



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2014년07월14일
(11) 등록번호 10-1417759
(24) 등록일자 2014년07월03일

(51) 국제특허분류(Int. Cl.)
G06F 15/00 (2006.01) G06F 9/06 (2006.01)
G06F 15/16 (2006.01)
(21) 출원번호 10-2007-0059550
(22) 출원일자 2007년06월18일
심사청구일자 2012년06월18일
(65) 공개번호 10-2007-0120446
(43) 공개일자 2007년12월24일
(30) 우선권주장
1020060054746 2006년06월19일 대한민국(KR)
(56) 선행기술조사문헌
US20050010576 A1*
W02005119432 A2*
JP2005011209 A
JP평성05120067 A
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
삼성전자주식회사
경기도 수원시 영통구 삼성로 129 (매탄동)
(72) 발명자
오성조
경기도 수원시 권선구 덕영대로1323번길 25-33,
114동 403호 (권선동, 우남아파트)
(74) 대리인
윤동열

전체 청구항 수 : 총 26 항

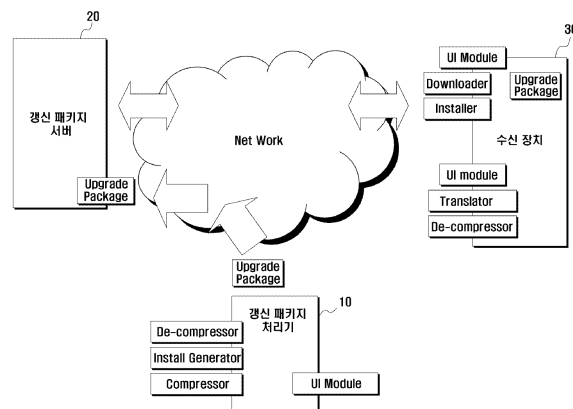
심사관 : 박승철

(54) 발명의 명칭 시스템 정보 갱신 장치 및 방법

(57) 요약

시스템의 버전을 갱신하는 방법이, 제1버전 및 제2버전을 입력받아, 상기 두 버전의 차에 따른 갱신 패키지를 생성하는 과정과, 상기 갱신패키지의 생성을 하나 이상의 수신시스템들에 통지하는 과정과, 상기 갱신패키지를 다운로드하며, 상기 제1버전을 저장하는 제1메모리에 상기 다운로드되는 갱신패키지를 저장하고, 갱신 요구시 상기 제1버전에 갱신패키지를 합성하여 제2버전을 생성하여 제2메모리에 저장하고, 상기 제2메모리의 제2버전을 통해 수신 시스템을 운용하는 과정으로 이루어진다.

대표도



특허청구의 범위

청구항 1

시스템의 버전을 갱신하는 방법에 있어서,

제1버전 및 제2버전을 입력받아, 상기 두 버전의 차에 따른 갱신 패키지를 생성하는 과정과,

상기 갱신패키지의 생성을 하나 이상의 수신시스템들에 통지하는 과정과,

상기 갱신패키지를 다운로드하며, 상기 제1버전을 저장하는 제1메모리에 상기 다운로드되는 갱신패키지를 저장하고, 갱신 요구시 상기 제1버전에 갱신패키지를 합성하여 제2버전을 생성하여 제2메모리에 저장하고, 상기 제2메모리의 제2버전을 통해 수신 시스템을 운용하는 과정으로 이루어지되,

상기 제1메모리에 상기 다운로드되는 갱신패키지를 저장하는 과정은

다수 번의 시스템의 버전 업데이트에 따라 다수의 갱신 패키지를 다운로드하여 저장하는 과정을 포함하며,

상기 제2버전을 생성하는 과정은

상기 제1버전과 상기 다수의 갱신패키지 중 선택된 갱신 패키지를 합성하여 상기 제2버전을 생성하는 과정을 포함함을 특징으로 하는 상기 시스템의 버전 갱신 방법.

청구항 2

제1항에 있어서, 상기 갱신패키지는 제1버전과 결합하기 위한 히스토리 데이터, 상기 갱신패키지를 상기 제1버전에 결합하기 위한 맵핑 정보들로 이루어지는 맵 데이터와, 상기 맵 데이터에 따라 상기 제1버전에 결합되는 갱신데이터들로 이루어짐을 특징으로 하는 상기 시스템 버전 갱신 방법.

청구항 3

제2항에 있어서, 상기 맵 데이터의 상기 맵핑 정보는 명령어 및 블록 인덱스 정보로 구성되며, 상기 명령어는 블록 인덱스 및 데이터가 동일하면 복사, 제2버전의 블록데이터가 삽입 또는 변경되었으면 변경, 상기 제2버전의 블록데이터가 일정 블록 이동되어 제1버전의 블록데이터와 동일하면 이동 명령어로 발생되며, 상기 갱신데이터는 상기 변경 명령어의 블록 인덱스에 대응되는 데이터임을 특징으로 하는 상기 시스템의 버전 갱신 방법.

청구항 4

제3항에 있어서, 상기 갱신패키지를 생성하는 과정이,

상기 입력된 제1 및 제2버전을 설정된 크기의 블록으로 구분하고, 상기 두 버전의 블록 데이터들을 비교하는 과정을 더 구비하며,

상기 맵데이터에서 변경 명령어 블록 데이터를 갱신데이터로 생성하는 과정과,

상기 히스토리 데이터, 맵데이터 및 갱신데이터를 결합하여 갱신패키지로 생성하는 과정으로 이루어짐을 특징으로 하는 상기 시스템의 버전 갱신 방법.

청구항 5

제3항에 있어서, 상기 갱신패키지를 생성하는 과정이,

상기 입력된 제1 및 제2버전을 설정된 크기로 구분하며, 상기 구분된 블록 데이터들을 압축하는 과정과,

상기 압축된 제1 및 제2버전 블록 데이터를 비교하는 과정을 더 구비하며,

상기 맵데이터에서 변경 명령어 블록 데이터를 갱신데이터로 생성하는 과정과,

상기 히스토리 데이터, 맵데이터 및 갱신데이터를 결합하여 갱신패키지로 생성하는 과정으로 이루어짐을 특징으로 하는 상기 시스템의 버전 갱신 방법.

청구항 6

제5항에 있어서, 상기 갱신패키지를 압축 부호화하는 과정을 더 구비함을 특징으로 하는 상기 시스템의 버전 갱신 방법.

청구항 7

제3항에 있어서, 상기 갱신패키지를 다운로드하여 제2버전을 생성하는 과정이,

상기 다운로드되는 갱신패키지를 불휘발성의 상기 제1메모리의 갱신패키지 저장 영역에 인스톨하며, 상기 제1메모리는 상기 제1버전을 저장하는 영역 및 적어도 하나의 갱신패키지 저장영역을 구비하는 과정과,

갱신 요구시 상기 갱신패키지를 상기 제1버전에 맵핑하여 제2버전을 생성하며, 상기 생성된 제2버전을 휘발성의 상기 제2메모리에 저장하는 과정과,

상기 제2메모리에 저장된 제2버전에 의해 상기 시스템을 운용하는 과정으로 이루어짐을 특징으로 하는 상기 시스템의 버전 갱신 방법.

청구항 8

제7항에 있어서, 상기 제2버전을 생성하는 과정이,

상기 맵데이터를 분석하는 과정과,

상기 맵데이터의 명령어가 복사이면 블록 인덱스에 해당하는 상기 제1버전의 블록 데이터를 상기 제2버전으로 복사하는 과정과,

상기 맵데이터의 명령어가 이동이면 상기 블록 인덱스에 대응되는 블록만큼 쉬프트된 상기 제1버전의 블록 데이터를 상기 제2버전에 이동하여 저장하는 과정과,

상기 맵데이터가 변경이면 상기 블록 인덱스에 대응되는 제2버전으로 갱신데이터를 삽입하는 과정으로 이루어짐을 특징으로 하는 상기 시스템의 버전 갱신 방법.

청구항 9

제8항에 있어서, 상기 갱신요구는 상기 갱신패키지를 저장한 후 이를 표시하고, 사용자의 요구에 의해 전원온시 시스템 초기화 수행 후 수행됨을 특징으로 하는 상기 시스템의 버전 갱신 방법.

청구항 10

제8항에 있어서, 상기 제1 및 제2버전은 프로그램 버전인 것을 특징으로 하는 상기 시스템의 버전 갱신 방법.

청구항 11

시스템의 버전을 갱신하는 방법에 있어서,

제1버전 및 제2버전을 입력받아, 상기 두 버전의 차에 따른 갱신 패키지를 생성하는 과정과,

상기 갱신패키지의 생성을 하나 이상의 수신시스템들에 통지하는 과정과,

상기 갱신패키지를 다운로드하며, 상기 제1버전 프로그램을 저장하는 제1메모리에 상기 다운로드되는 갱신패키지를 저장하고, 갱신 요구시 상기 제1버전 프로그램에 갱신패키지를 합성하여 제2버전을 생성하여 제2메모리에

저장하고, 상기 제2메모리의 제2버전을 통해 수신 시스템을 운용하는 과정으로 이루어지며,
 상기 갱신패키지를 생성하는 과정은,
 제1 및 제2버전을 입력하는 과정과,
 상기 입력된 제1 및 제2버전을 비교하는 과정과,
 상기 비교 결과에 따라 갱신데이터를 인스톨하기 위한 인스톨 데이터를 생성하는 과정과,
 상기 비교 결과에 따라 갱신데이터를 생성하며, 상기 인스톨데이터 및 갱신데이터를 결합하여 상기 갱신패키지를 생성하는 과정으로 이루어지고,
 상기 제1메모리에 상기 다운로드되는 갱신패키지를 저장하는 과정은
 다수 번의 시스템의 버전 업데이트에 따라 다수의 갱신 패키지를 다운로드하여 저장하는 과정을 포함하며,
 상기 제2버전을 생성하는 과정은
 상기 제1버전과 상기 다수의 갱신패키지 중 선택된 갱신 패키지를 합성하여 상기 제2버전을 생성하는 과정을 포함함을 특징으로 하는 상기 시스템의 버전 갱신 방법.

청구항 12

시스템의 버전을 갱신하는 방법에 있어서,
 제1버전 및 제2버전을 입력받아, 상기 두 버전의 차에 따른 갱신 패키지를 생성하는 과정과,
 상기 갱신패키지의 생성을 하나 이상의 수신시스템들에 통지하는 과정과,
 상기 갱신패키지를 다운로드하며, 상기 제1버전을 저장하는 제1메모리에 상기 다운로드되는 갱신패키지를 저장하고, 갱신 요구시 상기 제1버전에 갱신패키지를 합성하여 제2버전을 생성하여 제2메모리에 저장하고, 상기 제2메모리의 제2버전을 통해 수신 시스템을 운용하는 과정으로 이루어지며,
 상기 제1메모리에 상기 다운로드되는 갱신패키지를 저장하는 과정은
 다수 번의 시스템의 버전 업데이트에 따라 다수의 갱신 패키지를 다운로드하여 저장하는 과정을 포함하며,
 상기 제2버전을 생성하는 과정은,
 상기 다수의 갱신패키지를 제1메모리에 인스톨하는 과정과,
 갱신요구시 상기 제1메모리의 제1버전 및 상기 다수의 갱신패키지 중 선택된 갱신패키지를 로드하고, 상기 로드된 제1버전에 상기 선택된 갱신패키지를 결합하여 상기 제2버전을 생성하여 제2메모리에 저장하는 과정과,
 상기 제2메모리에 저장된 제2버전에 의해 시스템을 구동하는 과정으로 이루어짐을 특징으로 하는 상기 시스템의 버전 갱신 방법.

청구항 13

시스템의 버전을 갱신하는 방법에 있어서,
 제1 및 제2버전을 입력하는 과정과, 상기 입력된 제1 및 제2버전을 비교하는 과정과, 상기 비교 결과에 따라 갱신데이터를 인스톨하기 위한 인스톨 데이터를 생성하는 과정과, 상기 비교 결과에 따라 갱신데이터를 생성하며, 상기 인스톨데이터 및 갱신데이터를 결합하여 갱신패키지를 생성하는 과정으로 이루어져 갱신 패키지를 생성하는 과정과,
 상기 갱신패키지의 생성을 하나 이상의 수신시스템들에 통지하는 과정과,
 다수 번의 시스템의 버전 업데이트에 따라 다수의 갱신패키지를 다운로드하여 저장하는 과정과, 상기 다수의 갱신패키지를 제1메모리에 인스톨하는 과정과, 갱신요구시 상기 제1메모리의 제1버전 및 상기 다수의 갱신패키지 중 선택된 갱신패키지를 로드하고, 상기 로드된 제1버전에 상기 선택된 갱신패키지를 결합하여 상기 제2버전을 생성하여 제2메모리에 저장하는 과정과, 상기 제2메모리에 저장된 제2버전에 의해 시스템을 구동하는 과정으로 이루어져 시스템의 프로그램을 갱신하는 과정으로 이루어짐을 특징으로 하는 상기 시스템의 버전 갱신 방법.

청구항 14

시스템의 버전 갱신하는 장치에 있어서,

제1버전 및 제2버전을 입력받아, 상기 두 버전 정보들의 차에 따른 갱신 패키지를 생성하는 갱신패키지처리기와,

상기 갱신패키지의 생성을 하나 이상의 수신시스템들에 통지하는 갱신패키지서버와,

다수 번의 시스템의 버전 업데이트에 따라 다수의 갱신패키지를 다운로드하며, 상기 제1버전을 저장하는 제1메모리에 상기 다운로드되는 다수의 갱신패키지를 저장하고, 갱신 요구시 상기 제1버전에 상기 다수의 갱신패키지 중 선택된 갱신패키지를 합성하여 제2버전을 생성하여 제2메모리에 저장하고, 상기 제2메모리의 제2버전을 통해 시스템을 운용하는 수신시스템으로 구성된 것을 특징으로 하는 시스템의 버전 갱신장치.

청구항 15

제14항에 있어서, 상기 갱신패키지처리기는

제1버전과 결합하기 위한 히스토리 데이터 및 상기 제2버전의 갱신패키지를 상기 제1버전에 결합하기 위한 매핑 정보들로 이루어지는 맵 데이터를 발생하는 인스톨데이터발생기와,

상기 맵 데이터에 따라 상기 제1버전에 결합되는 갱신데이터를 발생하며, 상기 갱신데이터 및 상기 인스톨데이터를 결합하여 갱신패키지를 생성하는 패키지발생기로 구성된 것을 특징으로 하는 상기 시스템 버전 갱신 장치.

청구항 16

제15항에 있어서, 상기 맵 데이터의 상기 맵핑 정보는 명령어 및 블록 인덱스 정보로 구성되며, 상기 명령어는 블록 인덱스 및 데이터가 동일하면 복사, 제2버전의 블록데이터가 삽입 또는 변경되었으면 변경, 상기 제2버전의 블록데이터가 일정 블록 이동되어 제1버전의 블록데이터와 동일하면 이동 명령어로 발생되며, 상기 갱신데이터는 상기 변경 명령어의 블록 인덱스에 대응되는 데이터임을 특징으로 하는 상기 시스템의 버전 갱신 장치.

청구항 17

제16항에 있어서, 상기 갱신패키지처리기는

상기 입력된 제1 및 제2버전의 프로그램들을 설정된 크기의 블록으로 구분하고, 상기 두 버전의 블록 데이터들을 비교하는 비교기를 더 구비하며,

상기 맵데이터에서 변경 명령어 블록 데이터를 갱신데이터로 생성한 후, 상기 히스토리 데이터, 맵데이터 및 갱신데이터를 결합하여 갱신패키지로 생성하는 것을 특징으로 하는 상기 시스템의 버전 갱신 장치.

청구항 18

제17항에 있어서, 상기 갱신패키지처리기는,

상기 입력된 제1 및 제2버전을 설정된 크기로 구분하며, 상기 구분된 블록 데이터들을 압축하는 압축기를 더 구비하는 것을 특징으로 하는 상기 시스템의 버전 갱신 장치.

청구항 19

제18항에 있어서, 상기 갱신패키지처리기는 상기 갱신패키지를 압축 부호화하는 제2압축기를 더 구비하는 것을 특징으로 하는 상기 시스템의 버전 갱신 장치.

청구항 20

제16항에 있어서, 상기 수신시스템이,
상기 제1버전을 저장하는 영역 및 적어도 하나의 갱신팬키지 저장영역을 구비하는 제1메모리와,
갱신된 제2버전을 저장하는 제2메모리와,
상기 다운로드되는 갱신팬키지를 상기 제1메모리의 갱신팬키지 저장 영역에 저장하는 인스톨러와,
갱신 요구시 상기 갱신팬키지를 상기 제1버전에 맵핑하여 제2버전을 생성하며, 상기 생성된 제2버전을 상기 제2메모리에 저장하는 트랜스레이터로 구성된 것을 특징으로 하는 상기 시스템의 버전 갱신 장치.

청구항 21

제20항에 있어서, 상기 트랜스레이터가,
상기 맵데이터를 로드한 후, 상기 맵데이터의 명령어가 복사이면 블록 인덱스에 해당하는 상기 제1버전의 프로그램의 블록 데이터를 상기 제2 버전에 복사하고, 상기 맵데이터의 명령어가 이동이면 상기 블록 인덱스에 대응되는 블록만큼 쉬프트된 상기 제1버전의 블록 데이터를 상기 제2버전에 이동하여 저장하며, 상기 맵데이터가 변경이면 상기 블록 인덱스에 대응되는 제2버전에 갱신타데이터를 삽입하여 상기 제1버전을 제2버전으로 갱신하는 것을 특징으로 하는 상기 시스템의 버전 갱신 장치.

청구항 22

제21항에 있어서, 상기 갱신요구는 상기 갱신팬키지를 인스톨한 후 이를 표시하고, 사용자의 요구에 의해 발생됨을 특징으로 하는 상기 시스템의 버전 갱신 장치.

청구항 23

제21항에 있어서, 상기 갱신요구는 전원온 시 시스템 초기화 수행 후 수행됨을 특징으로 하는 상기 시스템의 버전 갱신 장치.

