



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2010-0041518
(43) 공개일자 2010년04월22일

(51) Int. Cl.

G06F 9/24 (2006.01) G06F 9/00 (2006.01)

(21) 출원번호 10-2008-0100745

(22) 출원일자 2008년10월14일

심사청구일자 2008년10월14일

(71) 출원인

포항공과대학교 산학협력단

경상북도 포항시 남구 효자동 산31 포항공과대학교내

(72) 발명자

박찬익

경북 포항시 남구 지곡동 교수아파트 9-1503

송재환

경상북도 포항시 남구 지곡동 756번지 포항공과대학원아파트 2-1405

박세진

대구광역시 북구 태전동 협화아파트 103-1506

(74) 대리인

리엔목특허법인

전체 청구항 수 : 총 5 항

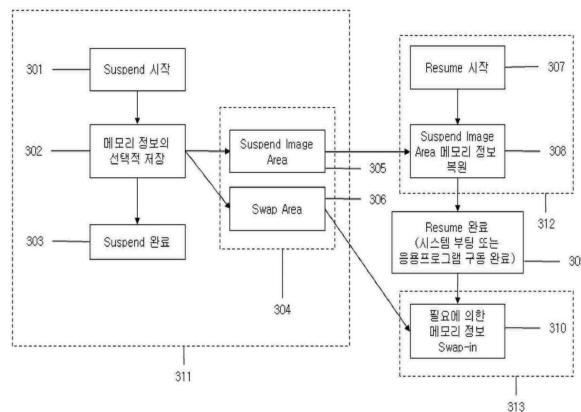
(54) 시스템에서 초기 구동시간을 단축시키는 선택적 서스펜드 리즘 방법 및 그 기록매체

(57) 요약

본 발명에 의한 시스템에서 초기 구동시간을 단축시키는 선택적 서스펜드 리즘 방법은, 시스템에서 사용 가능성이 높은 메모리 정보와 낮은 메모리 정보를 보조 메모리 장치에 설정한 서스펜드 이미지 영역과 스왑 영역에 각각 별도로 저장하여 서스펜드를 처리하는 과정; 및 상기 시스템을 부팅하는 동안에 상기 서스펜드 이미지 영역에 저장된 상기 사용 가능성이 높은 메모리 정보만을 메인 메모리에 복원하고, 리즘하여 상기 시스템을 기동하는 과정을 포함함을 특징으로 한다.

본 발명에 의하면 사용빈도가 높은 메모리 정보만을 우선적으로 메인 메모리에 복사함으로써, 시스템의 부팅 시간을 향상시킬 수 있으며, 사용빈도가 낮은 메모리 정보는 부팅 이후에 필요에 따라 메인 메모리에 로드가 이루어지므로, 시스템 자원을 효율적으로 사용할 수 있다.

대표도 - 도3



특허청구의 범위

청구항 1

시스템에서 사용 가능성이 높은 메모리 정보와 낮은 메모리 정보를 보조 메모리 장치에 설정한 서스펜드 이미지 영역과 스왑 영역에 각각 별도로 저장하여 서스펜드를 처리하는 과정; 및

상기 시스템을 부팅하는 동안에 상기 서스펜드 이미지 영역에 저장된 상기 사용 가능성이 높은 메모리 정보만을 메인 메모리에 로드하여 리즘하는 과정을 포함함을 특징으로 하는 시스템에서 초기 구동시간을 단축시키는 선택적 서스펜드 리즘 방법.

청구항 2

제 1항에 있어서, 부팅이 완료된 후 상기 시스템의 필요에 따라 상기 스왑 영역에 저장되어 있는 상기 메모리 정보를 상기 메인 메모리에 복사하는 과정을 더 포함함을 특징으로 하는 초기 구동시간을 단축시키는 선택적 서스펜드 리즘 방법.

청구항 3

제 1항에 있어서, 상기 메모리 정보의 사용 가능성의 높고 낮음은, 상기 메모리 정보에 대한 사용빈도나 특정 기간 동안 사용시간의 양에 따라 판단함을 특징으로 하는 초기 구동시간을 단축시키는 선택적 서스펜드 리즘 방법.

