



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2024-0170011
(43) 공개일자 2024년12월03일

(51) 국제특허분류(Int. Cl.)
G06F 8/654 (2018.01) G06F 12/1009 (2016.01)
G06F 9/38 (2024.01) G06F 9/48 (2018.01)
(52) CPC특허분류
G06F 8/654 (2018.02)
G06F 12/1009 (2013.01)
(21) 출원번호 10-2023-0067926
(22) 출원일자 2023년05월25일
심사청구일자 2023년05월25일

(71) 출원인
경희대학교 산학협력단
경기도 용인시 기흥구 덕영대로 1732 (서천동, 경희대학교 국제캠퍼스내)
(72) 발명자
조진성
경기도 수원시 영통구 봉영로 1620, 18층 101-1801호 (영통동, 대우 월드마크 영통)
손희승
경기도 용인시 수지구 대지로 27, 102동 308호 (죽전동, 대주한신아파트)
(74) 대리인
두호특허법인

전체 청구항 수 : 총 19 항

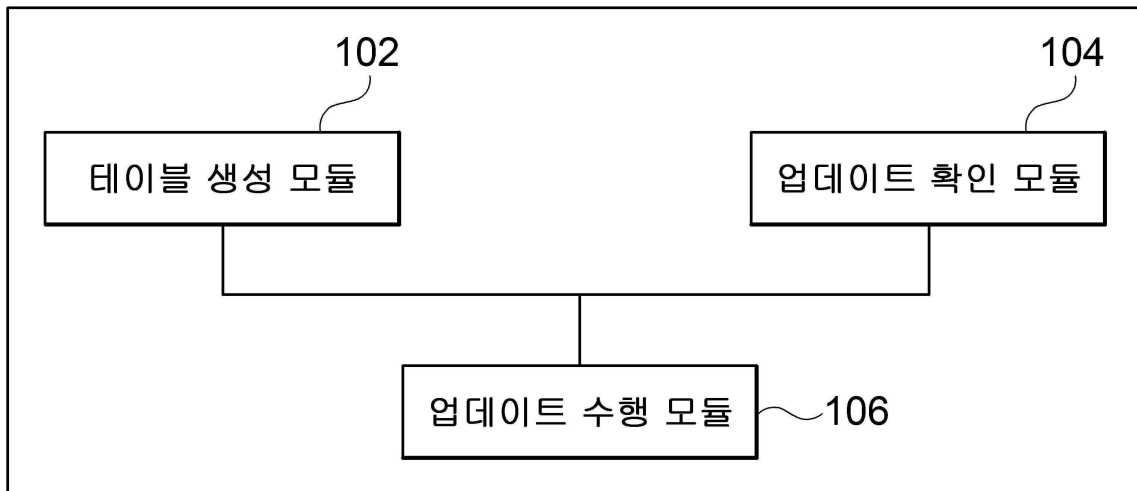
(54) 발명의 명칭 소프트웨어 업데이트 장치 및 방법

(57) 요약

소프트웨어 업데이트 장치 및 방법이 개시된다. 개시되는 일 실시예에 따른 소프트웨어 업데이트 장치는, 업데이트의 대상이 되는 소프트웨어에 대한 레퍼런스 테이블을 생성하는 테이블 생성 모듈, 소프트웨어의 업데이트 이벤트가 발생하는지를 확인하고, 업데이트 이벤트가 발생한 경우 업데이트의 구성 요소 종류 및 업데이트의 내용 종류 중 하나 이상을 확인하는 업데이트 확인 모듈, 및 레퍼런스 테이블에 기초하여 확인된 업데이트의 구성 요소 종류 및 내용 종류 중 하나 이상에 따라 소프트웨어의 업데이트를 수행하는 업데이트 수행 모듈을 포함한다.

대표도 - 도1

100



(52) CPC특허분류

G06F 9/3851 (2023.08)

G06F 9/4881 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711183722
과제번호	2022R1F1A1063724
부처명	과학기술정보통신부
과제관리(전문)기관명	한국연구재단
연구사업명	개인기초연구(과기정통부)
연구과제명	RTOS 기반 IoT 시스템 보안 기술 연구
기 여 율	1/2
과제수행기관명	경희대학교(국제캠퍼스)
연구기간	2023.03.01 ~ 2024.02.29

이 발명을 지원한 국가연구개발사업

과제고유번호	1711198879
과제번호	00266615
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	정보통신방송혁신인재양성
연구과제명	융합보안대학원 (경희대학교)
기 여 율	1/2
과제수행기관명	경희대학교산학협력단
연구기간	2023.07.01 ~ 2023.12.31

명세서

청구범위

청구항 1

하나 이상의 프로세서들, 및

상기 하나 이상의 프로세서들에 의해 실행되는 하나 이상의 프로그램들을 저장하는 메모리를 구비한 소프트웨어 업데이트 장치로서,

업데이트의 대상이 되는 소프트웨어에 대한 레퍼런스 테이블을 생성하는 테이블 생성 모듈;

상기 소프트웨어의 업데이트 이벤트가 발생하는지를 확인하고, 상기 업데이트 이벤트가 발생한 경우 업데이트의 구성 요소 종류 및 업데이트의 내용 종류 중 하나 이상을 확인하는 업데이트 확인 모듈; 및

상기 레퍼런스 테이블에 기초하여 상기 확인된 업데이트의 구성 요소 종류 및 내용 종류 중 하나 이상에 따라 상기 소프트웨어의 업데이트를 수행하는 업데이트 수행 모듈을 포함하는, 소프트웨어 업데이트 장치.

청구항 2

청구항 1에 있어서,

상기 테이블 생성 모듈은,

상기 소프트웨어를 구성하는 각 함수 및 전역변수들을 추출하고, 상기 추출한 함수 및 전역변수들의 식별 정보를 상기 함수 및 전역변수들이 메모리에 저장된 주소와 매핑하여 레퍼런스 테이블을 생성하는, 소프트웨어 업데이트 장치.

청구항 3

청구항 2에 있어서,

상기 업데이트 확인 모듈은,

상기 업데이트 이벤트가 발생한 경우 상기 업데이트의 구성 요소 종류가 함수 또는 전역변수인지를 확인하고, 상기 업데이트의 내용 종류가 내적 변경 또는 외적 변경인지를 확인하는, 소프트웨어 업데이트 장치.

청구항 4

청구항 3에 있어서,

상기 함수의 내적 변경은, 함수 내부의 인스트럭션만 변경되는 경우이고,

상기 함수의 외적 변경은, 새로운 함수가 추가되거나 함수의 인터페이스가 변경되는 경우인, 소프트웨어 업데이트 장치.

청구항 5

청구항 4에 있어서,

상기 업데이트 수행 모듈은,

상기 업데이트 이벤트가 상기 함수의 내적 변경인 경우, 업데이트된 함수를 메모리에 적재하고, 상기 레퍼런스

테이블에서 해당 함수의 주소를 업데이트된 함수의 주소로 변경하고,

상기 업데이트 이벤트가 상기 함수의 외적 변경인 경우, 업데이트된 함수를 메모리에 적재하고, 상기 레퍼런스 테이블에서 업데이트된 함수에 대한 항목을 추가하는, 소프트웨어 업데이트 장치.

청구항 6

청구항 3에 있어서,

상기 전역변수의 내적 변경은, 전역변수의 값(value)만 변경되는 경우이고,

상기 전역변수의 외적 변경은, 새로운 전역변수가 추가되거나 전역변수의 타입이 변경되는 경우인, 소프트웨어 업데이트 장치.