청구항 24

시스템의 버전을 갱신하는 장치에 있어서,
제1버전 및 제2버전을 입력받아, 상기 두 버전의 차에 따른 갱신 패키지를 생성하는 갱신팬키지처리기와,
상기 갱신팬키지의 생성을 하나 이상의 수신시스템들에 통지하는 갱신팬키지서버와,
다수 번의 시스템의 버전 업데이트에 따라 다수의 갱신팬키지를 다운로드하며, 상기 제1버전 프로그램을 저장하는 제1메모리에 상기 다운로드되는 다수의 갱신팬키지를 저장하고, 갱신 요구시 상기 제1버전에 상기 다수의 갱신팬키지 중 선택된 갱신팬키지를 합성하여 제2버전을 생성하여 제2메모리에 저장하고, 상기 제2메모리의 제2버전을 통해 시스템을 운용하는 수신시스템으로 구성되며,
상기 갱신팬키지처리기는,
제1 및 제2버전을 입력하여 비교하는 비교기와,
상기 비교 결과에 따라 갱신타데이터를 인스톨하기 위한 인스톨 데이터를 생성하는 인스톨데이터발생기와,
상기 비교 결과에 따라 갱신타데이터를 생성하며, 상기 인스톨데이터 및 갱신타데이터를 결합하여 상기 갱신팬키지를 생성하는 패키지발생기로 구성된 것을 특징으로 하는 상기 시스템의 버전 갱신 장치.

청구항 25

시스템의 프로그램을 갱신하는 장치에 있어서,
제1버전 및 제2버전을 입력받아, 상기 두 버전의 차에 따른 갱신 패키지를 생성하는 갱신팬키지처리기와,

상기 갱신패키지의 생성을 하나 이상의 수신시스템들에 통지하는 갱신패키지서버와,

다수 번의 시스템의 버전 업데이트에 따라 다수의 갱신패키지를 다운로드하며, 상기 제1버전 프로그램을 저장하는 제1메모리에 상기 다운로드되는 다수의 갱신패키지를 저장하고, 갱신 요구시 상기 제1버전에 상기 다수의 갱신패키지 중 선택된 갱신패키지를 합성하여 제2버전을 생성하여 제2메모리에 저장하고, 상기 제2메모리의 제2버전을 통해 시스템을 운용하는 수신시스템으로 구성되며,

상기 수신시스템은,

상기 제1버전을 저장하는 영역 및 상기 다수의 갱신패키지를 저장하는 갱신패키지 저장영역을 구비하는 제1메모리와,

갱신된 제2버전을 저장하는 제2메모리와,

상기 다운로드되는 다수의 갱신패키지를 상기 제1메모리의 갱신패키지 저장 영역에 인스톨하는 인스톨러와,

갱신 요구시 상기 다수의 갱신패키지 중 선택된 갱신패키지를 상기 제1버전에 맵핑하여 제2버전을 생성하며, 상기 생성된 제2버전을 상기 제2메모리에 저장하는 트랜스레이터로 구성된 것을 특징으로 하는 상기 시스템의 버전 갱신 장치.

청구항 26

시스템의 버전을 갱신하는 장치에 있어서,

제1 및 제2버전을 입력하여 비교하는 비교기와, 상기 비교 결과에 따라 갱신테이터를 인스톨하기 위한 인스톨테이터를 생성하는 인스톨테이터발생기와, 상기 비교 결과에 따라 갱신테이터를 생성하며, 상기 인스톨테이터 및 갱신테이터를 결합하여 갱신패키지를 생성하는 패키지발생기로 구성되는 갱신패키지처리기와,

상기 갱신패키지의 생성을 하나 이상의 수신시스템들에 통지하는 갱신패키지서버와,

상기 제1버전을 저장하는 영역 및 다수 번의 시스템의 버전 업데이트에 따른 다수의 갱신패키지를 저장하는 갱신패키지 저장영역을 구비하는 제1메모리와, 갱신된 제2버전 프로그램을 저장하는 제2메모리와, 상기 다수의 갱신패키지를 상기 제1메모리의 갱신패키지 저장 영역에 인스톨하는 인스톨러와, 갱신 요구시 상기 다수의 갱신패키지 중 선택된 갱신패키지를 상기 제1버전에 맵핑하여 제2버전 프로그램을 생성하며, 상기 생성된 제2버전을 상기 제2메모리에 저장하는 트랜스레이터로 구성되는 수신시스템으로 구성된 것을 특징으로 하는 상기 시스템의 버전 갱신 장치.

명세서

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

- [0026] 본 발명은 정보 갱신 장치 및 방법에 관한 것으로, 특히 시스템의 버전정보를 갱신할 수 있는 장치 및 방법에 관한 것이다.
- [0027] 일반적으로 단말 시스템에 포함되는 소프트웨어(software) 또는 펌웨어(firmware) 등은 많은 검증 과정을 거쳐 상품화된다. 그러나 상기 단말 시스템에 채택된 모든 소프트웨어 또는 펌웨어는 프로그램의 버그 또는 프로그램의 버전 변경 등에 의해 프로그램을 갱신하여야 한다. 이때 상기 프로그램의 변경이 일반적인 유형의 사용 중에 발생되면, 단말 시스템을 공급하는 제조사나 서비스 캐리어(service carrier) 등은 이런 문제점을 해결하기 위한 소프트웨어 또는 펌웨어 등을 갱신시켜 줄 의무를 가진다. 이때 상기 제조사 및 서비스캐리어의 갱신서비스는 좀 더 나은 형태의 소프트웨어 또는 펌웨어의 교체도 가능하다.
- [0028] 이때 상기 갱신 서비스는 주로 사용자가 서비스센터를 방문하여 받는 것이 일반적이다. 여기서 상기 단말기가 무선 통신 기능을 구비하는 휴대단말기인 경우를 가정하면, 상기 업그레이드 서비스는 OTA(over the air) 형태로도 가능하다. 상기 OTA의 경우 사용자의 단말 업그레이드를 위하여 패치 소프트웨어(patch software) 전송받을 수 있는 다운로드 모듈과 현재 단말의 소프트웨어 버전을 갱신할 수 있는 소프트웨어 패키지(software

package)를 이용하여 업그레이드 시키는 업그레이드 처리모듈이 필요하다.

발명이 이루고자 하는 기술적 과제

- [0029] 따라서 본 발명의 목적은 정보를 갱신하는 시스템에서 최초 버전을 가진 시스템을 갱신 패키지를 이용하여 새로운 버전의 시스템으로 갱신할 수 있는 장치 및 방법을 제공함에 있다.
- [0030] 본 발명의 다른 목적은 최초 버전 및 적어도 하나의 갱신버전들을 제1메모리에 저장하며, 제2메모리에 상기 최초 버전의 정보를 갱신 버전의 정보로 갱신하는 정보를 저장하고, 시스템이 상기 제2메모리의 정보를 이용하여 시스템을 운용할 수 있는 장치 및 방법을 제공함에 있다.
- [0031] 본 발명의 또 다른 목적은 최초 버전 정보와 적어도 하나의 갱신 버전 정보들을 저장하고 있으며, 상기 갱신버전 정보는 인스톨 데이터 및 갱신데이터로 구성되는 시스템에서 시스템 기동 또는 사용자의 초기화 명령에 의해 상기 최초 버전의 정보를 상기 갱신버전의 정보로 갱신하여 시스템을 운용하는 장치 및 방법을 제공함에 있다.
- [0032] 본 발명의 또 다른 목적은 최초버전과 갱신버전의 정보를 비교하여 갱신데이터를 생성하고, 상기 갱신데이터를 상기 갱신버전의 히스토리 정보와 머지하여 상기 갱신패키지로 생성한 후, 정보를 갱신하는 시스템에 전송할 수 있는 장치 및 방법을 제공함에 있다.
- [0033] 본 발명의 또 다른 목적은 갱신 데이터 및 갱신버전의 히스토리 정보가 머지된 갱신패키지를 수신하는 시스템에서 상기 갱신패키지 수신시 상기 갱신패키지의 인스톨 데이터를 추출한 후 이를 상기 갱신패키지와 함께 제1메모리에 저장하고, 상기 갱신명령시 상기 제2메모리에 상기 최초버전에 상기 인스톨 데이터에 따라 상기 갱신버전의 데이터를 갱신하여 저장하며, 상기 제2메모리의 내용으로 시스템을 운용할 수 있는 장치 및 방법을 제공함에 있다.

발명의 구성 및 작용

- [0034] 이하 본 발명의 바람직한 실시예들의 상세한 설명이 첨부된 도면들을 참조하여 설명될 것이다. 도면들 중 동일한 구성들은 가능한 한 어느 곳에서든지 동일한 부호들을 나타내고 있음을 유의하여야 한다.
- [0035] 하기 설명에서 갱신버전 블록의 수, 매크로블록의 크기 등과 같은 특정 상세들이 본 발명의 보다 전반적인 이해를 제공하기 위해 나타나 있다. 이들 특정 상세들 없이 또한 이들의 변형에 의해서도 본 발명이 용이하게 실시될 수 있다는 것은 이 기술분야에서 통상의 지식을 가진 자에게 자명할 것이다.
- [0036] 본 발명의 실시예에서, "갱신"(upgrade)이라는 용어는 시스템의 정보를 갱신하는 동작을 의미한다. 또한 "정보"(information)라는 용어는 상기 갱신 대상으로써, 상기한 바와 같이 펌웨어 및 소프트웨어 등의 프로그램 및 불휘발성 데이터 들을 의미한다. 여기서 상기 정보에서 프로그램은 펌웨어(firmware) 및 소프트웨어(software) 등을 의미하며, 불휘발성 데이터는 수신시스템에서 사용되는 폰트(font), 시스템 이미지, 사용자 데이터 등이 될 수 있다. 이때 상기 사용자 데이터는 불휘발성 데이터가 될 수 있다. 그리고 상기 불휘발성 데이터의 갱신 예를 살펴보면, 상기 수신시스템이 휴대단말기인 경우 사용자는 컴퓨터 등을 통해 휴대단말기에서 사용하기 위한 데이터(예를들면 가입자 정보, 멀티미디어 데이터 등)를 작성 또는 구매한 후 이를 서버로 업로드시킬 수 있으며, 이런 경우 상기 서버는 이전의 불휘발성 데이터와 비교하여 갱신패키지를 생성하며, 상기 생성된 갱신패키지를 상기 휴대단말기에 다운로드할 수 있다. 이하의 설명에서는 상기 정보가 프로그램인 경우를 가정하여 설명하기로 한다. 여기서 상기 프로그램은 펌웨어 및 소프트웨어가 될 수 있으며, 본 발명의 실시예에서의 정보 갱신은 상기 프로그램의 갱신패키지를 생성하여 처리하는 절차를 구체적으로 살펴보기로 한다. 상기 "갱신 패키지"는 갱신 데이터 및 인스톨 데이터들을 포함하는 정보를 의미한다. 상기 "갱신 데이터"(upgrade data)는 상기 프로그램 갱신시 이전 버전 정보를 대체하여 새로운 버전의 정보를 생성하기 위한 데이터를 의미한다. "인스톨 데이터"는 상기 정보 갱신시 이전 버전을 이용하여 갱신버전을 생성할 때의 인스톨 데이터를 의미하며, 상기 인스톨 데이터는 히스토리 데이터 및 맵(map) 데이터들로 구성될 수 있다. 그리고 상기 맵 데이터는 발명의 실시예에서는 새로운 버전의 데이터를 생성하기 위하여 복사(copy), 이동(shift) 및 변경(modify) 등의 명령어 및 이들 명령어들을 실행하기 위한 블록 위치 데이터들을 포함할 수 있다. "제1버전"은 시스템의 최초 버전 정보 또는 시스템의 기준이 되는 버전의 정보가 될 수 있다. "제2버전"은 갱신 버전으로써, 상기 제1버전 정보를 업그레이드한 정보가 될 수 있다. 이때 수신측은 상기 제1버전 정보를 다운로드하여 저장하여야 하며, 상기 제2버

전은 하나 또는 2 이상의 제2버전 정보들을 갱신팬키지로 저장할 수 있다. 따라서 상기 "갱신팬키지"는 상기 제1버전을 갱실키기 위한 정보로, 상기 제1버전 정보로부터 제2버전 정보를 생성할 때 필요한 인스톨 데이터 및 갱실키데이터를 포함할 수 있으며, 본 발명의 실시예에서 상기 제2버전의 갱신팬키지는 상기 제1버전 정보를 복사(copy), 쉬프트(shift) 및 변경(modify)의 명령어 및 블록 위치데이터, 그리고 이에 관련된 갱실키 데이터들을 포함할 수 있다. 상기 "프로그램"은 시스템의 펌웨어(firmware) 및 소프트웨어(software)를 포함하는 용어로 사용될 것이다.

[0037] "제1메모리"는 상기 제1버전 프로그램 및 제2버전의 갱신팬키지들을 저장하는 메모리를 의미한다. "제2메모리"는 상기 제1버전 정보에 상기 제2버전 갱신팬키지를 이용하여 생성된 갱실키 정보를 저장하는 메모리를 의미한다. 여기서 상기 제1메모리 및 제2메모리는 하나의 메모리 상에 제1영역 및 제2영역으로 구분되는 메모리가 될 수 있으며, 또한 서로 다른 2개의 메모리들이 될 수 있다. 본 발명의 실시예에서는 상기 제1 및 제2메모리가 물리적으로 다른 2개의 메모리들이라고 가정하며, 여기서 각각 상기 "제1메모리"는 비휘발성메모리(non-volatile memory)로써 플래시 메모리이며, "제2메모리"는 휘발성 메모리(volatile memory)로써 SRAM이라고 가정한다. 본 발명의 실시예에서 상기 제1메모리는 상기 제1버전 정보, 그리고 적어도 하나의 상기 제2버전 갱신팬키지들을 저장하며, 상기 제2버전의 갱신팬키지는 각 버전 정보를 식별하기 위한 히스토리 데이터(그리고 맵 데이터들을 포함할 수 있음) 및 갱실키데이터들을 각각 저장한다. 그리고 시스템 기동시 또는 사용자의 명령에 의해 상기 시스템은 제1버전 정보에 상기 제2버전 갱신팬키지를 이용하여 생성된 갱실키 정보를 제2메모리에 저장한 후, 갱실키된 제2버전 정보로 시스템을 운용한다. 여기서 상기 갱신팬키지는 인스톨 데이터 및 갱실키데이터로 구성될 수 있다. 그리고 상기 제1버전 정보는 시스템의 최초 프로그램, 또는 기준 프로그램이 될 수 있으며, 제2버전의 갱실키 프로그램은 상기 제1버전 프로그램의 갱실키 프로그램이 될 수 있으며, 상기 제2메모리에 저장되는 상기 제2버전의 프로그램은 상기 제1버전 정보에 상기 제2버전의 갱신팬키지를 결합하여 생성된 프로그램이 될 수 있다.

[0038] 본 발명의 실시예에 따른 시스템 갱실키 및 유지보수 방법은 갱실키 패키지를 생성하여 전송하는 송신 시스템 및 상기 갱신팬키지를 수신하여 시스템 정보를 갱실키하는 수신시스템 등으로 구성된다. 도 1은 본 발명의 실시예에 따라 갱신팬키지를 생성하여 전송하고, 상기 전송되는 갱신팬키지를 다운로드하여 시스템 정보를 갱실키하는 시스템의 구성을 도시하는 도면이다. 이하의 설명에서는 상기한 바와 같이 상기 정보가 프로그램인 경우를 가정하여 설명하기로 한다.

[0039] 상기 도 1을 참조하면, 갱신팬키지처리기10은 제2버전 프로그램정보 발생시 상기 제1버전 프로그램 정보와 비교하여 그 차에 따른 갱신팬키지를 생성한 후, 이를 갱신팬키지서버20에 전송한다. 이때 상기 갱신팬키지처리기10은 상기 갱신팬키지서버20과의 연결 방법에 따라 무선(air protocols: CDMA, UMTS 등), 근거리 무선 통신(bluetooth, Zigbee 등), 근거리 유선통신(USB, UART), 유무선 인터넷(Wibro, Wifi, Wimax 등) 방식으로 전달될 수 있다. 그러면 상기 갱신팬키지서버20은 상기 갱신팬키지가 발생되었음을 하나 이상의 수신시스템30들에 통지(notification)하며, 응답하는 수신시스템30들에게 상기 갱신팬키지를 다운로드한다. 이때 상기 수신시스템의 갱신팬키지 다운로드 방법은 상기한 바와 같이 상기 갱신팬키지서버20 및 수신시스템30의 연결 방법에 따라 무선(air protocols: CDMA, UMTS 등), 근거리 무선 통신(bluetooth, Zigbee 등), 근거리 유선통신(USB, UART), 유무선 인터넷(Wibro, Wifi, Wimax 등) 방식 등을 사용할 수 있다. 본 발명의 실시예에서는 상기 갱실키 패키지 처리기10과 갱신팬키지서버20이 독립적으로 구성된 예로 설명되고 있지만, 상기 갱신팬키지처리기10과 갱신팬키지서버20은 일체형으로 구성할 수도 있다.

[0040] 그러면 상기 수신시스템30은 상기 다운로드되는 갱신팬키지를 제2버전 프로그램을 생성하기 위한 정보로 메모리에 저장한다. 이때 상기 수신시스템30의 메모리는 제1메모리 및 제2메모리로 구성될 수 있다. 여기서 상기 제1메모리 및 제2메모리는 하나의 메모리 상에 영역을 할당하여 구분된 메모리들로 구성될 수 있으며, 또한 각각 서로 다른 메모리들이 될 수 있다. 여기서 상기 제1메모리는 다운로드되는 제1버전 프로그램 및 갱신팬키지를 저장하는 메모리이고, 제2메모리는 상기 제1버전 프로그램을 상기 갱신팬키지에 의해 갱실키하여 생성된 제2버전 프로그램을 저장하는 메모리이다. 즉, 상기 수신시스템30은 상기 다운로드되는 갱신팬키지를 상기 제1메모리에 제2버전 프로그램의 갱실키정보로 저장한다. 이후 갱실키요구시 상기 수신시스템은 상기 제1버전 프로그램정보에 상기 다운로드된 갱신팬키지를 결합(merge)하여 제2버전 프로그램을 생성하며, 상기 생성된 제2버전 프로그램을 제2메모리에 저장한다. 이후 상기 수신시스템30은 상기 제2메모리에 저장된 갱실키된 제2버전 프로그램을 통해 시스템을 운용한다.