청구항 4

제 1항에 있어서, 상기 메모리 정보의 사용 가능성의 높고 낮음은, 특정 기간 동안의 사용자 사용 패턴에 의한 이용률 분석에 기초하여 판단함을 특징으로 하는 초기 구동시간을 단축시키는 선택적 서스펜드 리즘 방법.

청구항 5

제2항에 기재된 선택적 서스펜드 리즘 방법을 실행하는 프로그램을 저장한 컴퓨터 독취 가능한 저장매체.

명세서

발명의 상세한 설명

기술분야

[0001] 본 발명은 컴퓨터, 임베디드 장치 및 응용 프로그램의 초기 구동시간을 단축시키는 서스펜드 리즘(suspend resume) 방법 및 그 기록매체에 관한 것이다.

배경기술

[0002] 일반적으로 서스펜드 리즘은 전통적인 부팅방법과 달리 이전에 사용하던 메모리 정보를 보조 기억 장치(하드 디스크, 플래시 메모리 등)에 저장시킨 후에 부팅시에 저장되어 있는 메모리 정보를 원래 메모리 위치로 복원시킴으로 부팅 시간을 단축시키고 이전에 수행되던 작업 상태로 복원시키는 기술을 의미한다.

[0003] 이러한 방법은 현재 리눅스에서 Software Suspend Resume 과 Suspend 2로 구현되어 있으며, 윈도우에서는 최대 절전 모드라는 방식으로 구현이 되어 있고, Hibernation(수면) 방법으로도 불려진다.

[0004] 도 1은 종래의 서스펜드 과정을 설명하기 위한 흐름도이다.

[0005] 시스템 및 응용프로그램의 서스펜드가 시작되면 서스펜드를 위한 초기화 작업을 수행한다(101, 102단계). 이 초기화 작업에는 현재 수행중인 프로세스들을 중지시키는 작업과 각 디바이스 상태를 전이시키는 작업등이 포함된다.

[0006] 이 후 서스펜드 이미지를 만들기 위한, 정보 수집이 수행되며, 우선, 컴퓨터 정보(CPU 패밀리, 메모리 사이즈, 주변 디바이스 정보)들을 포함하는 메타 정보를 수집하고(103 단계), 현재 수행중인 메모리 정보를 수집한다(104 단계).

- [0007] 일반적인 메모리 정보는 페이지라는 단위로 수집되며, 수집된 메모리 정보를 하드 디스크나, 플래시 메모리와 같은 보조 기억장치에 저장하고(105 단계), 시스템 종료를 위해 각 디바이스의 동작을 중지시킨다(106 단계). 모든 작업들이 완료가 되면 서스펜드 작업이 종료된다(107 단계).
- [0008] 도 2는 종래의 리즘 과정을 설명하기 위한 흐름도이다.
- [0009] 우선 부팅을 위해 시스템이 기동되면(201 단계) 시스템은 부트로더, 운영체제를 로드한 후 리즘 작업을 시작한다(202, 203 단계). 이 리즘작업은 부트로더에서 수행될 수도 있고, 운영체제에서 수행될 수도 있다. 우선, 이 리즘을 하기 위해 현재 수행중인 프로세스를 멈추는 등의 초기화 작업이 수행되고(204 단계), 보조 기억장치에 저장되어 있던 메타 정보를 로드한다(205 단계). 그리고 보조 기억장치에 저장되어 있던 메모리 정보를 로드하여 이전 수행 상태로 복원한다(206, 207 단계).
- [0010] 리즘이 완료되면(208 단계) 이전의 수행 상태로 사용자가 컴퓨터 또는 임베디드 장치를 즉시 사용할 수 있게 된다.
- [0011] 한편, 응용프로그램의 리즘 과정이 적용되는 경우에는 도 2의 201 단계와 202 단계에서 응용프로그램을 로드하는 과정이 수행된다. 응용프로그램이 로드되면서 203 단계의 리즘이 수행되어 응용프로그램의 구동이 이루어진다.
- [0012] 이러한 종래의 리즘 과정에 있어서, 가장 처리 시간이 많이 소요되는 부분은 보조 기억장치에 저장되어 있는 메모리 정보를 모두 리드하여, 원래 존재하던 위치로 복사시키는 과정으로, 리즘 작업의 대부분을 차지하고 있다.
- [0013] 이에 따라 리즘하는 동안에 메모리 정보의 복사량을 작게 하여 종래의 리즘 처리를 보다 신속히 처리할 수 있는 새로운 방안이 요구되어 왔다.