청구항 7

청구항 6에 있어서,

상기 업데이트 수행 모듈은,

상기 업데이트 이벤트가 상기 전역변수의 내적 변경인 경우, 업데이트된 전역변수를 메모리에 적재하고, 상기 레퍼런스 테이블에서 해당 전역변수의 주소를 업데이트된 전역변수의 주소로 변경하고,

상기 업데이트 이벤트가 상기 전역변수의 외적 변경인 경우, 업데이트된 전역변수를 메모리에 적재하고, 상기 레퍼런스 테이블에서 업데이트된 전역변수에 대한 항목을 추가하는, 소프트웨어 업데이트 장치.

청구항 8

청구항 3에 있어서,

상기 업데이트 수행 모듈은,

상기 확인된 업데이트의 구성 요소 종류 및 내용 종류에 따라 업데이트 된 함수 또는 전역변수를 메모리에 적재하는 쓰레드 및 상기 레퍼런스 테이블을 변경하는 쓰레드를 수행하되, 상기 업데이트에 따라 발생하는 쓰레드와 사용자 명령 쓰레드 간에 우선순위를 설정하는, 소프트웨어 업데이트 장치.

청구항 9

청구항 8에 있어서,

상기 업데이트 수행 모듈은,

상기 함수 또는 전역변수를 메모리에 적재하는 쓰레드는 상기 사용자 명령 쓰레드 보다 우선순위를 낮게 설정하고, 상기 레퍼런스 테이블을 변경하는 쓰레드는 상기 사용자 명령 쓰레드 보다 우선순위를 높게 설정하는, 소프트웨어 업데이트 장치.

청구항 10

하나 이상의 프로세서들, 및

상기 하나 이상의 프로세서들에 의해 실행되는 하나 이상의 프로그램들을 저장하는 메모리를 구비한 컴퓨팅 장치에서 수행되는 방법으로서,

업데이트의 대상이 되는 소프트웨어에 대한 레퍼런스 테이블을 생성하는 단계;

상기 소프트웨어의 업데이트 이벤트가 발생하는지를 확인하고, 상기 업데이트 이벤트가 발생한 경우 업데이트의

구성 요소 종류 및 업데이트의 내용 종류 중 하나 이상을 확인하는 단계; 및

상기 레퍼런스 테이블에 기초하여 상기 확인된 업데이트의 구성 요소 종류 및 내용 종류 중 하나 이상에 따라 상기 소프트웨어의 업데이트를 수행하는 단계를 포함하는, 소프트웨어 업데이트 방법.

청구항 11

청구항 10에 있어서,

상기 레퍼런스 테이블을 생성하는 단계는,

상기 소프트웨어를 구성하는 각 함수 및 전역변수들을 추출하는 단계; 및

상기 추출한 함수 및 전역변수들의 식별 정보를 상기 함수 및 전역변수들이 메모리에 저장된 주소와 매핑하여 레퍼런스 테이블을 생성하는 단계를 포함하는, 소프트웨어 업데이트 방법.

청구항 12

청구항 11에 있어서,

상기 확인하는 단계는,

상기 업데이트 이벤트가 발생한 경우 상기 업데이트의 구성 요소 종류가 함수 또는 전역변수인지를 확인하는 단계; 및

상기 업데이트의 내용 종류가 내적 변경 또는 외적 변경인지를 확인하는 단계를 포함하는, 소프트웨어 업데이트 방법.

청구항 13

청구항 12에 있어서,

상기 함수의 내적 변경은, 함수 내부의 인스트럭션만 변경되는 경우이고,

상기 함수의 외적 변경은, 새로운 함수가 추가되거나 함수의 인터페이스가 변경되는 경우인, 소프트웨어 업데이트 방법.

청구항 14

청구항 13에 있어서,

상기 업데이트를 수행하는 단계는,

상기 업데이트 이벤트가 상기 함수의 내적 변경인 경우, 업데이트된 함수를 메모리에 적재하고, 상기 레퍼런스 테이블에서 해당 함수의 주소를 업데이트된 함수의 주소로 변경하는 단계; 및

상기 업데이트 이벤트가 상기 함수의 외적 변경인 경우, 업데이트된 함수를 메모리에 적재하고, 상기 레퍼런스 테이블에서 업데이트된 함수에 대한 항목을 추가하는 단계를 포함하는, 소프트웨어 업데이트 방법.

청구항 15

청구항 12에 있어서,

상기 전역변수의 내적 변경은, 전역변수의 값(value)만 변경되는 경우이고,

상기 전역변수의 외적 변경은, 새로운 전역변수가 추가되거나 전역변수의 타입이 변경되는 경우인, 소프트웨어

업데이트 방법.

청구항 16

청구항 15에 있어서,

상기 업데이트를 수행하는 단계는,

상기 업데이트 이벤트가 상기 전역변수의 내적 변경인 경우, 업데이트된 전역변수를 메모리에 적재하고, 상기 레퍼런스 테이블에서 해당 전역변수의 주소를 업데이트된 전역변수의 주소로 변경하는 단계; 및

상기 업데이트 이벤트가 상기 전역변수의 외적 변경인 경우, 업데이트된 전역변수를 메모리에 적재하고, 상기 레퍼런스 테이블에서 업데이트된 전역변수에 대한 항목을 추가하는 단계를 포함하는, 소프트웨어 업데이트 방법.

청구항 17

청구항 12에 있어서,

상기 업데이트를 수행하는 단계는,

상기 확인된 업데이트의 구성 요소 종류 및 내용 종류에 따라 업데이트 된 함수 또는 전역변수를 메모리에 적재하는 쓰레드 및 상기 레퍼런스 테이블을 변경하는 쓰레드를 수행하되, 상기 업데이트에 따라 발생하는 쓰레드와 사용자 명령 쓰레드 간에 우선순위를 설정하는 단계를 더 포함하는, 소프트웨어 업데이트 방법.

청구항 18

청구항 17에 있어서,

상기 우선순위를 설정하는 단계는,

상기 함수 또는 전역변수를 메모리에 적재하는 쓰레드는 상기 사용자 명령 쓰레드 보다 우선순위를 낮게 설정하고, 상기 레퍼런스 테이블을 변경하는 쓰레드는 상기 사용자 명령 쓰레드 보다 우선순위를 높게 설정하는, 소프트웨어 업데이트 방법.

청구항 19

비일시적 컴퓨터 판독 가능한 저장 매체(non-transitory computer readable storage medium)에 저장된 컴퓨터 프로그램으로서,

상기 컴퓨터 프로그램은 하나 이상의 명령어들을 포함하고, 상기 명령어들은 하나 이상의 프로세서들을 갖는 컴퓨팅 장치에 의해 실행될 때, 상기 컴퓨팅 장치로 하여금,

업데이트의 대상이 되는 소프트웨어에 대한 레퍼런스 테이블을 생성하는 단계;

상기 소프트웨어의 업데이트 이벤트가 발생하는지를 확인하고, 상기 업데이트 이벤트가 발생한 경우 업데이트의 구성 요소 종류 및 업데이트의 내용 종류 중 하나 이상을 확인하는 단계; 및

상기 레퍼런스 테이블에 기초하여 상기 확인된 업데이트의 구성 요소 종류 및 내용 종류 중 하나 이상에 따라 상기 소프트웨어의 업데이트를 수행하는 단계를 수행하도록 하는, 컴퓨터 프로그램.

발명의 설명

기술 분야

[0001] 본 발명의 실시예는 소프트웨어 업데이트 기술과 관련된다.