[0041] 먼저 상기 갱신팬키지처리기10의 동작을 살펴본다. 도 2는 본 발명의 실시예에 따른 갱신팬키지처리기10과 주변 구성의 관계를 설명하기 위한 도면이다.

- [0042] 상기 도 2를 참조하면, 제1버전 프로그램50과 제2버전 프로그램 55는 상기 갱신패키지처리기10에 입력된다. 여기서 상기 제1버전 프로그램50은 최초 버전의 프로그램이 될 수 있으며, 상기 제2버전 프로그램 55는 상기 갱신패키지처리기10에 의해 상기 제1프로그램 50으로부터 생성되는 프로그램이 될 수 있다. 그러면 상기 갱신패키지처리기10은 먼저 상기 제1버전에서 현재의 제2버전으로 이동하기 위한 제2버전의 갱신 패키지(upgrade package)를 생성한다. 그리고 상기 제1버전 및 제2버전 프로그램들을 비교하며, 비교 결과에 따른 갱신 패키지를 생성한 후, 상기 갱신패키지 서버20에 전달한다. 상기 갱신패키지는 갱신 데이터 및 인스톨 데이터로 구성될 수 있다. 이런 경우 상기 갱신패키지는 제1 및 제2버전 프로그램의 차에 따른 데이터가 될 수 있다. 그리고 상기 인스톨데이터는 히스토리 데이터 및 맵 데이터로 구성될 수 있으며, 상기 히스토리 데이터는 제1버전 프로그램과 결합하기 위한 데이터들이 될 수 있고, 상기 맵 데이터는 상기 제1 및 제2버전 프로그램의 비교 결과에 따라 상기 제1버전 프로그램에 제2버전프로그램을 맵핑하기 위한 복사, 변경 및 쉬프트의 명령어 및 상기 명령어들의 인덱스 정보들을 포함할 수 있다. 이때 상기 갱신패키지를 상기 갱신패키지서버20에 전송하는 방법은 그 연결 방법에 따라 무선, 근거리통신, 무선 인터넷, 유선 통신 방식들 중에 한 방법을 사용할 수 있다.
- [0043] 도 3a 및 도 3b는 상기 갱신패키지처리기10에서 갱신패키지를 생성하는 방법을 설명하기 위한 도면이다.
- [0044] 여기서 도 3a는 제1버전 정보의 예를 도시하고 있으며, 도 3b는 제2버전정보의 예를 도시하고 있다. 여기에서 상기 제2버전 정보는 상기 제1버전 정보를 그대로 복사(copy)하거나 수정(modify)할 수 있으며, 경우에 따라 수정된 데이터 양에 따라 그 정보의 위치가 쉬프트(shift) 될 수 있다.
- [0045] 그리고 도 4는 상기 도 2의 갱신패키지처리기10의 구성예를 도시하는 도면으로써, 제1 및 제2버전 프로그램을 압축하고, 상기 압축된 두 버전 프로그램들을 비교 분석하여 맵데이터 및 갱신패키지데이터를 생성하고, 상기 맵데이터를 포함하는 인스톨데이터 및 상기 갱신패키지데이터를 합성하여 갱신패키지를 생성하는 갱신패키지 생성기의 구성을 도시하는 도면이다.
- [0046] 또한 도 5는 상기 도 2의 갱신패키지처리기10의 또 다른 구성예를 도시하는 도면으로써, 제1 및 제2버전을 압축하고, 상기 압축된 두 버전들을 비교 분석하여 갱신패키지데이터를 생성하고, 상기 맵데이터를 포함하지 않는 인스톨데이터 및 상기 갱신패키지데이터를 합성하여 갱신패키지를 생성하는 갱신패키지 생성기의 구성을 도시하는 도면이다.
- [0047] 그리고 도 6a는 상기 도 4와 같은 갱신패키지처리기10에서 발생하는 갱신패키지의 구성을 도시하고 있으며, 도 6b 는 상기 도 5와 같은 갱신패키지처리기10에서 발생하는 갱신패키지의 구성을 도시하고 있다.
- [0048] 도 7은 상기 도 2의 갱신패키지처리기10의 구성예를 도시하는 또 다른 도면으로써, 압축하지 않은 제1 및 제2버전들을 비교 분석하여 갱신패키지데이터를 생성하고, 상기 맵데이터를 포함하는 인스톨데이터 및 상기 갱신패키지데이터를 합성하여 갱신패키지를 생성하는 갱신패키지처리기10의 구성을 도시하는 도면이다. 그러나 상기 도 5와 유사한 방법으로, 상기 갱신패키지처리기10은 상기 맵데이터를 포함하지 않는 인스톨데이터를 생성한 후 이를 갱신패키지에 합성하여 갱신패키지를 생성할 수도 있다.
- [0049] 먼저 상기 도 3a - 도 6b를 참조하면, 상기 제1버전 프로그램 50과 상기 제2버전 프로그램 55는 상기 갱신패키지처리기10에 입력되는 데, 도 4 및 도 5의 경우와 같이 제1압축기160에 입력되어 압축된 형태로 입력될 수 있고, 도 7의 경우와 같이 원래의 정보 그대로 입력될 수도 있다. 상기 제1압축기160은 상기 제1버전 프로그램을 설정된 크기(이하 매크로 블록(macro block)이라 칭한다)로 분할하여 압축하며, 또한 상기 제2버전 프로그램도 상기 매크로 블록 단위로 분할하여 압축한다. 상기 제1압축기160이 설정된 블록 크기로 압축을 하는 이유는 뒷단의 비교기110이 두 버전들을 블록 단위로 비교할 수 있도록 하기 위함이다. 그러면 상기 비교기110은 압축된 두 버전의 정보들을 비교하여 두 블록 데이터의 동일 여부를 표시한다. 이때 상기 두 버전들의 대응되는 블록 데이터들이 동일하지 않으면, 상기 비교기110은 상기 제1버전 및 제2버전의 블록 데이터가 설정된 구간 내의 제1버전 블록 데이터들과 비교한다. 상기과 같이 두 버전의 블록 데이터들을 비교한 후, 상기 비교기110은 비교 결과값 및 대응되는 상기 제1버전의 블록 인덱스 값을 상기 인스톨데이터 발생기180에 전달한다.
- [0050] 상기 인스톨데이터 발생기180은 상기 도 4에 도시된 바와 같이 히스토리데이터발생기120 및 맵데이터발생기150으로 구성될 수 있으며, 또한 상기 도 5에 도시된 바와 같이 히스토리데이터발생기120만으로 구성될 수도 있다.
- [0051] 먼저 상기 도 4를 참조하면, 상기 히스토리데이터발생기120은 제2버전의 버전 번호(version number)를 포함한다. 즉, 버전 번호가 5인 경우, 상기 제2버전은 버전번호 5가 되며, 상기 제1버전 및 버전번호 5인 제2버전을 결합하여 갱신한다는 의미를 갖는다. 상기 맵데이터발생기150은 상기 비교기110로부터 출력되는 블록 데이터를 분석하여 갱신패키지를 인스톨하기 위한 명령어(command)를 결정하는 맵 데이터를 발생한다. 본 발명의

실시예에서 상기 맵데이터에 포함된 명령어는 복사(copy: C), 변경(modify: M) 및 이동(shift: S)라고 가정한다. 여기서 상기 복사 C는 제1 및 제2버전의 블록 인덱스가 동일한 경우에 매핑되는 데이터를 의미하며, 변경 M은 제1 및 제2버전의 해당 블록 데이터의 크기가 다르거나(크거나 작은 경우) 또는 해당 블록의 데이터 값이 다른 경우에 매핑되는 데이터를 의미하고, 이동 S는 제1 및 제2 버전의 블록 데이터는 동일하지만 해당 블록의 위치 값이 다른 경우에 매핑되는 데이터를 의미한다. 그리고 상기와 같은 맵 데이터들은 각 블록의 수에 대응되도록 매핑되며, 이동 S인 경우에는 쉬프트 값을 포함하며, 변경 M인 경우에는 해당 블록의 제2버전 데이터를 갱신데이터로서 가진다. 따라서 상기 맵데이터는 제1 및 제2버전의 블록 데이터가 동일하면 복사 C, 블록 데이터가 다르면(블록 데이터의 추가 또는 내용이 다른 경우) 변경 M, 블록 데이터가 동일한데 위치가 다르면 이동 S의 맵 데이터를 해당 블록에 매칭시켜 설정한다. 주의할 점은 상기 제1버전 프로그램 및 상기 명령어들을 이용하여 상기 제2버전 프로그램 생성시, 실제 블록 데이터가 필요한 것은 상기 변경 M의 경우이며, 나머지 복사 C 및 이동 S의 경우에는 복사 또는 이동되는 제1버전의 블록 및 위치(블록 인덱스)정보로 충분하다는 점이다. 따라서 상기 갱신데이터는 상기한 바와 같이 변경 M에 해당하는 블록들의 데이터가 될 수 있으며, 상기 복사 C 및 이동 S의 경우에는 블록 또는 블록 및 위치에 대한 정보들이 인스톨 데이터로 구성된다.

[0052] 상기한 바와 같이 맵데이터 발생기150은 제1 및 제2버전의 블록 인덱스 값 및 비교 결과 값을 분석하여 갱신패키지를 생성하기 위한 맵 데이터를 발생한다. 이때 상기 맵데이터발생기150은 상기한 바와 비교기110에서 출력되는 제1버전 및 제2버전의 블록 인덱스들 및 비교결과 값에 따라 맵데이터를 생성한다. 이때 제1 및 제2버전의 동일한 블록 인덱스에서 동일한 비교값이 발생되면, 상기 맵데이터 발생기150은 해당 블록의 명령어를 복사 C로 결정하고, 이런 블록번호들을 누적한다. 상기와 같은 동작은 제2버전의 변경 블록이 발생되거나 또는 상기 제1버전의 삭제 블록이 발생될 때 까지 계속될 수 있다.

[0053] 또한 상기 비교기110이 제1 및 제2버전의 동일한 블록 인덱스에 다른 비교값을 발생하면, 상기 맵데이터 발생기150은 제2버전의 해당 블록의 명령어를 변경M 또는 쉬프트 S로 결정하게 한다. 이때 상기 변경 M은 두 가지 형태로 발생될 수 있다. 그 하나는 삽입이 될 수 있으며, 다른 하나는 대체가 될 수 있다. 상기 맵데이터발생기150은 상기 비교기110에서 출력되는 비교결과값 및 블록 인덱스를 연속으로 분석한다. 이때 블록 인덱스들을 분석하여 상기 제1버전에 제2버전의 데이터를 삽입하는 변경으로 판단되면, 상기 맵데이터발생기150은 명령어를 변경M으로 하는 맵 데이터를 생성한다. 또한 상기 변경 블록 데이터가 동일한 크기로 대체되는 경우, 상기 맵데이터 발생기150은 두 블록의 차 데이터를 앤트로피 코딩하여 맵데이터에 포함시킬 수 있다. 이런 경우, 변경 블록이 동일한 크기의 블록으로 대체되는 경우, 이어지는 제2버전의 뒤 블록들은 쉬프트 동작이 발생되지 않을 수 있다. 즉, 제1버전과 동일한 블록 인덱스를 가지는 제2버전의 블록을 동일한 크기로 변경하는 경우, 이어지는 제2버전의 다음 블록들은 쉬프트되지 않게 된다. 또한 상기 제2버전이 상기 제1버전의 일부 블록들을 삭제하는 프로그램인 경우, 상기 맵데이터발생기150은 이어지는 제2버전의 다음 블록들을 쉬프트시키는 맵데이터를 생성한다. 이런 경우, 상기 쉬프트 동작은 상기 제1버전의 삭제된 블록 인덱스로 쉬프트되어야 한다.

[0054] 상기와 같이 변경 블록이 삽입되는 경우, 이후의 블록 데이터들을 쉬프트되어야 한다. 따라서 상기 맵데이터발생기150은 상기 변경 블록을 삽입하는 맵데이터를 생성한 후, 이후 입력되는 비교기110의 출력을 분석하여 제1버전의 블록 인덱스와 제2버전의 블록 인덱스를 분석한다. 그리고 제2버전의 블록 인덱스, 쉬프트할 블록들의 수, 그리고 제1버전과 제2버전의 쉬프트 블록 수를 포함하는 맵데이터를 생성한다.

[0055] 상기와 같은 동작 수행 중 상기 갱신패키지 발생기100은 상기 맵 데이터를 발생하지 않을 수도 있다. 이는 수신측에서 상기 제1버전의 정보를 이용하여 갱신데이터를 인스톨하기 위한 맵 데이터를 생성할 수 있기 때문이다. 따라서 상기 인스톨데이터에서 상기 맵데이터 발생기150은 생략될 수 있다.

[0056] 상기 패키지발생기130은 상기 제1압축기160에서 출력되는 제2버전의 압축 정보 및 상기 맵데이터 발생기150에서 상기 맵데이터를 분석하여 갱신패키지를 발생한다. 이때 상기 패키지발생기130은 상기 맵데이터발생기150에서 발생되는 맵데이터를 분석하여 갱신 데이터의 발생 여부를 결정한다. 즉, 상기 패키지발생기130은 상기 맵 데이터의 명령어를 분석하여 복사C 및 쉬프트S의 명령어를 가지는 블록인 경우에는 갱신데이터를 발생하지 않는다. 그러나 상기 맵데이터의 명령어가 변경M인 경우, 상기 패키지발생기130은 상기 비교기110에서 상기 압축된 제2버전의 블록 데이터를 입력하여 갱신데이터로 설정한다. 즉, 상기 패키지발생기130은 상기 맵데이터를 분석하여 복사 C 및 쉬프트 S 명령어를 가지는 맵데이터이면 갱신데이터를 발생하지 않으며, 변경 M 명령어이면 갱신데이터를 발생한다.

[0057] 이후 상기 패키지발생기130은 상기 인스톨데이터발생기180에서 발생되는 인스톨 데이터에 상기 갱신데이터를 머지(merge)하여 갱신 패키지로 생성한다. 이때 상기 인스톨 데이터는 상기 히스토리데이터 또는 상기 히스토리데

이터 및 맵 데이터들이 될 수 있다. 즉, 상기 도 4와 같이 인스톨데이터발생기180의 구성이 히스토리데이터발생기120 및 맵데이터발생기150으로 구성된 경우, 상기 인스톨 데이터는 히스토리 및 맵 데이터들이 되며, 상기 도 5와 같이 인스톨데이터발생기180의 구성이 히스토리데이터발생기120만으로 구성된 경우, 상기 인스톨 데이터는 히스토리 데이터가 될 수 있다.

[0058] 두 번째로 상기 도 5를 참조하면, 상기 인스톨데이터발생기180은 맵데이터 발생기150을 구비하지 않는다. 이런 경우 상기 패키지발생기130은 상기 제2버전의 각 블록들이 상기 제1버전의 각 블록들에 매핑되는 블록 인덱스 값 및 블록 데이터들에 정보를 포함하는 갱신데이터를 생성한다. 이때 상기 갱신데이터는 상기 맵데이터 발생기 150과 유사한 명령어들을 포함할 수 있다. 즉, 상기 갱신데이터는 상기 제2버전 프로그램의 블록 데이터를 상기 제1버전 프로그램에서 복사하는 명령어는 시작 블록의 번호 및 복사할 블록의 수를 포함하는 정보를 가지도록 구성하고, 상기 제2버전 프로그램의 블록데이터를 상기 제1버전 프로그램에 추가 또는 변경하는 명령어는 시작 블록의 번호, 변경 블록의 수 및 각 변경 블록의 데이터들이 순차적으로 연결되는 정보를 가지도록 구성하며, 상기 제2버전 프로그램의 블록 데이터를 상기 제1버전 프로그램 블록에 쉬프트하는 명령어는 시작블록의 번호, 쉬프트블록의 수 및 첫 번째 쉬프트블록에 대응되는 제1버전 프로그램의 블록 인덱스의 정보를 가지도록 구성한다.

[0059] 이후 상기 패키지발생기130은 상기 갱신데이터와 히스토리 데이터를 결합하여 갱신패키지로 생성하며, 상기 갱신패키지는 갱신패키지서버20에 전달된다. 이때 상기 패키지발생기130에서 생성되는 갱신패키지는 제2압축기140을 통해 압축되어 상기 갱신패키지서버20에 전달될 수도 있다. 상기과 같이 맵 데이터를 생성하지 않고 갱신데이터를 이용하여 제2버전 프로그램의 갱신패키지를 생성하는 경우, 갱신 패키지 생성 속도를 향상시킬 수 있다.

[0060] 따라서 상기 갱신패키지는 히스토리데이터, 맵 데이터들 및 갱신데이터들이 될 수 있으며, 또한 히스토리데이터 및 갱신데이터들이 될 수 있다. 따라서 상기 패키지발생기130에서 생성되는 갱신패키지는 도 4에서와 같이 히스토리데이터 및 맵데이터로 구성되는 인스톨데이터를 발생하는 경우 도 6a와 같이 생성되며, 도 5에서와 같이 히스토리데이터로만 구성되는 인스톨데이터를 발생하는 경우 도 6b와 같이 생성된다.

[0061] 이후 상기 패키지발생기130에서 출력되는 갱신패키지는 제2압축기140에서 압축되어 갱신패키지서버20에 전달된다. 여기서 상기 제2압축기140은 생략될 수 있다. 이는 상기 갱신패키지가 이미 상기 제1압축기160을 통해 압축 부호화된 상태이기 때문이다. 그러나 전송효율을 높이기 위해 상기 도 4에 도시된 바와 같이 상기 제2압축기140을 통해 다시 압축할 수도 있다. 또한 상기 제1압축기160에서 상기 제1 및 제2버전 정보를 압축하는 경우, 제1 압축해제기165는 압축된 제1 및 제2버전 정보를 압축 해제한다. 이는 상기 제1압축기160에서 압축된 제1 및 제2버전 정보들이 정상적으로 압축되었는가 확인하기 위한 것으로, 압축 시 오류가 발생된 경우 상기 제1압축기160을 제어하여 재압축하도록 제어할 수 있다.