발명의 내용

해결 하고자하는 과제

- [0014] 본 발명의 과제는 상술된 종래 기술의 문제점을 감안하여, 컴퓨터 및 임베디드 시스템에서의 서스펜스 리즘 처리 과정에 있어서, 보조 기억장치에서 메인 메모리로 복사시킬 메모리 정보를 우선순위에 따라 부팅종과 이후로 선택적으로 복사함으로써, 서스펜드 리즘을 보다 신속하게 처리하는 방법을 제공하는 데 있다.

과제 해결수단

- [0015] 상기 과제를 달성하기 위한 본 발명에 의한 초기 구동시간을 단축시키는 선택적 서스펜드 리즘 방법은,
- [0016] 시스템에서 사용 가능성이 높은 메모리 정보와 낮은 메모리 정보를 보조 메모리 장치에 설정한 서스펜드 이미지 영역과 스왑 영역에 각각 별도로 저장하여 서스펜드를 처리하는 과정; 및
- [0017] 상기 시스템을 부팅하는 동안에 상기 서스펜드 이미지 영역에 저장된 상기 메모리 정보만을 메인 메모리에 리드하여 리즘하는 과정을 포함함을 특징으로 한다.
- [0018] 또한, 상기 컴퓨터의 부팅이 완료된 후 상기 시스템의 필요에 따라 상기 스왑 영역에 저장되어 있는 상기 메모리 정보를 상기 메인 메모리에 복사하는 과정을 더 포함함을 특징으로 한다.
- [0019] 또한, 상기 메모리 정보의 사용 가능성의 높고 낮음은, 상기 메모리 정보에 대한 사용빈도나 특정 기간 동안 사용시간의 양에 따라 판단함을 특징으로 한다.
- [0020] 또한, 상기 메모리 정보의 사용 가능성의 높고 낮음은, 특정 기간 동안의 사용자 사용 패턴에 의한 이용률 분석에 기초하여 판단함을 특징으로 한다.

효 과

- [0021] 본 발명은 컴퓨터 및 임베디드 장치의 서스펜드 리즘을 이용한 부팅 시, 사용빈도가 높은 메모리 정보만을 우선적으로 메인 메모리에 복사함으로써, 시스템의 부팅 시간을 향상시킬 수 있으며, 사용빈도가 낮은 메모리 정보는 부팅 이후에 필요에 따라 메인 메모리에 로드가 이루어지므로, 시스템 자원을 효율적으로 사용할 수 있다.
- [0022] 또한, 소프트웨어적으로 부팅 시간을 단축하는 방법이기 때문에 별도의 하드웨어의 변경 또는 추가 없이 적용시킬 수 있다.

[0023] 또한, 서스펜드 리즘을 지원하는 모든 종류의 운영체제 및 부트로더, 응용프로그램 로더 등에 다양하게 적용할 수 있는 장점이 있다.

[0024] 또한, 서스펜드 리즘이 적용된 응용프로그램의 구동시간을 줄여 사용자의 만족도를 향상시킬 수 있다.

발명의 실시를 위한 구체적인 내용

[0025] 이하, 첨부 도면을 참조하여 본 발명의 바람직한 실시예를 설명하기로 한다.

[0026] 도 3은 본 발명에 의한 선택적 서스펜드 리즘의 소프트웨어 구성을 설명하기 위한 도면이다.

[0027] 선택적 서스펜드 처리(311)

[0028] 시스템의 정보 또는 응용프로그램의 정보를 저장하고 리즘을 하기 위한 이미지를 생성하기 위해 서스펜드를 시작한다(301 단계). 서스펜드를 시작하게 되면 메모리 정보를 선택적으로 서스펜드 이미지 영역과 스왑 영역에 저장한다.