배경 기술

[0003] 일반적으로, 소프트웨어는 함수, 변수 등의 여러 구성 요소들이 유기적으로 구성되어 있다. 이러한 구성 요소들은 소프트웨어의 전체적인 컨텍스트(context)를 구성하는데, 소프트웨어의 구성 요소 간에 종속성이 존재하는 경우, 임의 위치의 구성 요소를 변경하면 소프트웨어 전체의 컨텍스트가 망가질 수 있다. 소프트웨어의 컨텍스트는 크게 명령어와 함수의 집합으로 생성되며, 각 집합의 구성 요소는 동일 집합 내의 인접한 구성 요소에 강한 종속성을 가지게 된다.

[0004] 1 : void foo(){

[0005] 2 : instr1;

[0006] 3 : instr2;

[0007] 4 : instr3;

[0008] 5 : }

[0009] 상기 소프트웨어의 컨텍스트에서, instr2는 instr1이 실행된 이후를 가정하여 실행되며, 마찬가지로 instr3은 instr2가 실행된 이후를 가정하고 실행되는 등 각 명령어 사이에는 강력한 종속성이 존재한다. 이와 마찬가지로 함수와 함수 간에도 전달하는 인자와 반환하는 값의 종류에 대해 강력한 종속성이 존재한다.

[0010] 따라서, 소프트웨어 업데이트 시 런타임(runtime)에서의 구성 요소 변경은 전체적인 컨텍스트의 무결성을 보존하기 어려우므로, 실행 중인 구 버전 소프트웨어를 종료하고 신 버전 소프트웨어로 교체하여 전체 프로세스를 재실행하는 방법을 사용하였다. 그러나, 실시간 서비스가 중요한 분야에서는 소프트웨어의 업데이트 시 중단 없이 서비스를 제공하는 것이 요구된다.

선행기술문헌

특허문헌

[0012] (특허문헌 0001) 한국공개특허공보 제10-2003-0006140호(2003.01.23)

발명의 내용

해결하려는 과제

[0013] 본 발명의 실시예는 실시간 업데이트 할 수 있는 소프트웨어 업데이트 장치 및 방법을 제공하기 위한 것이다.

과제의 해결 수단

[0015] 개시되는 일 실시예에 따른 소프트웨어 업데이트 장치는, 하나 이상의 프로세서들, 및 상기 하나 이상의 프로세서들에 의해 실행되는 하나 이상의 프로그램들을 저장하는 메모리를 구비한 소프트웨어 업데이트 장치로서, 업데이트의 대상이 되는 소프트웨어에 대한 레퍼런스 테이블을 생성하는 테이블 생성 모듈; 상기 소프트웨어의 업데이트 이벤트가 발생하는지를 확인하고, 상기 업데이트 이벤트가 발생한 경우 업데이트의 구성 요소 종류 및 업데이트의 내용 종류 중 하나 이상을 확인하는 업데이트 확인 모듈; 및 상기 레퍼런스 테이블에 기초하여 상기 확인된 업데이트의 구성 요소 종류 및 내용 종류 중 하나 이상에 따라 상기 소프트웨어의 업데이트를 수행하는 업데이트 수행 모듈을 포함한다.

[0016] 상기 테이블 생성 모듈은, 상기 소프트웨어를 구성하는 각 함수 및 전역변수들을 추출하고, 상기 추출한 함수 및 전역변수들의 식별 정보를 상기 함수 및 전역변수들이 메모리에 저장된 주소와 매핑하여 레퍼런스 테이블을

생성할 수 있다.

- [0017] 상기 업데이트 확인 모듈은, 상기 업데이트 이벤트가 발생한 경우 상기 업데이트의 구성 요소 종류가 함수 또는 전역변수인지를 확인하고, 상기 업데이트의 내용 종류가 내적 변경 또는 외적 변경인지를 확인할 수 있다.
- [0018] 상기 함수의 내적 변경은, 함수 내부의 인스트럭션만 변경되는 경우이고, 상기 함수의 외적 변경은, 새로운 함수가 추가되거나 함수의 인터페이스가 변경되는 경우일 수 있다.
- [0019] 상기 업데이트 수행 모듈은, 상기 업데이트 이벤트가 상기 함수의 내적 변경인 경우, 업데이트된 함수를 메모리에 적재하고, 상기 레퍼런스 테이블에서 해당 함수의 주소를 업데이트된 함수의 주소로 변경하고, 상기 업데이트 이벤트가 상기 함수의 외적 변경인 경우, 업데이트된 함수를 메모리에 적재하고, 상기 레퍼런스 테이블에서 업데이트된 함수에 대한 항목을 추가할 수 있다.
- [0020] 상기 전역변수의 내적 변경은, 전역변수의 값(value)만 변경되는 경우이고, 상기 전역변수의 외적 변경은, 새로운 전역변수가 추가되거나 전역변수의 타입이 변경되는 경우일 수 있다.
- [0021] 상기 업데이트 수행 모듈은, 상기 업데이트 이벤트가 상기 전역변수의 내적 변경인 경우, 업데이트된 전역변수를 메모리에 적재하고, 상기 레퍼런스 테이블에서 해당 전역변수의 주소를 업데이트된 전역변수의 주소로 변경하고, 상기 업데이트 이벤트가 상기 전역변수의 외적 변경인 경우, 업데이트된 전역변수를 메모리에 적재하고, 상기 레퍼런스 테이블에서 업데이트된 전역변수에 대한 항목을 추가할 수 있다.
- [0022] 상기 업데이트 수행 모듈은, 상기 확인된 업데이트의 구성 요소 종류 및 내용 종류에 따라 업데이트 된 함수 또는 전역변수를 메모리에 적재하는 쓰레드 및 상기 레퍼런스 테이블을 변경하는 쓰레드를 수행하되, 상기 업데이트에 따라 발생하는 쓰레드와 사용자 명령 쓰레드 간에 우선순위를 설정할 수 있다.
- [0023] 상기 업데이트 수행 모듈은, 상기 함수 또는 전역변수를 메모리에 적재하는 쓰레드는 상기 사용자 명령 쓰레드보다 우선순위를 낮게 설정하고, 상기 레퍼런스 테이블을 변경하는 쓰레드는 상기 사용자 명령 쓰레드보다 우선순위를 높게 설정할 수 있다.
- [0024] 개시되는 일 실시예에 따른 소프트웨어 업데이트 방법은, 하나 이상의 프로세서들, 및 상기 하나 이상의 프로세서들에 의해 실행되는 하나 이상의 프로그램들을 저장하는 메모리를 구비한 컴퓨팅 장치에서 수행되는 방법으로, 업데이트의 대상이 되는 소프트웨어에 대한 레퍼런스 테이블을 생성하는 단계; 상기 소프트웨어의 업데이트 이벤트가 발생하는지를 확인하고, 상기 업데이트 이벤트가 발생한 경우 업데이트의 구성 요소 종류 및 업데이트의 내용 종류 중 하나 이상을 확인하는 단계; 및 상기 레퍼런스 테이블에 기초하여 상기 확인된 업데이트의 구성 요소 종류 및 내용 종류 중 하나 이상에 따라 상기 소프트웨어의 업데이트를 수행하는 단계를 포함한다.

발명의 효과

- [0026] 개시되는 실시예에 의하면, 소프트웨어의 각 함수 및 전역변수의 주소를 저장하고 있는 레퍼런스 테이블을 이용하여 소프트웨어의 업데이트를 수행함으로써, 수행 중인 서비스에 영향을 미치지 않으면서 실시간으로 소프트웨어를 업데이트 할 수 있게 된다.