[0062] 상기 도 4 및 도 5는 갱신패키지처리기10이 제1 및 제2버전을 압축한 후 두 버전을 비교분석하여 갱신패키지를 생성하는 구조를 도시하고 있다. 이런 경우 상기 비교기110은 압축된 두 버전 정보들을 비교하게 된다. 그러나 상기 비교기110은 상기 제 1 및 제2버전 프로그램들을 압축하지 않은 데이터로 비교할 수 있다. 도 7은 상기과 같이 제1 및 제2버전 정보들을 압축하지 않은 데이터로 비교하여 갱신 패키지를 생성하는 구조를 도시하고 있다. 상기 도 7에서 상기 제1 및 제2버전프로그램을 압축하는 구성을 제외한 나머지 구성은 상기 도 4와 동일하며, 그 동작도 동일하다. 또한 상기 도 7은 상기 인스톨데이터발생기180이 히스토리 데이터 및 맵 데이터를 생성하는 예를 도시하고 있지만, 도 5와 같이 상기 인스톨데이터발생기180이 히스토리데이터발생기120만을 구비할 수도 있다.

[0063] 상기한 바와 같이, 갱신패키지처리기10은 상기 제1버전을 기반으로 상기 제2버전을 비교하며, 상기 비교결과에 따라 인스톨데이터를 포함하는 갱신 패키지를 생성한다. 이때 상기 비교 과정에서 상기 제2버전이 상기 제1버전에 새로운 데이터를 추가하거나 삭제하는 변경이 발생되면, 상기 제2버전의 블록들은 상기 제1버전의 해당 영역에 이어지는 다음 블록들에 비해 쉬프트되는 결과를 갖는다. 이런 경우, 상기 제2버전의 블록들은 상기 제1버전에서 상기 변경된 블록을 따라 앞으로 쉬프트되거나 또는 뒤로 쉬프트되는 결과를 초래한다. 또한 상기 제2버전의 변경된 블록은 해당 블록의 데이터를 엔트로피 코딩하게 된다. 그리고 상기과 같이 변경블록 이후의 제2버전의 블록 데이터들은 상기 제1버전의 블록 데이터들에서 상기 삽입 또는 삭제된 블록 수 만큼 쉬프트된다. 따라서 상기 갱신패키지처리기10은 상기 제1버전 및 제2버전을 비교한 후, 명령어 복사S, 변경M 및 쉬프트S로 이루어지는 맵 데이터(상기 맵데이터를 사용하지 않는 경우에는 갱신데이터에 이를 포함시켜 생성할 수 있음)를 생성하며, 맵데이터, 히스토리 데이터 및 갱신데이터(상기 맵데이터를 생성하지 않는 경우에는 히스토리 데이터 및 갱신데이터)로 이루어지는 상기 제2버전의 갱신패키지를 생성하여 상기 갱신패키지서버20에 전송한다. 이때

상기 갱신패키지처리기10에서 갱신패키지를 전송하는 방법은 상기 갱신패키지서버20과의 연결 방법에 따라 무선 통신, 근거리통신, 인터넷 통신 또는 유선 통신 방법을 사용할 수 있다.

[0064] 그러면 상기 갱신패키지서버20은 상기 갱신패키지를 다운로드할 수 있는 수신시스템30들에 새로운 갱신패키지의 발생을 통지한다. 그리고 상기 통지 결과에 따라 응답하는 수신시스템30들에 상기 갱신패키지를 다운로드한다. 따라서 상기 갱신패키지서버20은 갱신패키지의 발생을 각 수신시스템들에 통지하는 통지서버(notification server)를 더 구비할 수 있다.

[0065] 상기 수신시스템30은 다운로드할 갱신패키지의 발생을 통지받으면, 이에 대한 응답을 하고 갱신패키지서버20에서 다운로드되는 갱신패키지를 다운로드한다. 도 8은 본 발명의 실시예에 따른 수신시스템의 구성을 도시하는 도면이다.

[0066] 상기 도 8을 참조하면, 다운로더(down-loader)220은 상기 갱신패키지서버20에서 다운로드하는 갱신패키지를 다운로드하며, 인스톨러(installer)230은 상기 다운로드된 갱신패키지에서 인스톨데이터 및 갱신테이터를 분석하여 상기 제1메모리250에 저장한다. 이때 상기 인스톨데이터는 히스토리데이터 및 맵데이터로 구성될 수 있으며, 또한 히스토리데이터로만 이루어질 수도 있다. 여기서 상기 인스톨 데이터가 맵 데이터를 포함하지 않는 경우, 상기 제2버전의 제1버전 매핑 정보는 갱신테이터에 포함되는 구조를 가질 수 있다. 이때 상기 인스톨러230은 상기 맵데이터가 포함되지 않은 인스톨데이터를 수신하는 경우, 상기 제1버전과 상기 갱신테이터를 비교 분석하여 상기 맵데이터를 생성할 수 있으며, 또한 맵데이터를 생성하지 않을 수도 있다. 이때 상기 인스톨러230에서 상기 맵데이터를 생성하지 않는 경우, 트랜스레이터240에서 상기 갱신테이터의 매핑 정보를 이용하여 상기 갱신패키지를 직접 상기 제1버전에 결합할 수도 있다. 본 발명의 실시예에서는 상기 맵데이터를 생성하는 것으로 가정하여 설명한다. 그리고 상기 인스톨러230은 상기 제1메모리250의 제2버전의 갱신패키지를 저장하는 영역에서 히스토리 데이터, 맵 데이터 및 갱신테이터를 매칭시켜 저장한다. 이때 상기 제1메모리250에는 최초 버전인 제1버전 및 상기 제1버전을 제2버전으로 생성하기 위한 적어도 하나의 갱신패키지가 저장될 수 있다. 여기서 상기 갱신패키지는 적어도 하나가 될 수 있으며, 최대 N개의 갱신패키지를 저장할 수 있다. 본 발명의 실시예에서는 상기 N=6개라고 가정한다.

[0067] 또한 상기와 같이 새로운 버전의 갱신패키지를 다운로드하게 되면, 상기 시스템은 사용자에게 프로그램의 버전 갱신 시점임을 표시한다. 이때 상기 시스템 갱신 요구가 발생되거나 또는 시스템 초기화 동작을 수행한 후, 상기 수신시스템30의 트랜스레이터(translator)240은 상기 제1메모리250에서 제1버전 및 다운로드된 제2버전의 갱신패키지를 로딩한 후, 상기 제1버전에 상기 인스톨된 갱신패키지를 결합하여 제2버전을 생성한 후, 이를 상기 제2메모리260에 저장한다. 이때 상기 트랜스레이터240은 상기 갱신패키지의 인스톨데이터를 분석하여 버전번호 및 결합할 버전을 결정하며, 맵 데이터를 분석하여 상기 갱신패키지의 갱신테이터를 상기 결합할 버전(여기서는 상기 제1버전이 될 수 있음)의 해당 블록 내용을 갱신한다. 상기와 같은 동작을 수행하면서 제2버전의 프로그램으로 갱신하면, 상기 트랜스레이터240은 이를 상기 제2메모리260에 저장하며, 이후 수신시스템30은 상기 제2메모리260에 저장된 상기 제2버전 프로그램을 이용하여 시스템의 동작을 제어한다.

[0068] 상기한 바와 같이 상기 제1메모리250에는 제1버전 프로그램 및 제2버전 프로그램으로 생성하기 위한 적어도 하나의 갱신패키지가 저장된다. 이때 상기 갱신패키지는 인스톨데이터(히스토리 및 맵 데이터) 및 갱신테이터, 또는 히스토리 데이터 및 갱신테이터로 구성될 수 있다. 그리고 상기 인스톨데이터는 상기 제1버전에 상기 갱신테이터를 갱신할 때 사용하는 데이터로써 갱신패키지의 히스토리 데이터와 갱신테이터의 매핑 관련 정보인 맵데이터들로 구성될 수 있다. 여기서 상기 맵데이터는 두 버전 들 간의 관계를 나타내는 것으로, 상기한 바와 같이 3가지 형태(copy, modify, shift)들로 표시할 수 있다. 그리고 상기 맵 데이터는 특정 데이터(여기서는 제1버전)의 비교를 통해서 CMS(copy, modify, shift)를 찾는 생성방법으로 빠른 주소 계산을 목적으로 사용되며, 상기 제1메모리250에 포함되어 있는 제1버전을 토대로 상기 제2메모리260에서 상기 제2버전의 프로그램을 신속하게 생성하기 위한 데이터로 사용된다.

[0069] 이때 상기 다운로드되는 갱신패키지는 상기 갱신패키지처리기10에서 생성되는데, 이때 상기 인스톨데이터의 맵데이터는 생성될 수 있으며, 또한 생성되지 않을 수도 있다. 따라서 상기 수신시스템에 다운로드되는 갱신패키지에 맵 데이터가 포함될 수 있으며, 또한 포함되지 않을 수도 있다. 따라서 상기 맵 데이터가 포함되지 않은 갱신패키지를 다운로드한 경우, 상기 인스톨러230은 역으로 상기 제1메모리250에 저장된 제1버전과 상기 다운로드된 갱신패키지를 비교 분석하여 상기 갱신테이터를 상기 제1버전에 매핑하기 위한 맵데이터를 생성하여야 한다.

[0070] 또한 상기 트랜스레이터240은 상기 제1버전 프로그램과 가장 최근에 다운로드된 제2버전의 갱신패키지를 조합하

여 갱신하는 것이 일반적이지만, 사용자가 선택하는 특정 버전으로 갱신할 수도 있다. 즉, 적어도 하나의 버전을 저장하는데, 본 발명의 실시예에서는 상기 제2버전으로 생성하는 갱신패키지는 N개(본 발명의 실시예에서는 최대 6개로 가정하고 있음)까지 저장할 수 있다. 이런 경우 특정의 갱신패키지가 상기 제2메모리260에서 재조합이 실패되는 경우, 사용자에게 현재 시스템이 보유하고 있는 재조합 가능한 갱신패키지 리스트를 표시하고, 사용자의 선택에 따른 갱신패키지를 선택하여 제2버전 프로그램을 생성할 수도 있다.

[0071] 상기 제1메모리250은 다수개(여기서는 최대 6개로 가정하고 있음)의 갱신패키지를 저장할 수 있는 영역들을 구비한다. 따라서 상기 갱신패키지는 필요한 경우 다운로드되어 상기 제1메모리250의 저장영역에 저장되며, 다음의 갱신패키지가 다운로드되더라도 이전에 다운로드된 갱신패키지는 지우지 않는다. 따라서 버전의 변경에 따른 히스토리들이 누적되어 저장되며, 상기 제1버전의 저장상태를 그대로 유지된다. 따라서 제1버전 프로그램 및 제2버전의 갱신패키지들이 그대로 유지되므로, 갱신의 신뢰성(fault-tolerance)이 높다. 또한 최신의 제2버전으로 생성하기 위한 갱신패키지가 로딩되지 않는 경우, 사용자의 선택에 의해 특정 버전의 갱신패키지를 이용할 수 있으며, 모든 버전의 갱신 패키지 로딩이 실패되는 최악의 경우에도 최초 버전의 제1버전 프로그램을 로딩하여 사용할 수 있다.

[0072] 도 9는 본 발명의 실시예에 따른 수신시스템의 제1메모리250의 구조를 도시하는 도면이다. 상기 도 9에서 참조번호 310은 제1버전을 저장하는 제1저장영역으로써, 이때 상기 제1버전은 압축 또는 압축되지 않은 상태로 저장될 수 있다. 그리고 참조번호320은 적어도 하나의 제2버전 프로그램을 생성하기 위한 갱신패키지를 저장하는 제2저장영역으로써, 대응되는 제2버전의 갱신패키지들을 저장하기 위한 영역들이 될 수 있다. 즉, 상기 제2저장영역320에 저장되는 갱신패키지들은 각각 대응되는 버전의 갱신데이터 및 인스톨 데이터들이 될 수 있으며, 이때 상기 갱신 데이터는 상기 제1버전에서 변경되는 데이터(명령어, 실제 데이터 등)들 및 블록 인덱스들이 될 수 있다. 따라서 상기 제2저장영역320의 크기는 갱신패키지들의 수에 의해 결정될 수 있다. 예를들면 상기 도 9에 도시된 바와 같이 하나의 갱신패키지 저장영역의 크기가 300kbyte이고, 6개의 갱신패키지들을 저장하는 경우, 상기 제2저장영역320의 크기는 1.8Mbyte 크기로 할당할 수 있다. 그리고 참조번호330은 사용자 영역으로 파일 시스템(file system) 등이 저장될 수 있다.

[0073] 도 10a는 제1메모리250의 제2저장영역320에 저장되는 갱신패키지들의 구조를 도시하는 도면이며, 도 10b는 상기 도 10a와 같은 갱신패키지에서 히스토리데이터의 구조를 도시하는 도면이다.

[0074] 상기 도 10a를 참조하면 상기 제2저장영역320은 설정된 수(여기서는 6개로 가정하고 있다.)의 갱신패키지들을 저장할 수 있는 영역들을 구비하며, 각 갱신패키지의 저장영역은 대응되는 버전의 히스토리 데이터, 맵데이터 및 갱신데이터들을 저장할 수 있는 구조로 구성된다. 이때 상기 갱신패키지는 인스톨데이터 및 갱신데이터로 구성될 수 있으며, 상기 인스톨데이터는 상기한 바와 같이(도 4 및 도 5 참조) 히스토리데이터 또는 히스토리데이터 및 맵데이터가 될 수 있다. 또한 상기 제2저장영역320에는 갱신데이터의 히스토리데이터 및 맵 데이터들이 저장될 수 있다. 여기서 상기 갱신데이터의 히스토리 데이터를 저장하는 이유는 상기 제1저장영역310에 저장된 제1버전과 연결하기 위함이다. 그리고 상기 제1버전에 대한 맵 데이터 및 갱신데이터는 생략할 수 있으며, 또한 널 데이터(null data)로 채워넣을 수도 있다. 상기 도 10a는 상기 갱신패키지가 히스토리데이터, 맵데이터 및 갱신데이터 등으로 구성된 예를 도시하고 있다.

[0075] 도 10b를 참조하면, 상기 히스토리데이터 구조는 히스토리 데이터의 버전(version), 크기(size), 결합플래그(combined flag), 실패플래그(fail flag) 등으로 구성될 수 있다. 여기서 상기 버전(version)은 갱신패키지의 버전번호(도 10a의 경우 #2 - #7 중의 하나)가 될 수 있으며, 크기(size)는 해당 버전의 히스토리 데이터 크기가 될 수 있고, 결합플래그(combined flag)는 해당 버전의 갱신패키지를 결합하여 갱신할 버전번호(여기서는 제1버전의 번호 #1이 될 수 있음)가 될 수 있고, 실패플래그(fail flag)는 로딩 중에 실패(fail) 발생 여부를 표시하기 위한 플래그가 될 수 있다. 이런 경우 제1버전의 히스토리 데이터는 버전(version) 영역에 #1이 저장될 수 있으며, 결합플래그(combined flag) 영역에 #1이 연결될 수 있다. 또한 예를들어 임의의 갱신패키지 영역의 히스토리 데이터에서 버전(version) 영역에 #5가 저장되고 결합플래그(combined flag) 영역에 #1이 저장된 경우, 수신시스템30은 #5 번호의 제2버전을 상기 #1 번호인 제1버전에 결합(merge)하여 갱신하게 된다. 상기 다운로드되는 갱신패키지는 상기 도 10a와 같은 구조로 상기 도 9와 같은 제1메모리의 제2저장영역320에 저장된다. 이때 상기 제2저장영역320에 다운로드되는 갱신패키지는 갱신요구시 상기 제1저장영역310의 제1버전에 결합되어 갱신된 제2버전 프로그램을 생성하게 된다.

[0076] 이하의 설명에서는 상기 갱신패키지가 맵 데이터, 히스토리 데이터 및 갱신데이터로 구성된 경우를 가정하여 설명하기로 한다.