[0029] 즉, 보조 기억장치(304)에 스왑 영역을 설정하고, 메모리 정보를 선택적으로 저장시키는 302 단계가 종래의 서스펜드 과정에 추가된다.

[0030] 또한, 본 발명을 리눅스의 신속한 부팅을 적용한 예로서 다음과 같이 종래의 서스펜드 리즘의 과정에 일부의 수정만을 통해 제시하고 있는 시스템이 구축될 수 있다.

[0031] 현재 리눅스 시스템에는 스왑핑(Swapping) 메커니즘이 적용되어 있기 때문에 스왑핑 메커니즘에서 사용하고 있는 스왑영역을 두 영역으로 분할한다(305 및 306 단계).

[0032] 즉, 종래에는 리눅스의 스왑 영역에 서스펜드 이미지를 복사시켜 스왑 영역이 서스펜드 이미지 영역의 역할을 수행하는 구조였으나, 저장 영역을 분할시킴으로써 본 발명에서 제시하는 프레임 워크를 구축할 수 있다.

[0033] 리눅스에서는 메모리 정보를 페이지라는 단위로 나누어 처리하고 있으며, 메모리 정보의 선택적 저장, 즉 페이지의 선택적 저장은 다양한 요소들을 고려하여 처리할 수 있다. 주요 고려 요소는 다음과 같으며, 추가적인 고려 요소들로 확장시킬 수 있다.

[0034] 1. 페이지의 사용 빈도 - 사용빈도가 높은 페이지는 서스펜드 이미지 영역에 저장하고, 사용 빈도가 상대적으로 낮은 페이지들은 스왑 영역에 저장되도록 스왑 아웃시킨다.

[0035] 2. 페이지의 최근 사용 시간 - 최근에 사용한 시간이 가장 멀리 있는 페이지부터 스왑 영역에 스왑 아웃시켜서 사용한다.

[0036] 3. 특정 기간 동안의 사용자의 사용 패턴 분석에 따른 페이지의 이용률 - 특정기간 사용자의 수행 패턴을 분석하여 자주 쓰이는 페이지와 자주 쓰이지 않는 페이지를 구분하여 스왑 영역에 스왑 아웃시켜서 사용한다.

[0037] 4. 임의의 방법 - 랜덤하게 임의의 페이지들을 스왑 영역에 저장하여 스왑 아웃시켜서 사용한다.

[0038] 선택적 리즘 처리(312)

[0039] 또한, 리즘이 시작되면(307 단계), 종래의 리눅스에서의 처리 방식과 같이 서스펜드 이미지 영역에 있는 모든 메모리 정보를 메인 메모리에 복사하는 처리 방식(308 단계)을 수행하여 시스템을 부팅하거나 응용프로그램 구동을 완료한다(309 단계). 서스펜드 이미지 영역에 저장되어 있는 사용빈도가 높은 메모리 정보만이 메인 메모리에 복사된다.

[0040] 따라서, 현재 구성되어 있는 리눅스의 리즘 단계는 별도의 수정 없이 그대로 적용하여 사용할 수 있다. 리즘하는 동안 스왑 영역에 있는 페이지들은 복원시키지 않으며, 서스펜드 이미지 영역에 저장되어 있는 페이지들만 복원시켜 부팅을 완료한다. 이때, 스왑 영역에 저장된 사용빈도가 낮은 페이지는 복사되지 않으므로 부팅 시간을 그만큼 단축시킬 수 있다.

[0041] 부팅 이후 필요에 의한 메모리정보 스왑 인(Swap-in) 처리(313)

[0042] 리즘 작업이 완료가 되면(309 단계), 즉 시스템 부팅이나 응용프로그램의 구동이 완료가 되면, 사용자의 요청에 의해 필요한 메모리 정보만 스왑 영역에서 추가적으로 로드가 메인 메모리측에 이루어진다(310 단계). 이 스왑 인 작업은 필요에 따라 미도시된 스왑 인 처리기에 의해 처리된다.