도면의 간단한 설명

- [0028] 도 1은 본 발명의 일 실시예에 따른 소프트웨어 업데이트 장치의 구성을 나타낸 도면
- 도 2는 종래의 직접적인 메모리 참조와 개시되는 실시예에 따른 간접적인 메모리 참조를 나타낸 도면
- 도 3은 본 발명의 일 실시예에서 업데이트 구성 요소의 종류가 함수이고 업데이트 내용의 종류가 함수의 내적 변경인 경우의 업데이트 수행 상태를 개략적으로 나타낸 도면
- 도 4는 본 발명의 일 실시예에서 내적 변경된 함수가 호출되는 상태를 나타낸 도면
- 도 5는 본 발명의 일 실시예에서 업데이트 구성 요소의 종류가 함수이고 업데이트 내용의 종류가 함수의 외적 변경인 경우의 업데이트 수행 상태를 개략적으로 나타낸 도면
- 도 6은 본 발명의 일 실시예에서 외적 변경된 함수가 호출되는 상태를 나타낸 도면

도 7은 본 발명의 일 실시예에 따른 소프트웨어 업데이트 방법을 설명하기 위한 흐름도

도 8은 예시적인 실시예들에서 사용되기에 적합한 컴퓨팅 장치를 포함하는 컴퓨팅 환경을 예시하여 설명하기 위한 블록도

발명을 실시하기 위한 구체적인 내용

- [0029] 이하, 도면을 참조하여 본 발명의 구체적인 실시형태를 설명하기로 한다. 이하의 상세한 설명은 본 명세서에서 기술된 방법, 장치 및/또는 시스템에 대한 포괄적인 이해를 돕기 위해 제공된다. 그러나 이는 예시에 불과하며 본 발명은 이에 제한되지 않는다.
- [0030] 본 발명의 실시예들을 설명함에 있어서, 본 발명과 관련된 공지기술에 대한 구체적인 설명이 본 발명의 요지를 불필요하게 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명을 생략하기로 한다. 그리고, 후술되는 용어들은 본 발명에서의 기능을 고려하여 정의된 용어들로서 이는 사용자, 운용자의 의도 또는 관례 등에 따라 달라질 수 있다. 그러므로 그 정의는 본 명세서 전반에 걸친 내용을 토대로 내려져야 할 것이다. 상세한 설명에서 사용되는 용어는 단지 본 발명의 실시예들을 기술하기 위한 것이며, 결코 제한적이어서는 안 된다. 명확하게 달리 사용되지 않는 한, 단수 형태의 표현은 복수 형태의 의미를 포함한다. 본 설명에서, "포함" 또는 "구비"와 같은 표현은 어떤 특성들, 숫자들, 단계들, 동작들, 요소들, 이들의 일부 또는 조합을 가리키기 위한 것이며, 기술된 것 이외에 하나 또는 그 이상의 다른 특성, 숫자, 단계, 동작, 요소, 이들의 일부 또는 조합의 존재 또는 가능성을 배제하도록 해석되어서는 안 된다.
- [0031] 또한, 제1, 제2 등의 용어는 다양한 구성 요소들을 설명하는데 사용될 수 있지만, 상기 구성 요소들은 상기 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성 요소를 다른 구성 요소로부터 구별하는 목적으로 사용될 수 있다. 예를 들어, 본 발명의 권리 범위를 벗어나지 않으면서 제1 구성 요소는 제2 구성 요소로 명명될 수 있고, 유사하게 제2 구성 요소도 제1 구성 요소로 명명될 수 있다.
- [0033] 도 1은 본 발명의 일 실시예에 따른 소프트웨어 업데이트 장치의 구성을 나타낸 도면이다.
- [0034] 도 1을 참조하면, 소프트웨어 업데이트 장치(100)는 테이블 생성 모듈(102), 업데이트 확인 모듈(104), 및 업데이트 수행 모듈(106)을 포함할 수 있다. 소프트웨어 업데이트 장치(100)는 소프트웨어의 업데이트를 수행하기 위한 장치일 수 있다. 소프트웨어 업데이트 장치(100)는 소프트웨어가 탑재된 컴퓨팅 장치 내에 마련될 수 있다. 소프트웨어 업데이트 장치(100)는 하나 이상의 메모리를 포함할 수 있다.
- [0035] 테이블 생성 모듈(102)은 업데이트의 대상이 되는 소프트웨어에 대한 레퍼런스 테이블을 생성할 수 있다. 레퍼런스 테이블은 소프트웨어를 구성하는 각 함수 및 전역변수들에 대해 함수 및 전역변수의 식별 정보와 함수 및 전역변수의 메모리에 실제 저장된 주소 값을 매핑하여 저장한 것일 수 있다. 테이블 생성 모듈(102)은 소프트웨어를 구성하는 각 함수와 각 전역변수들을 추출하고, 추출한 함수와 전역변수들의 식별 정보를 함수와 전역변수들이 메모리에 실제 저장된 주소 값과 매핑하여 레퍼런스 테이블을 생성할 수 있다. 테이블 생성 모듈(102)은 레퍼런스 테이블을 메모리에 적재할 수 있다.
- [0036] 개시되는 실시예에서는, 레퍼런스 테이블을 통해 소프트웨어의 함수 또는 전역변수에 대해 간접적인 메모리 참조가 이루어지게 된다. 즉, 일반적으로 소프트웨어에서는 함수 또는 전역변수의 호출 시 직접적인 메모리 참조가 이루어지나, 개시되는 실시예에서는 레퍼런스 테이블을 통해 간접적인 메모리 참조가 이루어지게 된다.
- [0037] 도 2는 종래의 직접적인 메모리 참조와 개시되는 실시예에 따른 간접적인 메모리 참조를 나타낸 도면이다. 도 2를 참조하면, 종래에는 소프트웨어의 함수 또는 전역변수가 다른 함수 또는 전역변수를 호출할 때, 메모리에서 해당 함수 또는 전역변수가 저장되어 있는 곳을 직접 참조하게 된다(도 2의 (a)).
- [0038] 반면, 개시되는 실시예에서는, 소프트웨어의 함수 또는 전역변수가 다른 함수 또는 전역변수를 호출할 때, 메모리의 레퍼런스 테이블(reference table)에서 해당 함수 또는 전역변수의 실제 저장된 주소 값을 확인하는 과정을 거치게 되어 간접 참조를 하게 된다.
- [0039] 업데이트 확인 모듈(104)은 소프트웨어의 업데이트 이벤트가 발생하는지를 확인할 수 있다. 업데이트 확인 모듈(104)은 소프트웨어의 업데이트를 관리하는 외부 서버(미도시)와 연결될 수 있다. 업데이트 확인 모듈(104)은 외부 서버로부터 업데이트 이벤트가 발생하는지를 확인할 수 있다.