- [0077] 도 11은 상기 제1메모리250 및 제2메모리260의 구조를 도시하는 도면이다.
- [0078] 상기 도 11을 참조하면, 갱신요구시 도시하지 않은 로더는 제2저장영역320에서 갱신할 버전의 갱신팬키지를 로딩하고, 트랜스레이터240은 상기 로딩된 갱신팬키지를 상기 제1저장영역310의 제1버전에 머지하여 제2버전 프로그램을 생성한 후, 제2메모리260에 저장한다. 이때 상기 갱신요구는 수신시스템30의 사용자 응답에 의해 결정될 수 있다. 즉, 상기 수신시스템30은 상기 갱신팬키지를 다운로드하는 시점 또는 시스템 초기화 수행 후 갱신되지 않은 버전의 갱신팬키지가 존재하는 경우, 갱신 여부를 안내하는 메시지를 표시하며, 사용자가 갱신요구시 상기 와 같은 동작을 수행하면서 버전을 갱신한 후 갱신된 버전의 프로그램을 제2메모리260에 저장한다. 이후 상기 수신시스템30은 상기 제2메모리260에 저장된 제2버전 프로그램을 이용하여 수신시스템30을 운용한다.
- [0079] 또한 상기와 같은 갱신 동작은 수신시스템의 동작을 초기화한 후에 수행될 수 있다. 즉, 상기 도 9에 도시된 바와 같이 상기 제1 및 제2버전 프로그램은 서로 결합되지 않은 상태로 상기 제1메모리250에 저장되어 있으며, 갱신이 요구되는 시점에서 상기 제1버전 프로그램에 상기 제2버전의 갱신팬키지를 결합하여 제2버전 프로그램을 생성하고, 이 제2버전 프로그램을 제2메모리260에 저장하여 사용한다.
- [0080] 도 12는 제2버전이 상기 제1메모리250에 저장되지 않은 경우의 동작을 설명하기 위한 도면이며, 도 13은 제2버전의 갱신팬키지가 상기 제1메모리250의 제2영역320에 저장된 경우의 동작을 설명하기 위한 도면이다.
- [0081] 상기 도 12를 참조하면, 상기 제1메모리250은 제1버전을 저장하고 있으며, 여기서 상기 제1버전은 초기 버전(initial version)의 프로그램이 될 수 있다. 그리고 상기 제1버전은 상기 도 12에 도시된 바와 같이 n개의 블록(B#1-B#n)들로 이루어졌음을 가정하고 있다. 여기서 상기 제2버전의 인스톨 데이터는 히스토리 데이터와 맵 데이터로 구성될 수 있다. 여기서 상기 히스토리데이터는 상기한 바와 같이 상기 제1버전의 결합 정보들이 저장될 수 있다. 그리고 상기 도 12와 같은 경우 맵데이터 영역에는 C:B#1-#n이 저장될 수 있다.
- [0082] 이때 상기 갱신요구 명령이 발생되면, 트랜스레이터240은 상기 인스톨데이터를 분석한다. 이때 상기 도 12와 같은 경우 상기 제1메모리250의 제1버전을 그대로 상기 제2메모리260에 복사하라는 명령이므로, 상기 트랜스레이터240은 상기 제1메모리250의 제1버전 #1을 상기 제2메모리260에 그대로 복사한다. 그러면 상기 수신시스템30은 상기 제2메모리260에 상기 도 12와 같이 복사된 제1버전에 따라 수신시스템의 동작을 제어한다. 이때 상기 제1메모리250에 저장되는 버전들은 압축되어 저장될 수 있다. 따라서 상기 버전이 압축되어 저장된 경우, 상기 트랜스레이터240은 압축해제기270을 통해 상기 압축된 버전의 압축을 해제한 후 상기 제2메모리260에 복사한다. 또한 상기 갱신팬키지처리기10이 제2압축기140을 통해 상기 갱신팬키지를 압축하는 경우, 상기 트랜스레이터240은 먼저 상기 로딩되는 갱신팬키지의 압축을 해제한 후 상기와 같은 트랜스레이팅 동작을 수행할 수 있다.
- [0083] 상기 도 13을 참조하면, 상기 제1메모리250은 제1저장영역310에 제1버전 #1을 저장하고 있으며, 제2저장영역320에 제2버전의 갱신팬키지를 저장하고 있다. 여기서 상기 제1버전은 초기 버전(initial version) 또는 기준으로 설정된 버전이 될 수 있으며, 제2버전의 갱신팬키지는 갱신데이터 및 인스톨데이터들로 구성될 수 있다. 그리고 상기 인스톨데이터는 상기한 바와 같이 대응되는 제2버전 번호 및 결합하여야 할 버전(여기서는 제1버전)의 번호 등을 저장하는 히스토리데이터 및 맵 데이터들로 구성될 수 있다. 그리고 상기 제1버전은 상기 도 12에 도시된 바와 같이 n개의 블록(B#1-B#n)들로 이루어졌음을 가정하고 있다. 여기서 상기 히스토리데이터는 결합플래그 영역에 #1이 저장된 경우를 가정하고 있으며, 상기 맵데이터 영역에는 C:B#1-B#2, M:B#3, S:[B#4-B#n+1][B#3-B#n]이 저장된 경우를 가정하고 있다. 이런 경우 상기 맵데이터는 제2버전의 B#1-B#2는 제1버전의 B#1-B#2를 복사(copy)하고, 제2버전의 B#3은 갱신 데이터이고, 제2버전의 B#4-B#n+1은 쉬프트된 블록들으로써 제1버전의 B#3-B#n와 동일함을 의미한다.
- [0084] 이때 상기 갱신요구 명령이 발생되면, 트랜스레이터240은 상기 인스톨데이터를 분석한다. 이때 상기 도 13과 같은 경우 상기 제1메모리250의 제1버전 프로그램에 갱신데이터 UP #2를 갱신하여 상기 제2메모리260에 복사하여야 한다. 이때 상기 트랜스레이터240은 상기 제1메모리250의 제1버전에서 B#1 - B#2의 데이터를 복사하며, B#3의 데이터를 UP#2의 데이터로 변경하고, B#4 이후의 데이터는 제1버전의 B#3 이후의 데이터를 이동시킨 후 이를 상기 제2메모리260에 복사한다. 이런 경우 상기 제2메모리260에는 도 13과 같이 B#3의 데이터는 UP#2의 갱신데이터로 변경되고, B#4의 데이터는 설정된 블록 수만큼(여기서는 1 블록) 쉬프트된 상태가 된다. 이후 상기 수신시스템30은 상기 제2메모리260에 상기 도 13과 같이 갱신된 제2버전에 따라 수신시스템30의 동작을 제어한다. 이때 상기 제1메모리250에 저장되는 제1버전 및 제2버전의 갱신팬키지들은 전송시 압축되어 저장될 수 있다.(갱신팬키지처리기10에서 생성되는 갱신팬키지를 제2압축기140을 사용하여 전송하는 경우) 따라서 상기 제1버전 및 갱신팬키지들이 압축되어 저장된 경우, 상기 트랜스레이터240은 압축해제기270을 통해 상기 압축된 제1버전 및 갱신팬키지들의 압축을 해제할 수 있다. 또한 상기 제1버전 및 제2버전의 프로그램들을 압축한 후(갱신팬키지처

리기10의 제1압축기160을 이용하여 압축하는 경우), 압축된 제1버전 및 제2버전의 블록들을 비교하면서 갱신패키지를 생성한 경우, 상기 트랜스레이터240에서 트랜스레이팅한 제2버전의 블록 데이터들은 압축된 데이터들이 된다. 이런 경우 상기 트랜스레이터240은 압축해제기275를 이용하여 압축을 해제한 후 제2메모리260에 저장한다.

[0085] 도 14는 상기 수신시스템30이 상기 제1버전 프로그램에 갱신패키지를 결합하여 제2버전 프로그램을 생성하는 절차를 종합적으로 설명하기 위한 도면이다.

[0086] 상기 도 14를 참조하면, 상기 제1메모리250은 상기한 바와 같이 제1버전 프로그램250 및 제2버전의 갱신패키지210들을 저장하고 있으며, 갱신 요구시 트랜스레이터240은 상기 갱신패키지210을 상기 제1버전 프로그램250에 결합하여 제2버전 프로그램을 생성한 후 이를 제2메모리260에 저장한다. 따라서 수신 시스템30은 제1메모리250에 저장하고 있는 제1버전의 프로그램250 및 제2버전의 갱신패키지210을 제2메모리260에서 결합(merge)하여 사용하며, 상기와 같은 갱신은 시스템 초기화 수행 후 또는 갱신 요구에 의해 반복적으로 수행된다.

[0087] 상술한 바와 같이 본 발명의 실시예에 따른 시스템의 정보 갱신 방법은 갱신 패키지를 생성하고, 상기 생성된 갱신패키지를 결정된 네트워크 통신 방법에 따라 다운로드하고, 상기 다운로드된 갱신패키지를 인스톨하여 제1저장하고, 상기 제1저장된 갱신패키지를 이용하여 시스템의 갱신 프로그램을 생성하여 제2저장한 후, 상기 제2저장된 갱신 갱신 프로그램을 이용하여 시스템을 운용한다. 이를 위하여 본 발명의 실시예에서는 먼저 빠른 갱신을 위한 갱신패키지 생성 방법을 제안하고, 두 번째로 갱신패키지를 다운로드한 후 빠른 실행을 위한 인스톨 데이터 처리 방법을 제안하며, 세 번째로 상기 갱신패키지를 다운로드하는 시스템에서 다운로드된 갱신패키지를 보관 및 처리하는 방법을 제안하며, 네 번째로 상기 보관하고 있는 갱신패키지를 이용하여 시스템 갱신을 수행하는 방법을 제안한다.

[0088] 먼저 상기 빠른 갱신을 위한 갱신패키지의 생성 방법을 살펴보면, 제1버전의 프로그램과 제2버전의 프로그램을 갱신패키지의 생성을 위한 입력으로 하며, 두 버전은 상기 갱신패키지 생성시 압축하거나 또는 압축하지 않은 상태로 비교된 후, 상기 두 버전들 간에 차이들을 찾아낸다. 이때 상기 두 버전들 간의 차이들은 이를 수신하는 시스템에서 제1버전에 상기 갱신 패키지를 재조합하기 위한 인스톨데이터(install data: 특히 map data가 될 수 있음)로 생성된다. 그리고 상기 인스톨데이터와 함께 갱신데이터를 생성한 후, 이를 갱신패키지로 생성한다.

[0089] 두 번째로 상기와 같이 생성된 갱신패키지는 수신시스템에 다운로드된다. 이때 상기 수신시스템은 상기 갱신패키지를 다운로드한 후 빠른 실행을 위한 갱신 패키지 처리 절차를 수행한다. 상기 갱신패키지에 포함되는 인스톨데이터는 기준이 되는 프로그램(여기서는 제1버전 프로그램)과의 비교를 통해서 얻을 수 있으며, 빠른 주소 계산을 목적으로 사용된다. 즉, 제1메모리250에 저장된 제1버전과 상기 다운로드되는 갱신패키지를 제2메모리260에서 결합할 때, 상기 인스톨데이터를 이용하여 빠르게 블록별로 처리할 수 있다.

[0090] 세 번째로 갱신패키지를 다운로드한 후 빠른 실행을 위한 인스톨 처리 방법을 살펴보면, 상기 인스톨 데이터는 기준 프로그램과의 비교를 통해서 얻을 수 있는 맵데이터를 찾는 생성방법으로 빠른 주소 계산을 목적으로 사용되며, 제1메모리250에 포함되어 있는 제1버전과 갱신 패키지를 제2메모리260에서 재조립하기 위한 도움 역할로 사용된다. 이때 상기 맵데이터가 상기 갱신패키지에 포함되어 있지 않은 경우, 수신시스템30은 역으로 제1메모리250에 포함된 상기 제1버전 프로그램과 갱신 데이터를 비교하여 통해 해당 맵 기능을 위한 데이터를 생성할 수 있다. 또한 상기 인스톨데이터는 갱신패키지의 히스토리 데이터를 포함하고 있다. 이때 상기 히스토리데이터는 상기 제1메모리250에 저장되는 갱신패키지 및 갱신패키지가 결합될 버전을 나타내는 데이터들을 포함하고 있다. 본 발명의 실시예에서는 상기한 바와 같이 6개의 갱신패키지를 상기 제1메모리에 보관하는 것을 가정하여 설명하고 있으며, 이때 특정 갱신패키지가 제2메모리260에서 결합 실패가 발생하는 경우, 수신시스템30이 보관하고 있는 다른 갱신패키지들의 리스트를 표시하고, 사용자의 선택에 따른 갱신패키지로 갱신할 수 있다.

[0091] 네 번째로 갱신패키지의 보관 및 처리 방법을 살펴보면, 상기 갱신패키지는 제1메모리250 상에 구성되는 특정 저장영역들에 보관되며, 상기 저장영역들은 복수개로 구비된다. 따라서 상기 갱신패키지는 다운로드되어 제1메모리250에 저장되고 그 다음 갱신패키지가 다운로드되는 경우에도 이전 갱신패키지는 지워지지 않는다. 그리고 특정 갱신패키지가 로딩되지 않는 경우, 수신시스템30은 사용자에게 보관하고 있는 갱신패키지들의 리스트를 제공하여 특정 버전을 재조합하기 위한 선택을 지원한다. 그리고 최악의 경우 모든 갱신패키지들의 로딩이 실패되는 경우에는 최초 버전인 제1버전을 로딩하여 사용할 수 있다.

[0092] 도 15는 본 발명의 실시예에 따른 갱신패키지 생성 및 갱신 절차를 설명하는 흐름도이다. 상기 도 15의 동작은 상기 도 1에서 갱신패키지처리기10 및 수신시스템30의 동작을 중심으로 도시되어 있다.

- [0093] 상기 도 15를 참조하면, 갱신패키지처리기10은 411단계에서 갱신패키지를 생성하기 위한 버전들을 수신한다. 여기서 상기 갱신패키지는 새로운 버전의 프로그램이 생성되는 경우에 생성될 수 있으며, 상기 생성되는 갱신 패키지는 기준이 되는 버전 프로그램과 비교되어 생성되는 데이터가 될 수 있다. 본 발명의 실시예에서 상기 기준이 되는 버전은 제1버전이라 칭하고 있으며, 상기 새로운 버전은 제2버전이라 칭하고 있다. 이때 상기 제1버전은 최초 버전 또는 다른 버전들과 결합될 수 있는 기준 버전이 될 수 있다. 즉, 상기 제1버전은 갱신되는 버전들과 결합되는 기준 버전이 될 수 있다. 그리고 상기 제2버전의 갱신패키지는 상기 제1버전과 결합되어 갱신된 제2버전의 프로그램을 생성하는 정보로써, 상기 제2버전의 갱신패키지는 적어도 하나 이상이 될 수 있다. 따라서 상기한 바와 같이 제1 및 제2버전들이 수신되면, 상기 갱신패키지처리기10은 413단계에서 상기 두 버전들의 차이를 분석하며, 415단계에서 상기 두 버전들의 차이에 따른 갱신패키지를 생성한다. 이때 상기 갱신패키지는 상기 두 버전들의 차이에 따라서 상기 제1버전에 갱신하여야 하는 갱신데이터와, 상기 갱신데이터를 상기 제1버전에 결합하기 위한 정보인 인스톨데이터 등으로 구성될 수 있다. 그리고 상기 인스톨데이터는 상기 제2버전의 히스토리를 표시하는 히스토리 데이터와, 상기 갱신데이터를 제1버전에 갱신할 때의 매핑 정보를 나타내는 맵 데이터등으로 구성될 수 있다. 이때 상기 맵데이터는 갱신패키지에서 생략될 수 있으며, 이런 경우 상기 맵데이터는 수신시스템30에서 상기 갱신패키지를 인스톨할 때 생성할 수 있다.
- [0094] 상기와 같이 생성되는 갱신패키지는 갱신패키지서버20에 전달되며, 상기 갱신패키지서버20은 수신시스템30들에 상기 갱신패키지의 생성을 통지한다. 그리고 상기 갱신패키지의 발생을 감지한 수신시스템30이 상기 갱신패키지 서버20에 다운로드를 요구하면, 상기 갱신패키지서버20은 대응되는 수신시스템30에 갱신패키지를 다운로드한다. 이때 상기 다운로드 방법은 상기 수신시스템30의 통신 방법에 따라 결정된다. 즉, 상기 수신시스템30이 무선 통신 기능을 구비하는 휴대단말기이면, 무선통신 방식에 따라 CDMA, UMTS, GSM 등과 같은 무선 통신 방식으로 다운로드되며, 또한 인터넷 망과 접속된 경우에는 USB, 블루투스, 와이브로, 무선랜 등과 같은 통신 방법으로 상기 갱신패키지를 다운로드할 수 있다.
- [0095] 상기 수신시스템30은 451단계에서 상기 갱신패키지를 다운로드한 후, 453단계에서 상기 다운로드되는 갱신패키지를 상기 제1메모리250에 저장한다. 이때 상기 제1메모리250은 상기 제1버전의 프로그램을 저장하는 제1영역 310 및 상기 갱신패키지들을 저장하는 저장영역들로 구성되는 제2영역320으로 구성된다. 그리고 상기 제2영역 320은 다수개의 저장영역들로 구성되어 복수의 제2버전정보들을 저장할 수 있다. 본 발명의 실시예에서는 상기 제2영역320이 6개의 저장영역들로 구성되는 것으로 가정하고 있다. 그리고 상기 갱신패키지를 저장하는 저장영역은 인스톨 데이터인 히스토리데이터 및 맵데이터들을 저장하는 영역과 갱신데이터를 저장하는 영역 등으로 구성된다. 이때 상기 갱신패키지를 인스톨하는 과정에서 상기 갱신패키지에 맵데이터가 포함되지 않은 경우, 상기 수신시스템30의 인스톨러230은 상기 제1버전 프로그램에 상기 갱신데이터를 매핑하여 갱신하기 위한 맵데이터를 생성하여 상기 제1메모리250에 저장할 수 있다. 상기와 같이 갱신패키지가 상기 제1메모리250에 저장되면, 상기 수신시스템30은 갱신 요구시 455단계에서 상기 제1버전 프로그램에 상기 갱신패키지를 머지하여 제2 버전의 프로그램을 생성한 후 이를 제2메모리260에 저장하며, 상기 저장된 제2버전 프로그램에 따라 시스템을 운용한다. 이때 상기 제2메모리260은 워크메모리(work memory)가 될 수 있으며, 이런 경우 상기 제2메모리260은 휘발성 메모리로 구현할 수 있다. 따라서 상기 수신시스템30은 시스템 초기화 과정 동작이 수행되거나 또는 상기 갱신패키지가 다운로드된 후 사용자에게 의해 갱신 요구 명령이 발생된 경우, 상기 제1메모리250에 저장된 제1버전 및 가장 최근에 다운로드된 제2버전의 갱신패키지를 결합시켜 생성되는 제2버전의 프로그램을 제2메모리260에 저장하며, 시스템은 상기 제2메모리260에 저장된 상기 제2버전을 이용하여 시스템을 운용한다. 또한 상기 수신시스템30은 상기 프로그램의 갱신이 실패하는 경우, 이전 버전 프로그램을 이용하여 자동으로 프로그램 갱신을 시도할 수 있다. 또한 상기 수신시스템30은 프로그램 갱신이 실패하는 경우, 이전 버전들의 목록을 표시하고, 사용자에게 의해 선택된 버전을 이용하여 수동으로 프로그램 갱신을 수행할 수 있다. 또한 상기 시스템 초기화시 사용자는 원하는 프로그램의 버전을 선택할 수 있으며, 상기 수신시스템30은 상기 사용자에게 의해 선택된 버전의 프로그램을 시스템의 프로그램을 갱신할 수도 있다.
- [0096] 이하의 설명에서 갱신패키지처리기10, 갱신패키지 서버20 및 수신시스템30의 동작 절차를 구체적으로 살펴본다.
- [0097] 먼저 상기 갱신 패키지를 생성하는 절차를 상세하게 살펴본다.
- [0098] 도 16a - 도 16c는 상기 갱신패키지를 생성하는 절차를 도시하는 흐름도이다.
- [0099] 상기 도 16a를 참조하면, 먼저 상기 갱신패키지처리기10은 501단계에서 상기 제1 및 제2버전의 프로그램을 입력한다. 이후 상기 갱신패키지처리기10은 503단계에서 상기 제1 및 제2버전의 프로그램을 설정된 크기의 블록 단위로 비교하여 두 버전들 간의 차이를 확인한 후, 이를 비교데이터로 발생한다. 그리고 상기 갱신패키지처리기

10은 505단계 및 507단계에서 상기 비교데이터를 이용하여 갱신데이터 및 인스톨데이터를 생성한다. 이때 상기 갱신패키지처리기10은 각각 패키지발생기130 및 인스톨발생기180을 각각 구비하는 경우, 도 16a에 도시한 바와 같이 상기 갱신데이터 및 인스톨데이터를 병렬로 생성할 수 있다. 그러나 상기 패키지발생기130 및 인스톨발생기180을 구비하지 않는 경우, 상기 갱신패키지처리기10은 상기 갱신데이터 및 인스톨데이터를 순차적으로 생성할 수 있다. 이때 상기 갱신데이터 및 인스톨데이터의 생성 순서는 갱신데이터-인스톨데이터의 순서로 생성할 수 있으며, 또한 인스톨데이터-갱신데이터의 순서로 생성할 수도 있다.