- [0043] 예컨대, 리눅스에서 서스펜드 리즘을 구현할 경우 다른 모듈의 추가나 변경 없이 리눅스에서 요청한 페이지가 메모리에 로드되어 있지 않을 경우인 Page-fault를 처리하기 위해 적용하는 Page-fault Handler를 바로 사용할 수 있다. 즉, 서스펜드시에 선택된 페이지를 스왑 아웃시킬 때 사용한 스왑 영역을 리눅스가 스왑핑 메커니즘에서 사용하는 스왑 영역으로 설정해서 주어진 프레임 워크를 활용할 수 있다.
- [0044] 도 4는 본 발명에 의한 서스펜드 과정을 설명하기 위한 흐름도이며, 상술된 종래 기술과 대응된 부분에 대한 상세한 설명을 생략하기로 한다.
- [0045] 도 4는 도 1에 도시된 종래의 서스펜드 과정과 상이한 부분으로서 선택적으로 사용빈도가 낮은 메모리 정보를 스왑 영역에 별도로 저장하기 위한 스왑 아웃(Swap-out)하는 403 단계가 추가된다. 도면상에서는 스왑 아웃하는 403 단계가 서스펜드 작업의 초기화 과정(402 단계)을 수행 후, 시스템의 메타 정보의 수집(404 단계) 이전에 수행되지만, 이 선택적 메모리 스왑 아웃(Swap-out)(403 단계)은 서스펜드 작업이 이루어지기 전에 미리 처리될 수 있으며, 메모리 정보가 보조 기억장치에 기록이 될 때 동시에 두 영역에 기록이 될 수도 있다. 즉, 이 처리 과정은 구현에 따라 처리 순서가 다르게 될 수도 있다.
- [0046] 도 5는 본 발명에 의한 리즘 과정을 설명하기 위한 흐름도이다.
- [0047] 리즘의 전 과정은 크게 스펜드 이미지 영역의 메모리 정보 복사를 위한 동작(510 단계)과 부팅 또는 응용프로그램의 구동 후의 동작(511 단계)으로 나누어진다. 리즘 이전의 동작에서 보조기억장치에서 메모리 정보를 로드할 때(506 단계) 서스펜드 이미지 영역(305)에 저장되어 있는 일부분의 메모리 정보만을 로드하여 복원 시간을 단축시킬 수 있다.
- [0048] 또한, 본 발명은 상기 실시예의 서스펜드 리즘 방법을 지원하는 운영체제와 이 운영체제를 로드시키는 부트로더로서 구현될 수 있다.
- [0049] 또한, 운영체제 도움없이 상기 실시예의 서스펜드 리즘 방법을 자체적으로 지원하는 부트로더로서 구현될 수 있으며, 이를 지원하는 응용프로그램으로 구현될 수도 있다.
- [0050] 이처럼, 본 발명의 상세한 설명에는 구체적인 적용 예에 관해 설명하였으나, 본 발명의 범주를 벗어나지 않는 한도 내에서의 다양한 변형 또한 가능하다. 그러므로 본 발명의 범위는 설명된 실시예나 흐름도 등에 국한되어 정해져서는 안 되며, 후술되는 특허 청구범위뿐만 아니라, 이 특허 청구 범위와 균등한 것들에 의해 정해져야 할 것이다.