- [0040] 업데이트 확인 모듈(104)은 업데이트 이벤트가 발생한 경우, 업데이트의 구성 요소가 무엇인지를 확인(구성 요소의 종류 확인)할 수 있다. 즉, 업데이트 확인 모듈(104)은 업데이트의 구성 요소가 함수에 대한 것인지 아니면 전역변수에 대한 것인지를 확인할 수 있다. 업데이트 확인 모듈(104)은 업데이트의 구성 요소 종류를 확인한 후, 업데이트 내용의 종류(예를 들어, 내적 변경 또는 외적 변경)를 확인할 수 있다.
- [0041] 일 실시예에서, 업데이트 확인 모듈(104)은 업데이트의 구성 요소가 함수인 경우, 업데이트 내용이 함수의 내적 변경인지 아니면 외적 변경인지를 확인할 수 있다. 여기서, 함수의 내적 변경이라 함은 함수 내부의 인스트럭션(instruction)만 변경되는 경우를 의미할 수 있다. 또한, 함수의 외적 변경이라 함은 새로운 함수가 추가되거나 인터페이스(interface)가 변경되는 경우를 의미할 수 있다.
- [0042] 일 실시예에서, 업데이트 확인 모듈(104)은 업데이트의 구성 요소가 전역변수인 경우, 업데이트 내용이 전역변수의 내적 변경인지 아니면 외적 변경인지를 확인할 수 있다. 여기서, 전역변수의 내적 변경이라 함은 해당 전역변수의 값(value)만 변경되는 경우를 의미할 수 있다. 또한, 전역변수의 외적 변경이라 함은 새로운 전역변수가 추가되거나 전역변수의 타입이 변경되는 경우를 의미할 수 있다.
- [0043] 업데이트 수행 모듈(106)은 업데이트 확인 모듈(104)에서 확인된 사항에 따라 소프트웨어의 업데이트를 수행할 수 있다. 업데이트 수행 모듈(106)은 업데이트 구성 요소의 종류 및 업데이트 내용의 종류에 따라 소프트웨어의 업데이트를 수행할 수 있다.
- [0044] 구체적으로, 업데이트 수행 모듈(106)은 업데이트 구성 요소의 종류가 함수이고, 업데이트 내용의 종류가 함수의 내적 변경인 경우, 업데이트된 함수(즉, hotpatched function)를 메모리에 적재하고, 소프트웨어의 레퍼런스 테이블에서 해당 함수의 주소를 기존의 주소(즉, 업데이트 전 함수의 주소)에서 업데이트된 함수의 주소로 변경할 수 있다.
- [0045] 도 3은 본 발명의 일 실시예에서 업데이트 구성 요소의 종류가 함수이고 업데이트 내용의 종류가 함수의 내적 변경인 경우의 업데이트 수행 상태를 개략적으로 나타낸 도면이다. 도 3을 참조하면, 소프트웨어에서 함수 3(Func3)이 내적 변경의 업데이트가 된 경우, 업데이트 수행 모듈(106)은 업데이트된 함수(즉, hotpatched function)를 메모리에 적재하고(①), 레퍼런스 테이블에서 함수 3(Func3)의 주소를 업데이트 된 메모리의 주소로 변경할 수 있다(②). 이때, 업데이트 된 함수는 기존의 함수와는 메모리의 다른 영역에 적재될 수 있다.
- [0046] 도 4는 본 발명의 일 실시예에서 내적 변경된 함수가 호출되는 상태를 나타낸 도면이다. 도 4를 참조하면, 소프트웨어에서 함수 1(Func1)이 함수 3(Func3)을 호출하는 경우를 보면, 함수 1(Func1)은 레퍼런스 테이블에서 함수 3(Func3)의 변경된 주소 값을 확인하여 업데이트 된 함수3(Func3_hopatched)을 호출할 수 있다.
- [0047] 여기서, 업데이트 전 함수 3(Func3_legacy)의 저장 영역(주소)은 메모리의 text 영역이었으나, 업데이트된 후에는 그와는 다른 메모리의 영역에 적재될 수 있고, 그에 따라 레퍼런스 테이블에서 함수 3(Func3)의 주소를 업데이트 된 메모리의 주소로 변경하게 된다.
- [0048] 한편, 업데이트 구성 요소의 종류가 전역변수이고, 업데이트 내용의 종류가 전역변수의 내적 변경(예를 들어, 전역변수의 값의 변경)인 경우에도, 함수의 내적 변경과 동일 또는 유사하게 업데이트를 수행할 수 있다. 즉, 업데이트 수행 모듈(106)은 업데이트된 전역변수를 메모리에 적재하고, 소프트웨어의 레퍼런스 테이블에서 해당 전역변수의 주소를 기존의 주소에서 업데이트 된 전역변수의 주소로 변경할 수 있다.
- [0049] 또한, 업데이트 수행 모듈(106)은 업데이트 구성 요소의 종류가 함수이고, 업데이트 내용의 종류가 함수의 외적 변경인 경우, 함수 간 종속성에 문제가 발생할 수 있으므로, 업데이트 수행 모듈(106)은 업데이트된 함수(즉, hotpatched function)를 메모리에 적재하고, 소프트웨어의 레퍼런스 테이블에 업데이트된 함수에 대한 항목(함수의 식별 정보 및 함수의 메모리에 실제 저장된 주소 값)을 추가할 수 있다.
- [0050] 도 5는 본 발명의 일 실시예에서 업데이트 구성 요소의 종류가 함수이고 업데이트 내용의 종류가 함수의 외적 변경인 경우의 업데이트 수행 상태를 개략적으로 나타낸 도면이다. 도 5를 참조하면, 소프트웨어에서 함수 3(Func3)이 외적 변경의 업데이트가 된 경우, 업데이트 수행 모듈(106)은 업데이트된 함수(즉, hotpatched function)를 메모리에 적재하고(①), 레퍼런스 테이블에서 업데이트 된 함수 3(Func3')에 대해 항목을 추가할 수 있다(②). 이때, 업데이트 된 함수는 기존의 함수와는 메모리의 다른 영역에 적재될 수 있다.
- [0051] 도 6은 본 발명의 일 실시예에서 외적 변경된 함수가 호출되는 상태를 나타낸 도면이다. 여기서는, 함수 1(Func1)이 함수 3(Func3)을 호출하는 경우를 실시예로 하였으며, 함수 3(Func3)이 외적 변경(예를 들어, 인터페이스 변경)된 경우를 나타내었다.