[0100] 여기서 상기 인스톨데이터는 수신측에서 상기 제1버전에 갱신패키지를 결합(merge)하기 위해 사용할 데이터들으로써, 상기한 바와 같이 히스토리 데이터와 맵 데이터들이 있을 수 있다. 이때 상기 히스토리데이터는 상기한 바와 같이 해당하는 갱신패키지의 버전, 갱신패키지가 결합될 버전(여기서는 제1버전이 될 수 있음), 크기 등으로 이루어질 수 있으며, 맵 데이터는 상기 제1버전 프로그램이 각 블록 데이터들과 제2버전 프로그램의 각 블록 데이터를 결합하기 위한 매핑 정보(mapping information)들이 될 수 있다. 이때 상기 인스톨 데이터에서 상기 맵데이터는 상기 갱신패키지처리기10에서 생성할 수 있으며, 또한 수신측에서도 생성할 수 있다. 따라서 상기 맵데이터는 상기 갱신패키지에 포함시킬 수 있으며, 또한 포함시키지 않을 수도 있다.

[0101] 따라서 상기 갱신패키지처리기10은 도 16b와 같이 인스톨데이터 생성시 맵데이터를 생성하거나 또는 생성하지 않을 수 있다. 그리고 도 16c와 같이 인스톨 데이터를 상기 갱신 패키지에 결합(merge)할 때 상기 맵데이터를 포함시켜 합성할 수 있으며, 또한 상기 맵데이터를 생략하고 결합할 수도 있다. 상기 도 16b를 참조하면, 상기 갱신패키지처리기10은 521단계에서 상기 히스토리데이터를 생성한 후, 523단계에서 맵데이터 생성이 필요한가를 검사한다. 이때 맵데이터 생성이 필요하다면, 상기 갱신패키지처리기10은 525단계에서 상기 비교데이터를 이용하여 상기 갱신데이터를 매핑하기 위한 맵데이터를 생성한 후 상기 도 16a로 리턴한다. 그리고 상기 도 16c를 참조하면, 상기 갱신패키지처리기10은 531단계에서 상기 인스톨데이터를 상기 갱신패키지에 머지할 때 맵데이터도 함께 머지하는가를 검사한다. 이때 상기 갱신패키지처리기10은 상기 맵데이터를 머지하는 경우 533단계에서 상기 히스토리 및 맵데이터로 구성되는 인스톨데이터를 상기 갱신패키지에 합성하여 갱신패키지를 생성하며, 맵데이터를 머지하지 않는 경우 535단계에서 상기 히스토리 데이터만으로 구성된 인스톨데이터를 상기 갱신패키지에 결합하여 갱신패키지를 생성한다.

[0102] 또한 상기 도 16a에서 상기 501단계에서 로드되는 제1 및 제2버전의 프로그램들을 압축한 후(예를들면 상기 갱신패키지처리기10의 제1압축기160을 통해 수행할 수 있음) 503단계로 진행할 수 있으며, 또한 상기 509단계를 수행하여 생성된 갱신패키지를 압축할 수 있다.(예를들면 상기 갱신패키지처리기10의 제2압축기140을 통해 수행할 있음) 이때 압축 부호화 동작을 수행하는 경우, 상기 503단계에서 비교하는 버전들의 크기가 줄어들어 데이터 처리 시간을 줄일 수 있으며, 또한 갱신패키지를 압축 부호화하는 경우에는 전송되는 정보의 양을 줄일 수 있게 된다. 이때 상기 압축부호화 동작을 수행하는 경우, 상기 갱신패키지처리기10은 이를 다시 압축 해제하여 정상적으로 압축부호화가 이루어졌는가 확인하며, 정상적으로 압축 부호화가 이루어진 경우에만 다음 단계로 진행하도록 제어할 수 있다.

[0103] 도 17은 본 발명의 실시예에 따라 갱신패키지처리기10에서 갱신패키지를 생성하는 절차를 도시하는 흐름도이며, 도 18은 도 17에서 제1 및 제2버전을 압축 및 검증하는 절차를 도시하는 흐름도이고, 도 19는 도 17에서 인스톨데이터를 생성하는 절차를 도시하는 흐름도이며, 도 20은 도 17에서 갱신패키지를 생성하는 절차를 도시하는 흐름도이다.

[0104] 상기 도 17 - 도 20을 참조하면, 상기 갱신패키지처리기10은 551단계에서 컨피규어 파일(configure file)들을 입력한다. 이때 상기 컨피규어 파일에는 상기 갱신패키지처리기10의 동작을 규정하는 플래그들의 설정 여부를 확인할 수 있다. 여기서 상기 플래그들 중에서 C_FLAG는 압축 여부를 설정하는 플래그(compression flag)이고, M_FLAG는 맵 데이터의 생성 여부를 결정하는 플래그(map generation flag)이고, V_FLAG는 압축된 정보를 압축 해제하여 검토하므로써 정상적으로 압축되었는지를 검증하는 플래그(verify flag)이다. 이후 갱신패키지처리기10은 553단계에서 제1 및 제2버전의 프로그램들을 로드한다. 여기서 상기 제1버전 프로그램은 상기한 바와 같이 휴대단말기의 최초 버전 또는 버전 갱신을 위한 기준 버전 프로그램이 될 수 있으며, 상기 제2버전 프로그램은 최근의 갱신 버전의 프로그램이 될 수 있다.

[0105] 상기 두 버전의 프로그램들을 로드한 후, 갱신패키지처리기10은 555단계에서 두 버전 프로그램들의 압축부호화가 필요한가(C_FLAG=1)를 검사한다. 이때 압축부호화가 필요하지 않은 경우, 갱신패키지처리기10은 561단계에서 상기 제1 및 제2버전프로그램들을 각각 컨피규어 버전으로 포맷한 후 563단계로 진행하여 두 버전 프로그램들의 비교를 수행한다. 그러나 상기 555단계에서 압축부호화가 필요한 경우, 갱신패키지처리기10은 557단계에서 제1

압축기160을 실행시킨 후, 559단계에서 상기 제1 및 제2버전 프로그램들을 상기 제1압축기160을 통해 압축부호화한 후, 563단계에서 상기 압축된 두 버전 프로그램들을 비교기110을 통해 비교한다.

[0106] 이때 상기 559단계에서 상기 두 버전 프로그램들을 압축하는 절차는 도 18과 같은 절차로 수행된다. 이때 상기 압축플래그가 설정된 경우에는 검증플래그도 함께 설정되는 것이 일반적이다. 이때 상기 검증플래그(V_FLAG)가 세트되면, 상기 갠신패키지처리기10은 상기 압축데이터의 압축을 해제한 후 상기 압축해제된 데이터를 압축전의 데이터와 비교하여 정상적으로 압축이 이루어졌는지는 검증하게 된다.

[0107] 상기 도 18을 참조하면, 상기 갠신패키지처리기10은 상기 제1 및 제2버전프로그램들을 각각 압축한 후, 601단계에서 검증이 필요한가(V_FLAG=1)를 검사한다. 이때 상기 검증이 필요한 경우, 상기 갠신패키지처리기10은 603단계에서 압축해제기165를 실행시켜 상기 압축된 제1 및 제2버전프로그램들의 압축을 해제한 후, 605단계에서 상기 압축전의 버전과 압축해제한 후의 버전 프로그램을 비교하며, 607단계에서 상기 비교 결과 압축전 및 압축해제 후의 버전이 동일한 값을 가지는가 검사한다. 이때 상기 압축전 및 압축해제 후의 버전이 동일하면, 상기 갠신패키지처리기10은 607단계에서 이를 확인하고, 609단계에서 상기 압축된 버전들을 상기 비교기110에 인가하는 동시에 정상 상태임을 검증하는 통지를 발생한다. 그러나 상기 607단계에서 상기 압축전 및 압축 해제한 후의 데이터가 동일하지 않은 경우 상기 갠신패키지처리기10은 611단계에서 이를 에러처리한 후 갠신패키지 처리전으로 리턴한다.

[0108] 상기과 같이 압축된 또는 압축되지 않은 제1버전 프로그램 및 제2버전프로그램이 발생되면, 상기 갠신패키지처리기10은 563단계에서 비교기110을 실행시킨 후, 565단계에서 인스톨데이터발생기180을 실행시킨다. 이때 상기 인스톨데이터발생기180은 맵데이터를 생성할 수 있으며, 또한 맵데이터를 생성하지 않을 수도 있다. 이는 M_FLAG에 의해 결정된다. 따라서 상기 갠신패키지처리기10은 567단계에서 맵데이터를 생성하여야 하는가(M_FLAG=1)를 확인한다. 이때 상기 맵데이터를 생성하는 경우(M_FLAG=1), 상기 갠신패키지처리기10은 567단계에서 이를 감지하고 569단계에서 상기 제1버전과 상기 제2버전의 프로그램을 비교기110을 통해 비교한다. 이때 상기 제1 및 제2버전 프로그램의 비교는 블록 크기로 비교할 수 있으며, 상기 제1버전 및 제2버전 프로그램의 블록 데이터가 동일하지 않은 경우에는 현재의 제2버전 프로그램의 블록 데이터를 상기 제1버전 프로그램의 설정된 탐색 구간 내의 블록 데이터들을 비교하여 일치 여부를 판단한다. 즉, 상기 비교기110은 현재의 제2버전 블록 데이터를 상기 제1버전의 설정된 탐색 구간 내의 블록 데이터들과 비교하여 일치 여부를 판정한다.

[0109] 이때 상기 제1버전 및 제2버전은 압축된 버전정보일 수 있으며, 또한 압축되지 않은 버전일 수 있다. 그리고 상기 제1 및 제2버전의 비교를 완료하면, 상기 갠신패키지처리기10은 571단계에서 맵데이터를 생성하기 위하여 상기 비교 데이터를 인스톨데이터발생기180에 전달하는 동시에 577단계로 진행하여 맵데이터 및 갠신패키지데이터를 생성하기 위해 상기 비교 결과 값을 저장영역에 저장한다. 그러나 상기 맵데이터를 생성하지 않아도 되는 경우(M_FLAG=0), 상기 갠신패키지처리기10은 567단계에서 이를 감지하고 575단계에서 상기 제1버전 프로그램과 상기 제2버전 프로그램을 비교기110을 통해 비교한다. 이때 상기 제1버전 및 제2버전은 상기한 바와 같이 압축된 버전일 수 있으며, 또한 압축되지 않은 버전일 수 있다. 그리고 상기 제1 및 제2버전의 비교를 완료하면, 상기 갠신패키지처리기10은 577단계에서 갠신패키지데이터를 생성하기 위해 상기 비교 결과 값을 저장영역에 저장한다. 상기한 바와 같이 맵데이터를 생성하는 경우에 상기 갠신패키지처리기10은 상기 두 버전들의 비교 결과 값을 인스톨데이터발생기180에 전달하는 동시에 저장영역에 저장하여 맵 데이터 및 갠신패키지데이터를 생성할 수 있도록 하며, 상기 맵데이터를 생성하지 않는 경우에 상기 갠신패키지처리기10은 상기 두 버전들의 비교 결과 값을 저장영역에 저장하여 갠신패키지데이터를 생성할 수 있도록 한다.

[0110] 이후 상기 갠신패키지처리기10은 579단계에서 인스톨데이터발생기180을 구동시켜 인스톨데이터를 생성하며, 상기 인스톨데이터의 생성은 도 19와 같은 절차로 진행된다.

[0111] 상기 도 19를 참조하면, 인스톨 데이터 생성시 상기 갠신패키지처리기10은 651단계에서 인스톨데이터의 생성을 활성화시킨 후, 653단계에서 두 버전들의 히스토리 데이터를 검사한다. 이후 상기 갠신패키지처리기10은 655단계에서 히스토리데이터 발생기120을 구동하고 657단계에서 히스토리 데이터를 생성한다. 이때 상기 히스토리 데이터는 도시된 바와 같이 헤더(header), 제1버전프로그램의 정보(first input version info), 제2버전프로그램의 정보(second input version info) 및 메모리 정보(memory info) 등으로 구성될 수 있다. 그리고 상기 제1 및 제2버전프로그램의 정보들은 각각 버전 식별정보(SW VER; 0x2AB, 0x2AC), 타임 스탬프(SW time stamp), 체크섬정보(SW checksum) 등으로 이루어질 수 있다. 따라서 상기 히스토리 데이터는 갠신패키지 기준이 되는 버전(여기서는 제1버전으로써 최초 버전임)과 상기 기준 버전에서 갠신패키지되어야 하는 데이터를 제공하는 갠신패키지(여기서는 제2버전)의 식별정보들이 상기 히스토리데이터에 포함된다.

- [0112] 이후 상기 갱신팬키지처리기10은 659단계에서 상기 맵데이터를 생성하여야하는가(M_FLAG=1)를 확인한다. 이때 맵데이터를 생성하여야 하는 경우, 상기 갱신팬키지처리기10은 603단계에서 맵데이터 발생기150을 구동한 후 665단계에서 맵 데이터를 생성한다. 이때 상기 맵데이터는 상기 갱신 데이터를 상기 기준버전인 제1버전에 포함시켜 갱신하는 경우, 상기 갱신데이터의 위치 정보 및 갱신 형태 등을 지정하는 매핑 정보가 될 수 있다. 여기서 상기 맵데이터는 상기한 바와 같이 복사C, 변경M 및 이동S 및 관련된 블록 인덱스 정보 등이 될 수 있으며, 상기 맵 데이터는 소정 크기를 가지는 매크로 블록으로 분할되어 설정된다. 이때 상기 맵데이터는 상기 두 버전의 비교 결과 값에 따라 생성되는데, 두 버전의 블록 데이터가 동일하면 해당 블록의 맵데이터는 복사C로 설정하며, 두 버전정보의 블록데이터 값이 다르거나 블록 데이터가 추가된 경우에는 해당 블록의 맵데이터를 변경 M으로 설정하고, 두 버전의 블록 데이터 값이 동일하지만 그 블록의 인덱스가 다른 경우에는 동일한 위치까지의 이동 S로 설정한다. 상기 도 19에 도시된 바와 같이 맵데이터가 설정된 경우, 제2버전 프로그램의 블록번호들은 #0000-#1F00이며, 여기서 상기 제1버전정보와 상기 제2버전정보의 비교 결과, #0000 - #0010, #0024 - #0220의 블록들은 동일하고, #0011 - #0023, #0221 블록은 변경된 블록이며, #0222 - #1F00는 블록 내의 위치만 다르고 이동하면 동일한 값들을 가지는 블록이다. 상기와 같은 경우, 제2버전의 #0011 - #001F, #0020 - #0023 블록은 상기 제1버전의 #0011 - #001F, #0020 - #0023 블록을 대체하는 변경이 발생된 경우이며, 제2버전의 #0221 블록은 상기 제1버전의 #0220 및 0221 사이에 삽입되는 변경 블록이 발생된 예가 될 수 있다. 그러면 상기 맵데이터발생기150은 C:B#0000 - B#0010, M:B#00011 - B#0023, C:B#0024 - B#0220, M:B#0221, S:[B#0222 - B#1F00][B#0221 - 1EFF]의 맵 데이터를 생성하게 되며, 이때 상기 갱신데이터는 상기 변경 M에 관련되는 블록들의 데이터로 생성할 수 있다. 상기와 같이 맵 데이터를 생성한 후, 상기 갱신팬키지처리기10은 667단계에서 상기 히스토리 데이터에 상기 맵데이터를 머지하여 인스톨데이터로 생성한다.
- [0113] 따라서 상기 661단계까지 수행하고 나면, 상기 갱신팬키지처리기10은 상기 M_FLAG의 설정 여부에 따라 인스톨데이터를 생성하게 되는데, 이때 생성되는 인스톨데이터는 히스토리데이터만으로 구성되거나 또는 상기 히스토리 데이터 및 맵데이터가 머지된 형태의 데이터가 될 수 있다.
- [0114] 상기와 같이 인스톨 데이터를 생성한 후, 상기 갱신팬키지처리기10은 581단계에서 패키지발생기130을 실행시키고 583단계에서 상기 패키지 발생기130에 의해 갱신팬키지를 생성한다. 도 20은 상기 갱신팬키지를 생성하는 절차를 도시하는 흐름도이다.
- [0115] 상기 도 20을 참조하면, 상기 갱신팬키지처리기10은 621단계에서 상기 비교 결과에 따른 갱신데이터를 발생한다. 여기서 상기 패키지발생기130은 상기 제1버전과 제2버전에서 비교 결과 값이 설정된 탐색 구간 내에서 동일한 블록 데이터를 찾지 못한 경우 해당 블록의 데이터를 갱신데이터로 설정할 수 있다. 그리고 상기 갱신데이터의 매핑 위치는 상기 맵 데이터에 의해 결정된다. 즉, 상기 제2버전의 블록데이터가 상기 제1버전의 탐색 구간에서 상기 제1버전의 블록 데이터의 크기와 다르거나 서로 다른 데이터인 경우, 해당 해당하는 블록의 데이터를 갱신데이터로 설정하며, 상기 갱신데이터는 맵데이터에서 변경 M에 대응되는 블록들이 될 수 있다.
- [0116] 또한 상기 패키지발생기130은 상기 인스톨 데이터가 맵 데이터를 포함하지 않는 경우, 상기 갱신데이터에 상기 제1버전과 결합하기 위한 제2버전의 각 블록들의 블록 인덱스 값을 포함시켜 갱신데이터를 생성할 수도 있다. 이런 경우 상기 갱신데이터는 복사 C, 변경 M 및 쉬프트 S의 명령어 포맷으로 생성될 수 있다.
- [0117] 상기 갱신데이터는 압축하여 다운로드하는 것이 바람직하다. 따라서 상기 갱신팬키지처리기10은 623단계에서 제2압축기140을 구동한 후, 625단계에서 상기 갱신데이터를 압축하며, 또한 상기 갱신데이터의 압축이 정상적으로 이루어졌는지 검증하기 위하여 627단계 및 629단계에서 상기 압축된 갱신데이터의 압축을 해제한 후 압축 전 및 압축 해제 후의 데이터를 비교하여 검증한다. 이때 상기 검증이 정상적으로 이루어지면, 상기 갱신팬키지처리기10은 631단계에서 이를 감지하고 633단계에서 상기 갱신데이터에 상기 인스톨데이터를 합성하여 갱신팬키지를 생성하고, 635단계에서 상기 갱신팬키지를 상기 갱신팬키지서버20에 전달한다. 그러나 상기 631단계에서 상기 검증이 실패된 경우, 상기 갱신팬키지처리기10은 637단계에서 에러 처리한다.
- [0118] 상기와 같이 갱신팬키지처리기10에서 갱신팬키지가 생성되면, 상기 갱신팬키지는 갱신팬키지서버20에 전달되어 이를 필요로 하는 수신시스템30들에 다운로드된다. 이때 상기 갱신팬키지는 기준버전인 제1버전과 갱신버전인 제2버전 간의 차를 가지는 갱신데이터 및 이런 갱신데이터를 인스톨하기 위한 데이터들로 구성된다.
- [0119] 도 21은 본 발명의 실시예에 따라 갱신팬키지를 전달받는 갱신팬키지서버20이 상기 갱신팬키지를 상기 수신시스템30들에 다운로드하는 절차를 도시하는 흐름도이다. 여기서 상기 갱신팬키지서버20은 상기 갱신팬키지처리기10과 일체형으로 구성할 수 있으며, 또한 분리하여 구성할 수도 있다.

