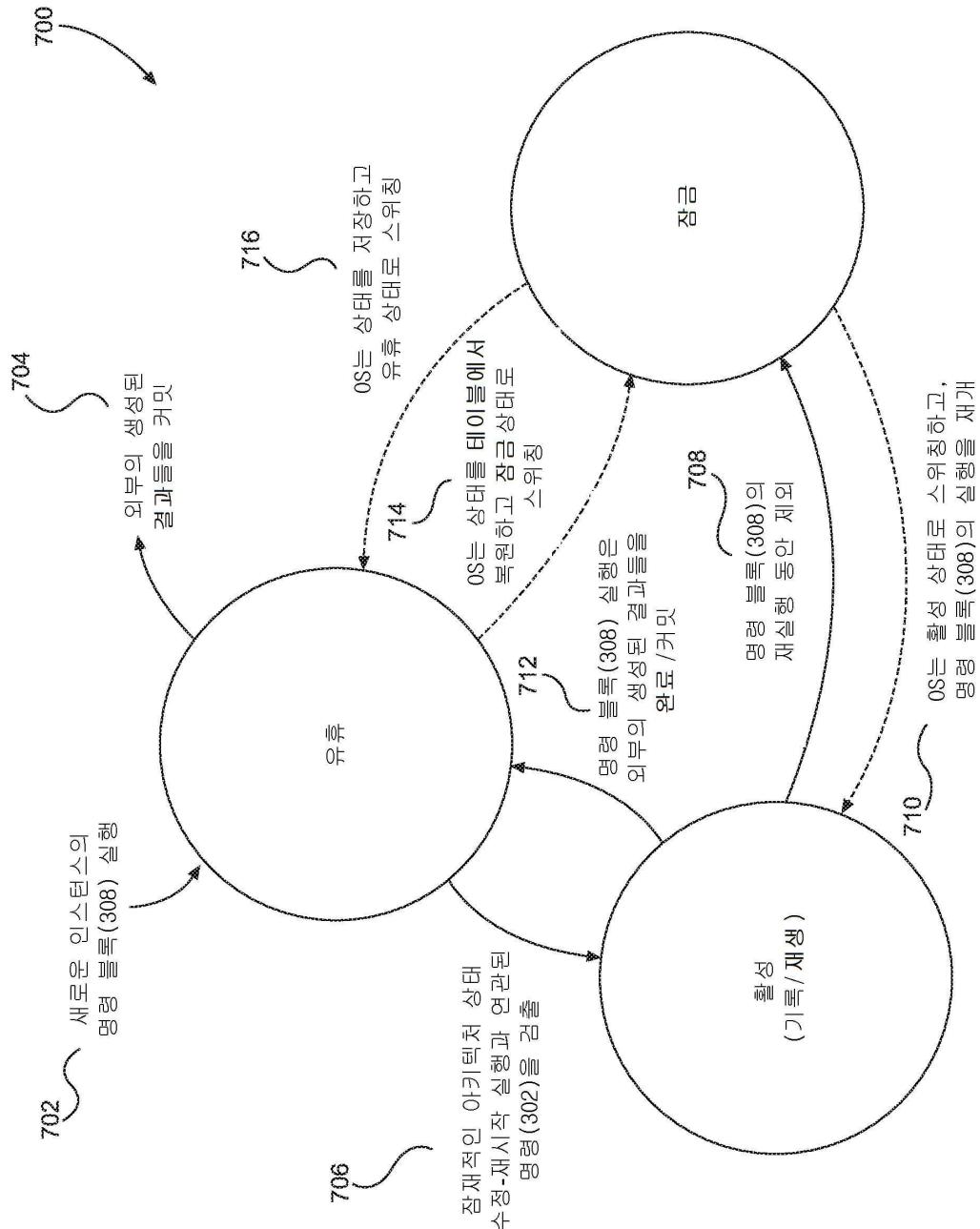
## 도면5



## 도면6



## 도면7



도면8