- [0052] 도 6을 참조하면, 함수 3(Func3)이 외적 변경된 경우, 외적 변경된 함수 3(Func3)을 호출하는 함수 1(Func1)도 그에 맞게 업데이트되어야 한다. 이때, 업데이트 된 함수 1(Func1_hotpatched)은 메모리의 일정 영역에 적재될 수 있다. 여기서, 업데이트 된 함수 1(Func1_hotpatched)이 외적 변경 된 함수 3(Func1_hotpatched)을 호출하는 경우, 레퍼런스 테이블에서 새로 추가된 외적 변경 된 함수 3(Func3')의 항목을 찾아 그 주소 값을 확인 한 후, 외적 변경 된 함수 3(Func1_hotpatched)을 호출할 수 있다. 만약, 기존 버전의 함수 1(Func1_legacy)이 함수 3을 호출하는 경우에는 레퍼런스 테이블에서 기존 버전의 함수 3(Func3)의 항목을 찾아 그 주소 값을 확인한 후, 기존 버전의 함수 3(Func3_legacy)을 호출할 수 있다.
- [0053] 한편, 업데이트 구성 요소의 종류가 전역변수이고, 업데이트 내용의 종류가 전역변수의 외적 변경(예를 들어, 전역변수의 타입의 변경 또는 새로운 전역변수의 추가 등)인 경우에도, 함수의 외적 변경과 동일 또는 유사하게 업데이트를 수행할 수 있다. 즉, 업데이트 수행 모듈(106)은 업데이트된 전역변수를 메모리에 적재하고, 소프트웨어의 레퍼런스 테이블에서 업데이트된 전역변수에 대해 항목을 추가할 수 있다.
- [0054] 업데이트 수행 모듈(106)은 소프트웨어의 업데이트 시 컴퓨팅 장치에서 사용자의 명령에 따라 수행되고 있는 스레드(thread)(사용자 명령 스레드)가 있는지를 확인할 수 있다. 업데이트 수행 모듈(106)은 소프트웨어의 업데이트에 따라 발생하는 스레드와 사용자 명령 스레드 간에 우선순위를 설정할 수 있다. 이때, 업데이트 수행 모듈(106)은 소프트웨어의 업데이트에 따라 발생하는 스레드의 종류에 따라 사용자 명령 스레드와의 우선순위를 다르게 설정할 수 있다.
- [0055] 구체적으로, 업데이트 수행 모듈(106)은 업데이트 된 함수 또는 전역변수를 메모리에 적재할 때, 사용자 명령 스레드가 있는지를 확인하고, 사용자 명령 스레드가 있는 경우, 업데이트 된 함수 또는 전역 변수를 메모리에 적재하는 스레드의 우선순위를 사용자 명령 스레드의 우선순위보다 낮게 설정할 수 있다.
- [0056] 또한, 업데이트 수행 모듈(106)은 함수 또는 전역변수의 업데이트에 따라 메모리의 레퍼런스 테이블을 변경해야 하는 경우, 레퍼런스 테이블 변경 스레드의 우선순위를 사용자 명령 스레드의 우선순위보다 높게 설정할 수 있다. 즉, 레퍼런스 테이블 변경은 사용자 명령 스레드가 있다 하더라도 그 보다 우선하여 실행될 수 있도록 할 수 있다.
- [0057] 개시되는 실시예에 의하면, 소프트웨어의 각 함수 및 전역변수의 주소를 저장하고 있는 레퍼런스 테이블을 이용하여 소프트웨어의 업데이트를 수행함으로써, 수행 중인 서비스에 영향을 미치지 않으면서 실시간으로 소프트웨어를 업데이트 할 수 있게 된다. 이 경우, 서비스 중단이 어려운 발전소, 사물 인터넷(IoT) 등과 같은 실시간 시스템에서 소프트웨어 서비스를 상시 제공할 수 있게 된다.
- [0058] 본 명세서에서 모듈이라 함은, 본 발명의 기술적 사상을 수행하기 위한 하드웨어 및 상기 하드웨어를 구동하기 위한 소프트웨어의 기능적, 구조적 결합을 의미할 수 있다. 예컨대, 상기 "모듈"은 소정의 코드와 상기 소정의 코드가 수행되기 위한 하드웨어 리소스의 논리적인 단위를 의미할 수 있으며, 반드시 물리적으로 연결된 코드를 의미하거나, 한 종류의 하드웨어를 의미하는 것은 아니다.
- [0060] 도 7은 본 발명의 일 실시예에 따른 소프트웨어 업데이트 방법을 설명하기 위한 흐름도이다. 도시된 흐름도에서는 상기 방법을 복수 개의 단계로 나누어 기재하였으나, 적어도 일부의 단계들은 순서를 바꾸어 수행되거나, 다른 단계와 결합되어 함께 수행되거나, 생략되거나, 세부 단계들로 나뉘어 수행되거나, 또는 도시되지 않은 하나 이상의 단계가 부가되어 수행될 수 있다.
- [0061] 도 7을 참조하면, 소프트웨어 업데이트 장치(100)는 업데이트의 대상이 되는 소프트웨어에 대한 레퍼런스 테이블을 생성할 수 있다(S 101). 소프트웨어 업데이트 장치(100)는 소프트웨어를 구성하는 각 함수와 각 전역변수들을 추출하고, 추출한 함수와 전역변수들의 식별 정보를 함수와 전역변수들이 메모리에 실제 저장된 주소 값과 매핑하여 레퍼런스 테이블을 생성할 수 있다.
- [0062] 다음으로, 소프트웨어 업데이트 장치(100)는 소프트웨어의 업데이트 이벤트가 발생하는지를 확인하고(S 103), 업데이트 이벤트가 발생하는 경우 업데이트의 구성 요소 종류 및 업데이트 내용 종류를 확인할 수 있다(S 105). 다음으로, 소프트웨어 업데이트 장치(100)는 확인된 업데이트의 구성 요소 종류 및 업데이트 내용 종류에 따라 레퍼런스 테이블을 이용하여 업데이트를 수행할 수 있다(S 107).
- [0064] 도 8은 예시적인 실시예들에서 사용되기에 적합한 컴퓨팅 장치를 포함하는 컴퓨팅 환경(10)을 예시하여 설명하

기 위한 블록도이다. 도시된 실시예에서, 각 컴포넌트들은 이하에 기술된 것 이외에 상이한 기능 및 능력을 가질 수 있고, 이하에 기술된 것 이외에도 추가적인 컴포넌트를 포함할 수 있다.

[0065] 도시된 컴퓨팅 환경(10)은 컴퓨팅 장치(12)를 포함한다. 일 실시예에서, 컴퓨팅 장치(12)는 소프트웨어 업데이트 장치(100)일 수 있다.

[0066] 컴퓨팅 장치(12)는 적어도 하나의 프로세서(14), 컴퓨터 판독 가능 저장 매체(16) 및 통신 버스(18)를 포함한다. 프로세서(14)는 컴퓨팅 장치(12)로 하여금 앞서 언급된 예시적인 실시예에 따라 동작하도록 할 수 있다. 예컨대, 프로세서(14)는 컴퓨터 판독 가능 저장 매체(16)에 저장된 하나 이상의 프로그램들을 실행할 수 있다. 상기 하나 이상의 프로그램들은 하나 이상의 컴퓨터 실행 가능 명령어를 포함할 수 있으며, 상기 컴퓨터 실행 가능 명령어는 프로세서(14)에 의해 실행되는 경우 컴퓨팅 장치(12)로 하여금 예시적인 실시예에 따른 동작들을 수행하도록 구성될 수 있다.

[0067] 컴퓨터 판독 가능 저장 매체(16)는 컴퓨터 실행 가능 명령어 내지 프로그램 코드, 프로그램 데이터 및/또는 다른 적합한 형태의 정보를 저장하도록 구성된다. 컴퓨터 판독 가능 저장 매체(16)에 저장된 프로그램(20)은 프로세서(14)에 의해 실행 가능한 명령어의 집합을 포함한다. 일 실시예에서, 컴퓨터 판독 가능 저장 매체(16)는 메모리(랜덤 액세스 메모리와 같은 휘발성 메모리, 비휘발성 메모리, 또는 이들의 적절한 조합), 하나 이상의 자기 디스크 저장 디바이스들, 광학 디스크 저장 디바이스들, 플래시 메모리 디바이스들, 그 밖에 컴퓨팅 장치(12)에 의해 액세스되고 원하는 정보를 저장할 수 있는 다른 형태의 저장 매체, 또는 이들의 적합한 조합일 수 있다.

[0068] 통신 버스(18)는 프로세서(14), 컴퓨터 판독 가능 저장 매체(16)를 포함하여 컴퓨팅 장치(12)의 다른 다양한 컴포넌트들을 상호 연결한다.

[0069] 컴퓨팅 장치(12)는 또한 하나 이상의 입출력 장치(24)를 위한 인터페이스를 제공하는 하나 이상의 입출력 인터페이스(22) 및 하나 이상의 네트워크 통신 인터페이스(26)를 포함할 수 있다. 입출력 인터페이스(22) 및 네트워크 통신 인터페이스(26)는 통신 버스(18)에 연결된다. 입출력 장치(24)는 입출력 인터페이스(22)를 통해 컴퓨팅 장치(12)의 다른 컴포넌트들에 연결될 수 있다. 예시적인 입출력 장치(24)는 포인팅 장치(마우스 또는 트랙패드 등), 키보드, 터치 입력 장치(터치패드 또는 터치스크린 등), 음성 또는 소리 입력 장치, 다양한 종류의 센서 장치 및/또는 촬영 장치와 같은 입력 장치, 및/또는 디스플레이 장치, 프린터, 스피커 및/또는 네트워크 카드와 같은 출력 장치를 포함할 수 있다. 예시적인 입출력 장치(24)는 컴퓨팅 장치(12)를 구성하는 일 컴포넌트로서 컴퓨팅 장치(12)의 내부에 포함될 수도 있고, 컴퓨팅 장치(12)와는 구별되는 별개의 장치로 컴퓨팅 장치(12)와 연결될 수도 있다.