- [0120] 상기 도 21을 참조하면, 상기 갱신파키지를 전달받으면 상기 갱신파키지서버20은 711단계에서 상기 갱신파키지가 존재함을 각 수신시스템30들에 통지한다. 이때 상기 갱신파키지서버20과 상기 수신시스템30간의 통신 방법은 상기 수신시스템30과 갱신파키지서버20 간의 연결 방법에 따라 결정된다. 즉, 상기 수신시스템30이 휴대용 단말기인 경우, 상기 수신시스템30은 공중 무선통신(예를들면 CDMA, GSM, GSM, GPRS 등) 방법으로 통신 기능을 수행한다. 또한 상기 수신시스템30이 컴퓨터이거나 컴퓨터를 통해 연결된 휴대단말기이면, 상기 갱신파키지의 통신을 인터넷망을 통해 수행할 수 있다. 여기서 상기 휴대단말기가 무선 통신 기능을 구비하는 경우, 상기 인터넷망은 무선 인터넷(wibro, wimax, wifi, 무선랜 등)을 사용할 수도 있다. 이후 상기 수신시스템30이 이에 대한 응답을 하면, 상기 갱신파키지서버20은 713단계에서 이를 수신한 후 715단계에서 다운로드를 허용함을 알린다. 이후 상기 수신시스템30이 이에 따른 응답을 하면, 상기 갱신파키지서버20은 717단계에서 이를 수신하고, 719단계에서 관리정보(management info)를 해당 수신시스템30에 전송한다. 이후 상기 수신시스템30이 이에 따른 응답을 하면, 상기 갱신파키지서버20은 723단계에서 상기 갱신파키지를 해당 수신시스템30에 다운로드한다. 상기 갱신파키지를 다운로드하는 상기 갱신파키지서버20은 725단계에서 수신시스템30에서 다운로드 완료 데이터를 수신하게 되며, 그러면 727단계에서 전송 종료 정보를 상기 수신시스템30에 전송하며, 729단계에서 응답 메시지 수신시 상기 갱신파키지의 다운로드 절차를 종료한다.
- [0121] 상기한 바와 같이 갱신파키지서버20은 상기 갱신파키지처리기10에서 전달하는 갱신파키지를 다수의 수신시스템30들에 통지하며, 상기 통지에 의해 응답하는 수신시스템30들에게 상기 갱신파키지를 다운로드한다. 이때 상기 수신시스템30은 상기 갱신파키지서버20에서 다운로드되는 갱신파키지를 제1메모리250에 인스톨하는 과정(installing) 및 갱신타기시 상기 제1메모리250에 인스톨된 갱신타기데이터를 이용하여 제2버전 정보를 생성하며, 이를 제2메모리260에 저장하여 운용하는 과정(translating)을 수행하게 된다.
- [0122] 도 22는 본 발명의 실시예에 따라 상기 수신시스템30이 상기 갱신파키지를 다운로드하여 인스톨하는 절차 및 상기 인스톨된 갱신파키지를 이용하여 제2버전 프로그램을 생성한 후 이를 이용하여 시스템을 운용하는 절차 등을 도시하는 흐름도이다.
- [0123] 상기 도 22를 참조하면, 상기 수신시스템30은 801단계에서 이미 상기 제1버전 프로그램을 제1메모리250에 저장하고 있는 상태이다. 이때 상기 제1버전은 시스템 제조시 상기 제1메모리250에 인스톨될 수 있으며, 또한 시스템 제조 후 기준이 되는 새로운 버전을 다운로드하여 이를 기준 버전의 프로그램으로 설정할 수도 있다. 상기과 같은 상태에서 갱신파키지의 다운로드를 통지받으면, 상기 수신시스템30은 상기 도 21과 같은 절차를 수행하면서 상기 갱신파키지서버20으로부터 갱신파키지를 다운로드하게 된다. 따라서 상기 수신시스템30은 803단계 - 807단계를 수행하면서 상기 갱신파키지서버20에서 다운로드되는 갱신파키지를 수신하여 일시 저장한다. 그리고 상기 일시저장된 상기 갱신파키지는 상기 제1메모리250에 인스톨하여야하는데, 이때 상기 인스톨 동작은 상기 갱신파키지를 다운로드하는 즉시 수행할 수 있으며, 또한 수신시스템30의 정상적인 동작모드를 수행한 후에 인스톨할 수도 있다. 따라서 상기 갱신파키지 다운로드 후 인스톨 명령이 발생되지 않으면, 상기 수신시스템30은 809단계에서 이를 감지하고 811단계에서 수신시스템30의 정상 동작모드로 되돌아간다. 상기과 같은 상태에서 정상모드를 수행한 후, 인스톨명령이 발생되면 상기 수신시스템30은 상기 일시 저장중인 갱신파키지를 상기 제1메모리250에 인스톨하는 동작을 수행하게 된다.
- [0124] 따라서 상기 809단계에서 인스톨 명령을 감지하면, 상기 수신시스템30은 813단계에서 상기 제1메모리250에 상기 다운로드된 갱신파키지를 인스톨시킨다. 이때 상기 제1메모리250은 비휘발성 메모리가 될 수 있으며, 버전 갱신시 기준이 되는 제1버전을 저장하는 영역과, 복수의 갱신파키지들을 저장하는 영역들로 구성된다. 즉, 상기 제1메모리250은 상기 도 9, 도10a-도 10b와 같이 버전들을 저장하는 제1 및 제2영역들로 할당할 수 있다. 따라서 상기 갱신파키지 다운로드시 상기 수신시스템30은 813단계에서 해당하는 갱신파키지를 저장하는 저장영역에 상기 다운로드된 갱신파키지를 인스톨시킨다. 이때 상기 갱신파키지는 이전의 갱신파키지와 독립적으로 저장되므로, 상기 제1메모리250은 상기 제2버전의 갱신파키지들이 생성된 순서대로 저장되어 있으며, 따라서 버전의 갱신 히스토리들을 모두 저장하고 있는 상태가 된다.
- [0125] 상기 갱신파키지를 상기 제1메모리250에 인스톨한 후, 상기 수신시스템30은 시스템 초기화(system reboot) 동작을 수행할 것인지를 표시한다. 이때 시스템 초기화 명령이 발생되지 않으면, 상기 수신시스템30은 815단계에서 이를 감지하고 817단계에서 정상 동작모드를 수행한다. 이런 경우 상기 동작모드는 현재 다운로드된 갱신파키지에 의한 버전 갱신이 수행되지 않은 상태이므로, 이전 버전으로 동작모드를 수행하게 된다.
- [0126] 그러나 상기 815단계에서 시스템 초기화 명령이 발생되면, 상기 수신시스템30은 821단계에서 시스템 초기화 동작을 수행하며, 상기 초기화 동작 수행 후 수신된 갱신파키지로부터 제2버전 프로그램을 생성하기 위해 823단계

에서 트랜스레이터240을 구동한다. 이때 상기 트랜스레이터240은 상기 제1메모리250에 인스톨된 갱신패키지를 상기 제1버전에 머지시켜 제2버전의 프로그램을 생성하여 제2메모리260에 저장하며, 이후 상기 수신시스템30은 상기 제2메모리260에 저장된 제2버전의 프로그램을 토대로 동작모드를 제어하도록 한다. 따라서 상기 수신시스템30은 825단계에서 상기 갱신패키지의 인스톨데이터를 검사하여 상기 갱신패키지의 상태를 확인한다. 이때 상기 갱신패키지의 상태에 이상이 있으면, 상기 수신시스템30은 833단계에서 갱신 버전 전의 버전을 상기 제2메모리260에 저장하여 시스템의 동작을 제어한다. 그러나 상기 갱신패키지의 상태가 정상 상태이면, 상기 수신시스템30은 825단계에서 이를 감지하고, 827단계에서 상기 제1메모리250에 인스톨된 해당하는 갱신패키지를 로드하고, 829단계에서 상기 제1메모리250에 저장하고 있는 제1버전에 상기 로드한 갱신패키지를 조합하여 제2버전의 프로그램을 생성하며, 이를 상기 제2메모리260에 저장한다. 이후 상기 수신시스템30은 831단계에서 상기 제2메모리260에 저장된 갱신버전의 프로그램을 이용하여 시스템의 동작을 제어한다.

[0127] 상기한 바와 같이 수신시스템30은 상기 갱신패키지를 다운로드하면, 인스톨과정 및 갱신 과정을 수행한다. 도 23은 본 발명의 실시예에 따라 상기 수신시스템30에서 상기 다운로드되는 갱신패키지를 상기 제1메모리250에 인스톨하는 절차를 도시하는 흐름도이며, 도 24는 본 발명의 실시예에 따라 상기 수신시스템30에서 상기 제1메모리250에 인스톨된 갱신패키지를 제1버전의 프로그램에 갱신하여 상기 제2메모리260에 저장한 후, 시스템을 운용하는 절차를 도시하는 흐름도이다.

[0128] 상기 도 23을 참조하면, 다운로드 필요시 상기 수신시스템30은 841단계에서 다운로더(down loader)를 실행시키며, 843단계에서 네트워크를 통해 상기 갱신패키지 서버20에서 전송되는 갱신패키지를 다운로드한다. 이때 갱신패키지의 다운로드는 수신시스템30에 따라 달라진다. 즉, 상기 수신시스템30이 이동통신 단말기이면, 상기 수신시스템30은 설정되는 무선링크(예를들면 CDMA, UMTS, GSM, GPRS 등)를 통해 상기 갱신패키지를 다운로드할 수 있으며, 인터넷을 통해 연결되는 단말기이면 또는 무선 인터넷(wibro, wifi, wimax, 무선 랜 등)을 통해 갱신패키지를 다운로드할 수 있다. 또한 USB와 같은 유선통신 링크 또는 블루투스 등과 같은 근거리 통신 링크를 통해 시스템에 연결되는 단말기인 경우에는 해당 유선통신 방법 또는 근거리 무선통신 방법에 따라 상기 갱신패키지를 다운로드할 수 있다. 이때 상기 다운로드는 갱신패키지 전체가 다운로드될 때 까지 계속되며, 다운로드 중에 에러가 발생되면 상기 수신시스템30은 845단계에서 이를 감지하고 847단계 및 849단계를 수행하면서 상기 갱신패키지의 재 다운로드를 수행한다.

[0129] 이때 상기 845단계에서 정상적으로 다운로드를 완료한 경우, 상기 수신시스템30은 851단계에서 인스톨러(installer)230을 로드하여 상기 갱신패키지를 상기 제1메모리250에 인스톨할 준비를 한다. 상기 갱신패키지를 상기 제1메모리250에 인스톨하는 절차를 살펴보면, 상기 수신시스템30은 853단계 - 857단계에서 상기 갱신패키지의 히스토리 데이터를 추출하고, 상기 추출된 히스토리 데이터를 확인한 후, 제1메모리250의 해당 갱신패키지 저장영역 내의 히스토리 테이블에 저장한다.

[0130] 이후 상기 수신시스템30은 상기 갱신패키지 내에 맵데이터가 포함되어 있는가 검사한다. 이때 상기 갱신패키지 내에 맵 데이터가 포함되어 있는 경우, 상기 수신시스템30은 859단계에서 이를 감지하고 875단계 및 877단계에서 상기 갱신패키지 내에서 맵 데이터를 추출하여 상기 제1메모리250의 해당 갱신패키지 저장영역 내에 맵 데이터를 저장하고, 879단계에서 상기 갱신패키지 저장영역 내의 갱신데이터 저장영역에 갱신데이터를 저장한다. 따라서 881단계가 되면, 상기 수신시스템30은 수신된 갱신패키지에서 히스토리데이터, 맵 데이터 및 갱신데이터를 각각 추출한 후 상기 제1메모리250의 해당하는 갱신패키지 저장영역 내에 히스토리데이터, 맵데이터 및 갱신데이터를 인스톨하여 저장한 상태가 된다.

[0131] 또한 상기 갱신패키지 내에 맵데이터가 포함되어 있지 않은 경우, 상기 수신시스템30은 859단계에서 이를 감지하고 861단계 및 863단계에서 압축해제기270을 구동하고 상기 갱신패키지에서 갱신데이터를 추출하여 압축을 해제한다. 이후 상기 수신시스템30은 865단계 - 869단계를 수행하면서 상기 압축해제된 갱신 데이터를 파싱한 후, 이를 상기 제1메모리250에 저장하고 있는 제1버전정보와 비교하여 차 데이터를 구하고, 이를 토대로 상기 갱신데이터를 상기 제1버전정보에 맵핑하기 위한 맵 데이터를 생성한다. 이후 상기 수신시스템30은 871단계 및 873단계에서 상기 제1메모리250에 할당된 갱신패키지의 저장영역에 상기 생성된 맵 데이터 및 상기 갱신데이터를 각각 저장한다.

[0132] 상기 도 23에 도시된 바와 같이, 상기 수신시스템30은 갱신패키지의 다운로드가 통지되면 상기 갱신패키지를 다운로드하고, 상기 다운로드된 갱신패키지를 분석하여 상기 제1메모리250의 대응되는 영역에 각각 히스토리데이터, 맵 데이터 및 갱신데이터를 인스톨한다. 이때 상기 맵데이터는 상기 갱신패키지에 포함될 수 있으며, 포함되지 않을 수도 있다. 따라서 상기 갱신패키지에 상기 맵데이터가 포함되지 않은 경우, 상기 수신시스템30은 상

기 제1메모리250에 저장하고 기준버전정보인 제1버전정보에 상기 다운로드된 갱신데이터를 비교하여 맵 데이터를 생성하고, 상기 생성된 데이터를 갱신패키지 저장영역에 인스톨 저장한다.

[0133] 도 24는 상기 수신시스템30에서 상기 도 23과 같은 절차로 다운로드하는 갱신패키지를 제2메모리 상에서 갱신하는 절차를 도시하는 흐름도이다.

[0134] 상기 도 24를 참조하면, 상기 갱신패키지를 이용하여 수신시스템30의 버전을 갱신하는 절차는 시스템 전원은 또는 사용자의 갱신 요구에 의해 수행된다. 즉, 본 발명의 실시예에 따른 수신시스템30은 시스템 초기화시 하드웨어 등의 초기화를 수행한 후, 상기 제1메모리250에 저장된 제1버전 프로그램 및 가장 최근에 갱신된 갱신패키지(또는 사용자에 의해 설정된 갱신패키지)를 조합하여 갱신된 제2버전 프로그램을 상기 제2메모리260에 저장하고, 이를 이용하여 시스템의 동작을 제어한다. 여기서 상기 제2메모리260에 저장되는 정보는 프로그램으로써 비휘발성 메모리에 인스톨될 수 있는 모든 정보들을 포함한다.