산업이용 가능성

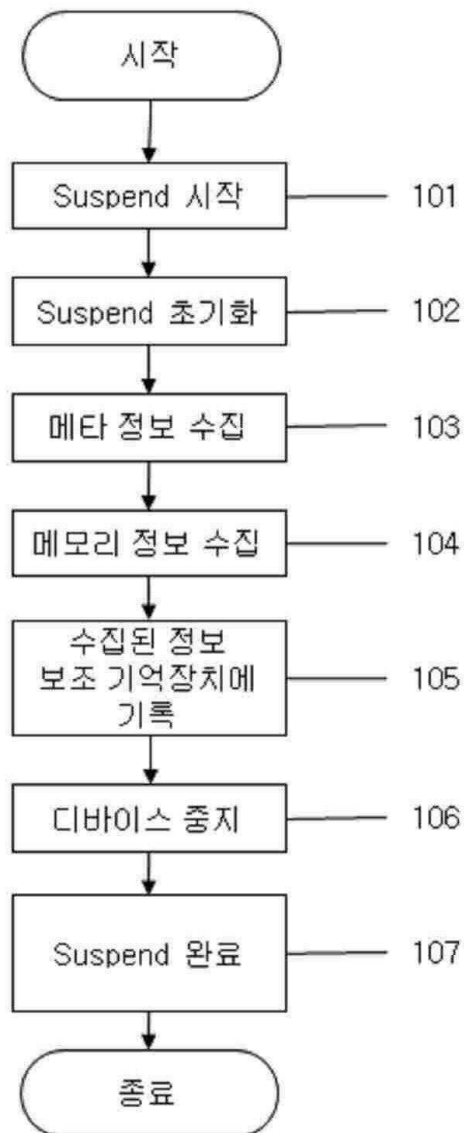
- [0051] 본 발명이 적용된 운영체제 및 부트로더는 기존 구동 시간보다 구동 시간을 단축시킬 수 있으며, 응용 프로그램의 구동 시간 역시 단축시킬 수 있어, 즉각적인 시스템 및 응용프로그램의 사용이 가능해 진다. 이는 컴퓨터 뿐만 아니라, 로딩 시간이 많이 걸리는 디지털 TV와 같은 가전 기기에서부터 MP3 플레이어와 같은 소형 전자 기기까지도 광범위하게 적용될 수 있다.

도면의 간단한 설명

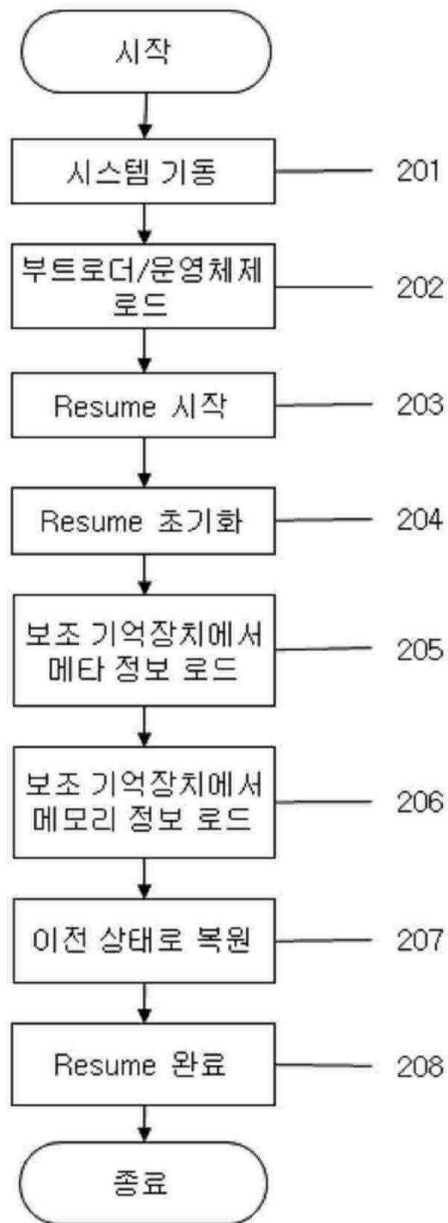
- [0052] 도 1은 종래의 서스펜드 과정을 설명하기 위한 흐름도이다.
- [0053] 도 2는 종래의 리즘 과정을 설명하기 위한 흐름도이다.
- [0054] 도 3은 본 발명에 의한 선택적 서스펜드 리즘의 소프트웨어 구성을 설명하기 위한 도면이다.
- [0055] 도 4는 본 발명에 의한 서스펜드 과정을 설명하기 위한 흐름도이다.
- [0056] 도 5는 본 발명에 의한 리즘 과정을 설명하기 위한 흐름도이다.