[0071] 이상에서 본 발명의 대표적인 실시예들을 상세하게 설명하였으나, 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자는 상술한 실시예에 대하여 본 발명의 범주에서 벗어나지 않는 한도 내에서 다양한 변형이 가능함을 이해할 것이다. 그러므로 본 발명의 권리범위는 설명된 실시예에 국한되어 정해져서는 안 되며, 후술하는 특허 청구범위뿐만 아니라 이 특허청구범위와 균등한 것들에 의해 정해져야 한다.

부호의 설명

[0073] 10 : 컴퓨팅 환경

12 : 컴퓨팅 장치

14 : 프로세서

16 : 컴퓨터 판독 가능 저장 매체

18 : 통신 버스

20 : 프로그램

22 : 입출력 인터페이스

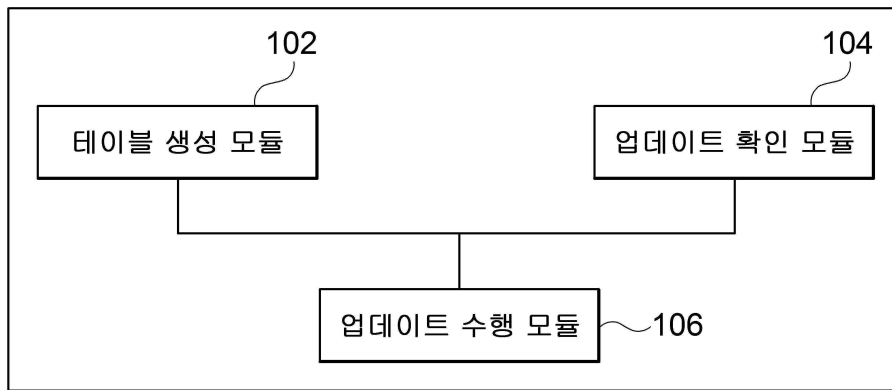
24 : 입출력 장치

- 26 : 네트워크 통신 인터페이스
- 100 : 소프트웨어 업데이트 장치
- 102 : 테이블 생성 모듈
- 104 : 업데이트 확인 모듈
- 106 : 업데이트 수행 모듈

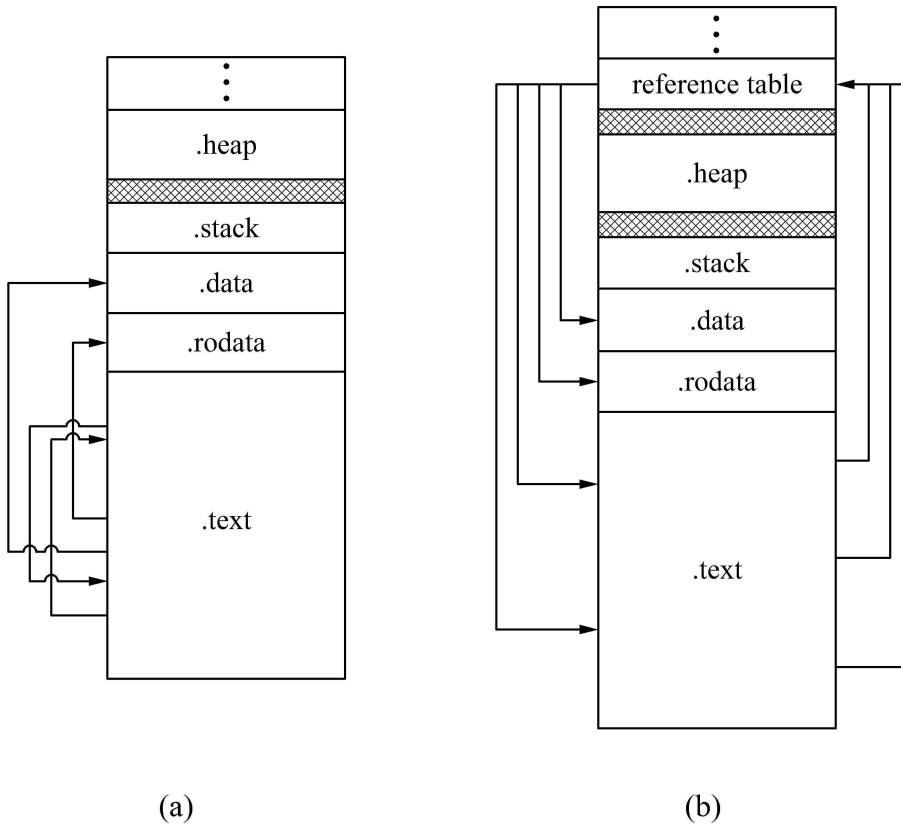
도면

도면1

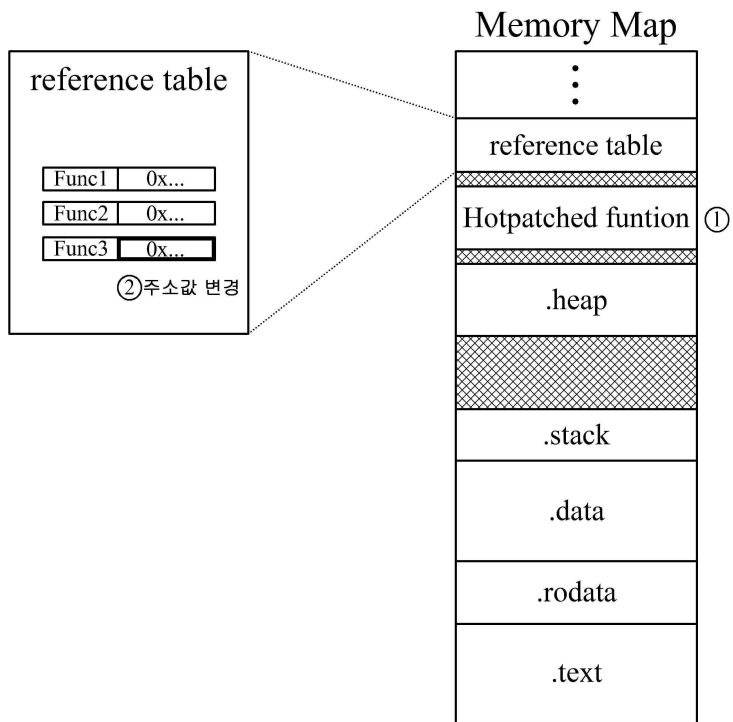
100



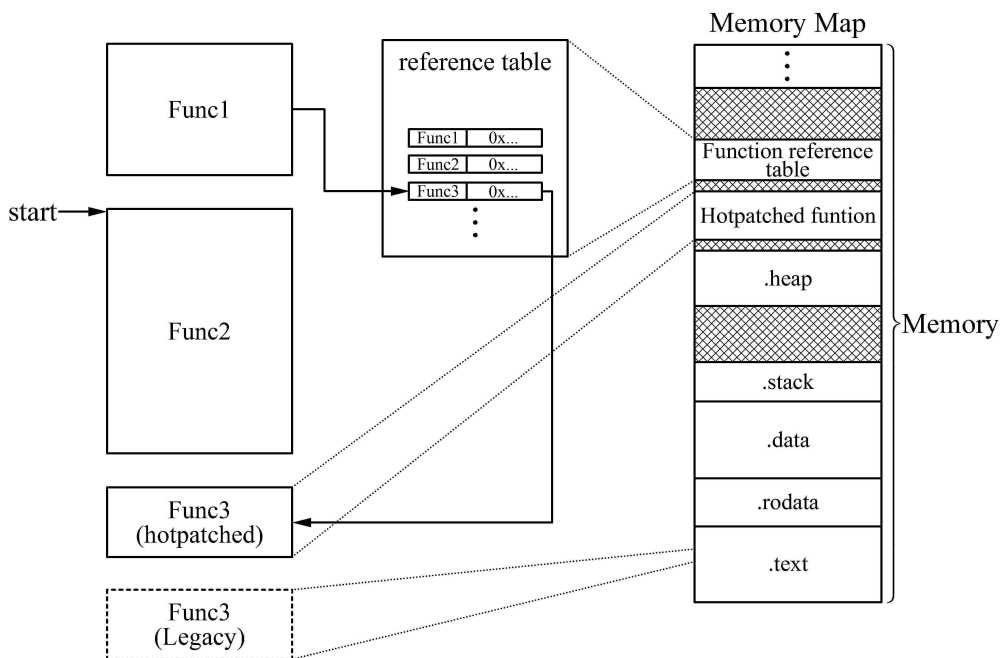
도면2



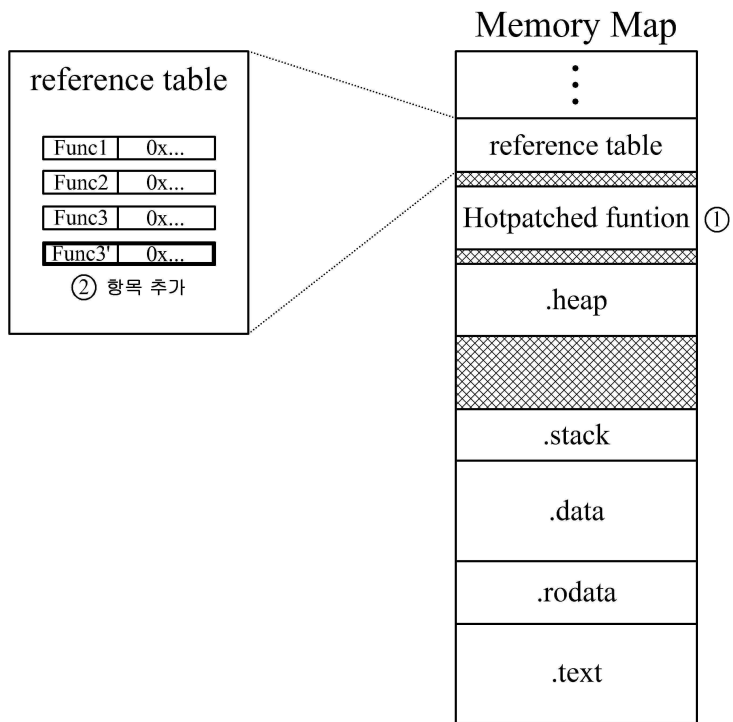
도면3



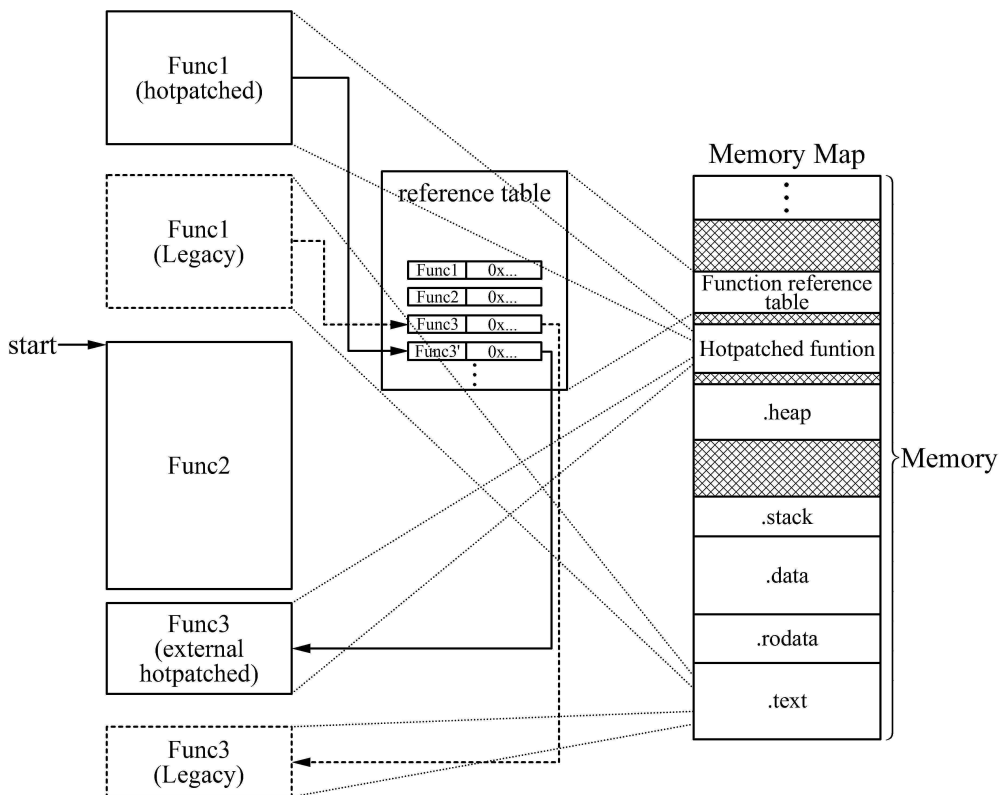
도면4



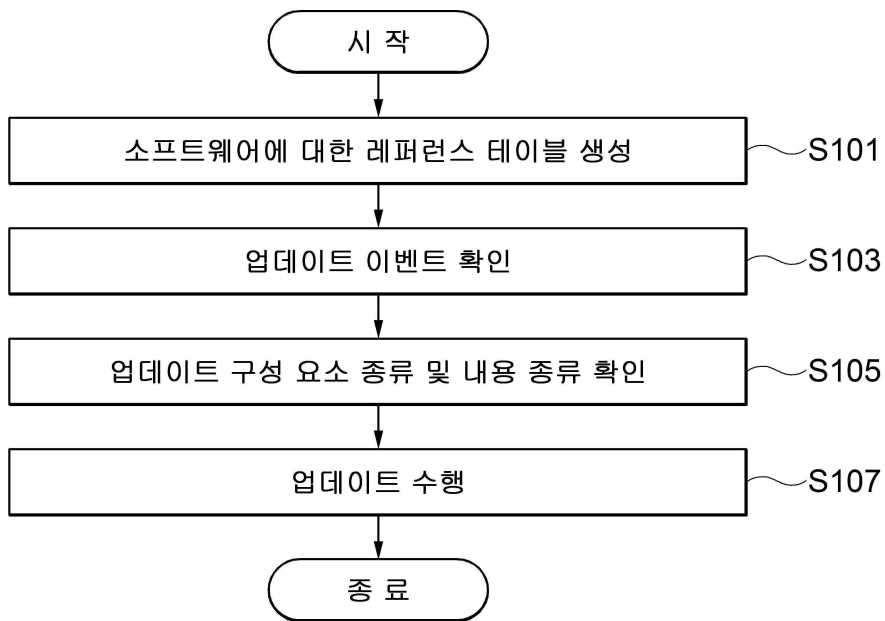
도면5



도면6



도면7



도면8

10