도면

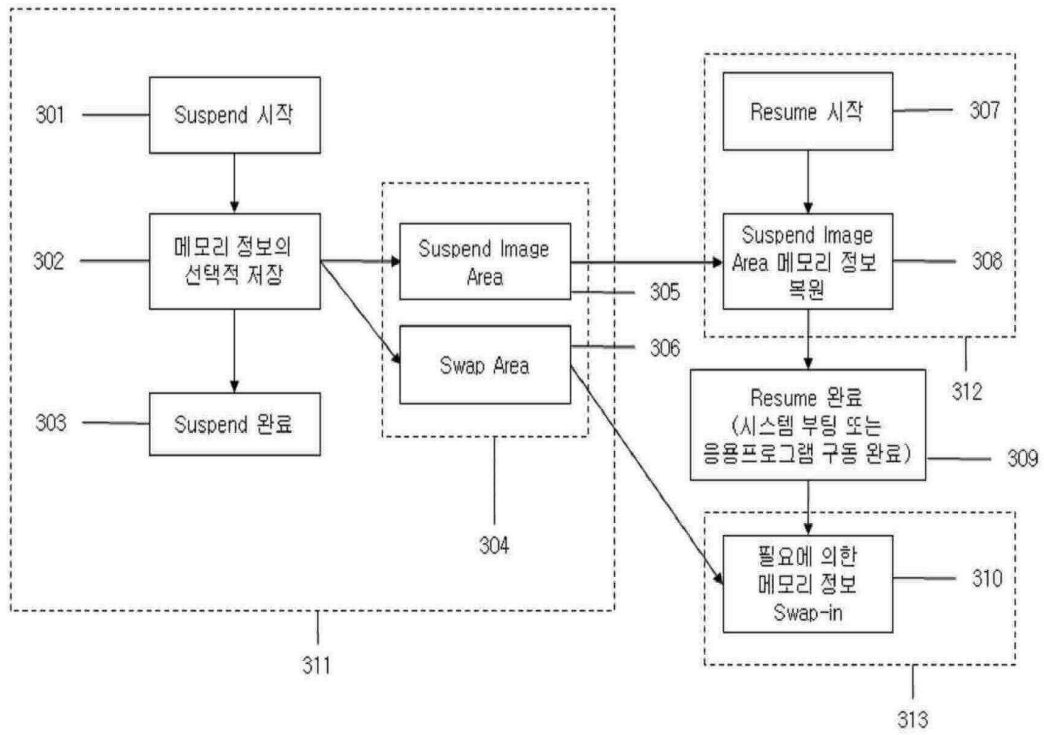
도면1



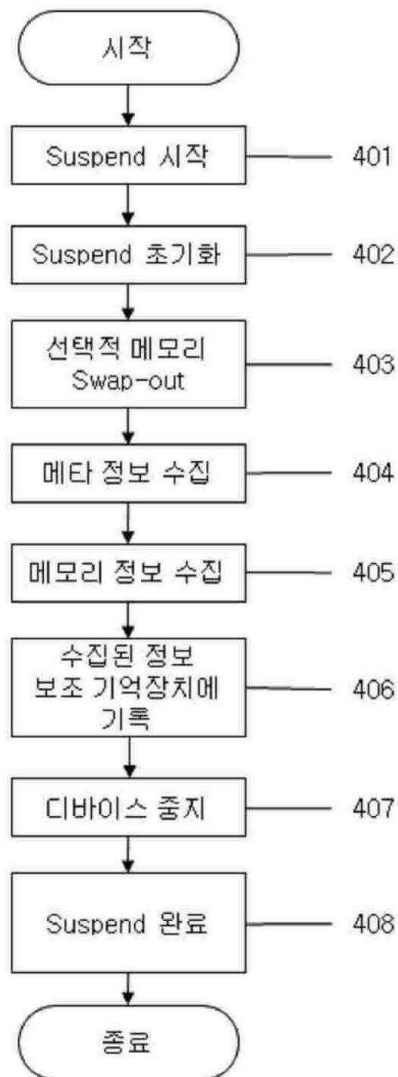
도면2



도면3



도면4



도면5