[0135] 따라서 시스템 초기화 동작이 수행되면, 상기 수신시스템30은 881단계 - 883단계를 수행하면서 시스템 초기화를 수행한 후 상기 제1메모리250에서 제1버전 프로그램 및 갱신패키지를 로딩하기 위한 로더를 실행한다. 이후 상기 수신시스템30은 884단계에서 상기 제1메모리250 내의 갱신패키지 저장영역들을 분석하여 상기 갱신패키지가 하나 이상 저장되어 있는지를 검사한다. 이때 상기 갱신패키지가 저장되어 있지 않은 경우, 상기 수신시스템30은 884단계에서 이를 감지하고 885단계에서 상기 갱신 절차를 수행하기 위한 트랜스레이터240을 로딩하며, 886단계에서 다운받은 갱신패키지가 수신측에 적합한지 여부와 바이러스 체크 등의 시큐리티를 검사한다. 이후 상기 수신시스템30은 887단계에서 상기 제1메모리250에 저장되어 있는 제1버전 프로그램이 압축된 상태인가 검사하며, 압축된 상태이면 888단계에서 압축해제기270을 구동하여 상기 제1버전의 압축을 해제한 후, 889단계에서 상기 제1버전을 상기 제2메모리에 저장하며, 890단계에서 상기 제2메모리260에 저장된 상기 제1버전을 이용하여 시스템을 운용한다. 그러나 상기 887단계에서 상기 제1버전이 압축되지 않은 것으로 판정되면, 상기 수신시스템30은 889단계 및 890단계를 수행하면서 상기 제1버전을 상기 제2메모리260에 저장하고, 이를 이용하여 시스템을 운용한다.

[0136] 그러나 상기 884단계에서 하나 이상의 갱신패키지가 상기 제1메모리250에 저장되어 있으면, 상기 수신시스템30은 891단계에서 트랜스레이터240을 로딩한 후, 892단계에서 가장 최근에 갱신된 갱신패키지를 로드한다. 이때 상기 제1메모리250에 인스톨된 갱신패키지는 히스토리 데이터, 맵 데이터 및 갱신 데이터, 또는 히스토리 데이터 및 맵 데이터 또는 히스토리 데이터 및 갱신데이터 등으로 구성될 수 있다. 이후 상기 수신시스템30은 893단계에서 압축해제기270을 구동하여 상기 로드된 갱신패키지(여기서 상기 갱신데이터만 압축되어 있을 수도 있음)의 압축을 해제하며, 894단계에서 시큐리티를 검사한다. 이후 상기 수신시스템30은 895단계에서 상기 제1메모리250에 저장되어 있는 제1버전이 압축된 상태인가 검사하며, 압축된 상태이면 896단계에서 압축해제기를 구동하여 상기 제1버전의 압축을 해제한 후, 897단계에서 상기 제1버전에 상기 갱신패키지를 결합하여 갱신된 제2버전 프로그램을 상기 제2메모리에 저장하며, 890단계에서 상기 제2메모리260에 상기 갱신된 제2버전 프로그램을 이용하여 시스템을 운용한다. 이때 897단계에서 상기 갱신패키지의 갱신데이터를 상기 제1버전에 결합하여 갱신할 때 맵데이터를 이용하여 상기 갱신데이터를 상기 제1버전에 맵핑한다. 또한 상기 895단계에서 상기 제1버전이 압축되지 않은 것으로 판정되면, 상기 수신시스템30은 897단계 및 890단계를 수행하면서 상기 제1버전에 갱신패키지를 조합하여 갱신 버전의 프로그램을 생성한 후 이를 상기 제2메모리에 저장하고, 이를 이용하여 시스템을 운용한다.

[0137] 도 25a - 도 25d는 상기 도 24와 같이 상기 제1메모리250에 인스톨된 갱신패키지를 기준버전 프로그램에 결합하여 갱신된 제2버전 프로그램을 생성한 후 제2메모리260에 저장하는 절차를 상세하게 도시하는 흐름도이다.

[0138] 상기 도 25a - 도 25d를 참조하면, 상기 수신시스템30은 901단계 - 905단계를 수행하면서 시스템 초기화를 수행한 후 버전을 갱신하기 위하여 로더를 실행시키며, 907단계 및 909단계를 수행하면서 갱신패키지가 유효한가 결정한 후, 상기 제1메모리250에 적어도 하나의 갱신패키지가 저장되어 있는가 검사한다. 이때 상기 제1메모리250에 유효한 갱신패키지가 하나도 저장되어 있지 않으면, 상기 수신시스템30은 911단계로 진행하여 트랜스레이터240을 로드한 후, 도 25b의 921단계 - 929단계를 수행하면서 상기 제1메모리250에 저장하고 있는 제1버전을 트랜스레이팅하여 제2메모리260에 저장하고, 이를 이용하여 시스템을 운용한다. 이때 상기 제1버전 V1이 압축된 상태이면, 상기 수신시스템30은 922단계에서 이를 감지하고 923단계 - 927단계를 수행하면서 제1버전 V1의 마지막 데이터까지 트랜스레이팅을 하여 제2메모리260에 저장한다. 그리고 상기 제1버전정보 V1이 압축된 데이터가 아니면, 상기 수신시스템30은 926단계 및 927단계를 반복 수행하면서 제1버전 V1의 마지막 데이터까지 트랜스레이팅하여 상기 제2메모리260에 저장한다. 상기과 같이 제1버전 V1을 저장한 후, 상기 수신시스템30은 928단계에

서 상기 제2메모리260에 저장된 버전(여기서는 시스템의 전체 소프트웨어 정보가 될 수 있음)를 검증한 후, 929 단계에서 이를 이용하여 시스템의 전체적인 동작을 수행시킨다.

[0139] 그러나 상기 909단계에서 적어도 하나의 갱신파키지가 제1메모리250에 저장되어 있음을 감지하면, 상기 수신시스템30은 913단계에서 상기 갱신파키지들이 히스토리 데이터들을 구비하고 있는가 검사한다. 이때 상기 히스토리 데이터를 포함하고 있는 갱신파키지이면, 상기 수신시스템30은 915단계에서 상기 히스토리데이터 내의 실패 플래그(fail flag)를 검사한다. 여기서 상기 실패플래그는 해당 갱신파키지의 로드 실패가 발생한 적이 있음을 표시하는 플래그로써, 상기 실패플래그가 세트(fail flag = true) 되었다는 의미는 해당 갱신파키지에 이상이 있었음을 나타내는 것이다. 따라서 상기 수신시스템30은 917단계에서 상기 히스토리데이터의 실패플래그가 세트되었다는가 검사한 후, 실패 플래그가 세트되지 않은 상태이면 도 25c의 931단계 - 946단계를 수행하면서 시스템 갱신 동작을 수행하며, 상기 실패플래그가 세트된 상태이면 도 25d의 951단계-969단계를 수행하면서 시스템 갱신동작을 수행한다.

[0140] 먼저 상기 갱신파키지의 히스토리데이터가 정상인 경우의 동작을 도 25c를 참조하여 살펴보면, 상기 수신시스템30은 931단계 - 937단계에서 상기 제1메모리250에 저장하고 있는 가장 최근의 갱신파키지의 히스토리데이터를 탐색하여 실패 플래그를 확인하고, 정상이면 맵데이터 및 갱신 데이터를 로드하며, 트랜스레이터를 로드한 후 시큐리티를 검사하고, 상기 갱신테이터가 압축된 경우에는 상기 갱신테이터의 압축을 해제한다. 이후 상기 제1버전 V1이 압축된 경우, 상기 수신시스템30은 939단계 및 941단계를 수행하면서 상기 제1버전 V1 프로그램의 압축을 해제하고, 942단계에서 상기 제1버전 V1에 상기 갱신파키지의 갱신테이터를 매핑하여 버전정보를 갱신하며, 상기 갱신되는 버전정보를 상기 제2메모리260에 저장한다. 상기과 같은 동작은 943단계 및 944단계를 반복 수행하면서 상기 갱신테이터의 마지막 데이터까지 갱신하는 동작을 수행한다. 그리고 상기 938단계에서 상기 제1버전 V1이 압축되지 않은 경우, 상기 수신시스템30은 938단계에서 이를 감지하고 940단계, 943단계 및 944단계를 수행하면서 상기 제1버전 V1에 상기 갱신파키지의 갱신테이터를 매핑시켜 갱신 버전 V2의 프로그램을 생성한 후 이를 상기 제2메모리260에 저장한다. 그리고 상기과 같이 제2메모리260에 갱신버전의 프로그램을 저장한 후, 상기 수신시스템30은 945단계에서 이를 검증한 후, 946단계에서 상기 제2메모리260에 저장된 갱신 버전정보를 이용하여 시스템을 구동한다.

[0141] 또한 상기 917단계에서 갱신파키지에 실패플래그가 세트된 경우, 상기 수신시스템30은 951단계에서 모든 갱신파키지들의 실패플래그가 세트되었는가 검사한다. 이때 모든 갱신파키지들의 실패플래그가 세트된 상태이면, 상기 수신시스템30은 952단계에서 트랜스레이터를 로드한 후, 상기 921단계 이후의 동작 절차를 수행하면서 상기 제1버전을 제2메모리260에 저장하고, 이를 이용하여 시스템의 구동을 제어한다. 즉, 인스톨된 모든 갱신파키지들에 이상이 발생한 경우, 상기 수신시스템30은 상기 제1메모리250에 저장하고 있는 제1버전을 로딩하여 제2메모리260에 저장하고, 이를 이용하여 시스템의 동작을 제어한다. 이때 상기 제1버전은 시스템을 제조시 상기 제1메모리250에 로딩되는 프로그램이 될 수 있다.

[0142] 그러나 상기 951단계에서 상기 모든 갱신파키지들의 실패 플래그가 세트된 상태가 아닌 경우, 상기 수신시스템30은 953단계 및 954단계에서 상기 실패플래그가 세트된 갱신파키지들을 검사한 후 사용 가능한 갱신파키지들의 리스트를 만들어 사용자에게 표시한다. 이후 사용자가 상기 리스트 중에서 원하는 갱신파키지를 선택하면, 상기 수신시스템30은 955단계에서 이를 감지하고 956단계 - 960단계를 수행하면서 상기 선택된 갱신파키지의 맵 데이터 및 갱신테이터를 로드하고, 트랜스레이터를 로드하고 시큐리티를 검사한 후, 상기 갱신테이터가 압축되었으면 상기 갱신테이터의 압축을 해제한다. 이후 961단계 - 969단계를 단계를 수행하면서 상기 제1메모리250에서 제1버전을 로드한 후 상기 갱신테이터를 매핑시켜 제2버전을 생성하여 제2메모리260에 저장한 후, 이를 이용하여 시스템의 구동을 제어한다. 상기 961단계 - 969단계의 동작은 상기 938단계 - 946단계와 동일한 방법으로 수행될 수 있다.

[0143] 상기한 바와 같이 본 발명의 실시예에서는 갱신파키지를 생성하는 송신측이 기준 버전과 갱신 버전의 차이를 비교하고, 비교 결과에 따른 갱신파키지를 생성하여 전송하며, 갱신파키지를 다운로드하는 수신측은 상기 다운로드되는 갱신파키지를 제1메모리250에 인스톨하고, 이후 상기 제1메모리250의 갱신파키지를 상기 제1버전의 프로그램에 매핑시켜 제2버전의 프로그램을 생성하여 제2메모리260에 저장한 후, 이를 이용하여 시스템의 구동을 제어한다.

[0144] 먼저 상기 갱신파키지처리기10에서 갱신파키지를 생성하는 방법(upgrade module generation; delta generation)을 살펴보면, 먼저 동영상신호를 부호화하는 방법과 같은 방법으로 압축되지 않은 초기 버전(제1버전 V1)과 압축되지 않은 현재 버전(제2버전 V2) 간의 차이를 부호화하기 위하여 코드 제1압축기(code

compressor)160을 용한다. 이때 상기 제1압축기160은 최초 버전 V1과 현재 버전 V2 간의 차를 부호화하여 압축한다. (domain : V1 (previous frame), range: V2 (current frame))

[0145] 상기 갱신패키지 생성 모듈의 특성은 고 압축률(high compression ratio : 평균 99%가 넘음), 빠른 갱신 시간(very fast updating time : file loading 시간보다 빠름) 및 생성 시간(generation time)이 또한 매우 빠른 특성을 가진다.

[0146] 본 발명의 실시예에 따른 갱신 패키지 생성의 특징은 다음과 같다.

[0147] 먼저 제1버전 및 제2버전을 수신하며, 두 버전들을 비교하여 그 차를 포함하는 비교 데이터를 생성한다. 여기서 상기 제1버전은 기준 버전으로써, 시스템 제조시 메모리에 인스톨되는 프로그램 또는 이후 결정된 기준 프로그램이 될 수 있다. 그리고 상기 제2버전은 상기 제1버전의 갱신버전으로써, 시스템 제조 후 시스템에 다운로드되는 버전들을 의미한다. 여기서 상기 제2버전은 복수 개로 발생될 수 있으며, 가장 최근에 생성되는 제2버전이 가장 최근에 갱신된 버전이 될 수 있다.

[0148] 상기 두 버전들을 비교할 때, 상기 버전들을 각각 압축(compress_1)한 후 비교할 수 있으며, 또한 압축하지 않은 버전들을 압축하지 않은 상태에서 비교할 수도 있다. 상기 버전들을 압축하여 비교하는 경우, 상기 버전들이 정상적으로 압축되었는지 검증하기 위하여 상기 압축 버전들의 압축을 해제한 후 압축 해제된 버전들을 각각 압축 전의 버전들과 비교하여 압축이 정상적으로 이루어졌는지 검증할 수 있다.

[0149] 두 상기 비교 데이터를 이용하여 인스톨 데이터를 생성한다.

[0150] 상기 인스톨데이터는 상기 제1버전에 갱신패키지를 맵핑할 때 사용하기 위한 인스톨 데이터를 의미한다. 상기 인스톨 데이터는 적어도 히스토리 데이터를 포함하여야 한다. 상기 히스토리 데이터는 갱신패키지의 버전식별정보, 결합되어야 할 버전 식별정보등을 포함하고 있으며, 로딩 실패 여부를 표시하는 플래그를 구비할 수 있다. 또한 상기 인스톨데이터는 상기 히스토리 데이터 이외에 맵데이터를 포함할 수 있다. 상기 맵데이터는 갱신패키지를 상기 제1버전에 맵핑하기 위한 데이터로써, 복사, 변경 및 이동 등의 명령어들과 및 상기 제1버전과 블록 맵핑을 위한 인덱스 정보들로 구분될 수 있다. 여기서 상기 인스톨데이터가 상기 맵데이터를 포함하지 않는 경우에는 수신측에서 상기 맵데이터를 생성할 수 있다.

[0151] 세 번째로 상기 비교데이터를 이용하여 갱신패키지를 생성하며, 상기 갱신패키지에 상기 인스톨데이터를 합성하여 갱신패키지를 생성한다.

[0152] 상기 갱신패키지는 상기 제1버전과 상기 제2버전을 블록 단위로 비교하고, 비교 결과 블록 데이터가 서로 다르거나 또는 삽입된 블록 데이터인 경우 해당하는 블록 데이터를 갱신패키지로 설정한다. 이때 상기 갱신패키지에 대응되는 맵 데이터는 변경(modify) M 데이터가 될 수 있다. 이때 상기 갱신패키지는 또한 상기 갱신패키지에 상기 인스톨데이터를 합성하여 갱신패키지로 생성한다. 이때 상기 인스톨 데이터에는 맵 데이터가 포함되지 않을 수 있다. 이런 경우 상기 수신측에서 생성하여 인스톨시킨다.

[0153] 또한 상기 갱신패키지 또는 갱신패키지는 압축시켜 다운로드시킬 수 있다. 이런 경우 상기 갱신패키지(또는 갱신패키지) 압축시키며, 상기 압축된 갱신패키지(또는 갱신패키지)의 압축을 해제하고, 상기 압축전의 갱신패키지(또는 갱신패키지)와 상기 압축해제한 갱신패키지(또는 갱신패키지)비교 검토하여 압축이 정상적으로 이루어졌는지 검증한다.

[0154] 상기와 같이 생성된 갱신패키지는 갱신패키지 서버20에 전달되며, 상기 갱신패키지 서버는 상기 전달되는 갱신패키지를 수신시스템들에 통지하고, 상기 통지에 응답하는 수신시스템들에게 상기 갱신패키지를 다운로드한다.

[0155] 본 발명의 실시예에 따른 수신시스템의 특징은 다음과 같다.

[0156] 먼저 제1메모리250에 제1 버전의 프로그램을 저장하며, 이후 다운로드되는 적어도 하나의 갱신 패키지를 저장한다. 여기서 제1버전은 최초 버전의 프로그램 또는 설정된 기준 프로그램을 의미한다. 상기 갱신 패키지는 갱신패키지 및 맵 데이터로 이루어지며, 갱신패키지는 이전 버전 패키지와 현 버전 패키지의 차이 데이터, 또는 제2버전 V2의 데이터이고, 맵 데이터는 명령어(여기서는 copy, shift, modify) 및 명령어가 수행될 어드레스(블록 인덱스)를 포함한다. 상기 갱신패키지는 상기 제1메모리250에 다수개(본 발명의 실시예에서는 최대 6개)를 저장할 수 있으며, 초과되는 경우 오버라이트(overwrite)한다. 상기 맵(map) 데이터는 서버에서 갱신 패키지에 포함시켜 전송할 수 있으며, 휴대단말에서 생성하는 방법을 사용할 수도 있다.

[0157] 두 번째로 시스템 기동시 상기 제1메모리250의 제1버전의 패키지 및 갱신 패키지를 제2메모리260에 로딩하여 프

로그래를 갱신할 수 있다.

[0158] 여기서 제1메모리250은 비휘발성 메모리이고, 제2메모리260은 휘발성 메모리를 사용할 수 있다. 상기 변경 패키지는 제1버전의 패키지를 제2메모리260에 copy하거나, 제1버전의 패키지를 상기 갱신 패키지에 의해 변경하여 생성한다. 이때 후자의 변경 패키지의 생성은 먼저 제1버전의 패키지를 제2메모리260에 복사하고, 상기 갱신패키지에 의해 상기 제1버전의 패키지를 갱신하여 변경 패키지를 생성한다. 이때 변경방법은 상기한 바와 같이 copy, shift, modify 등이 될 수 있다. 변경의 전반적인 동작은 로더(loader)에 의해 관리되며, 로더는 재부팅(re-boot)이 발생될 때 어셈블러 및 압축해제기의 구동, UI 제공등의 기능을 수행한다. 로더에 포함된 맵 관리기(map manager)는 히스토리 관리 및 맵 생성기(map generator)를 실행할 지를 결정한다. 이때 map data가 있으면 map generator를 실행하지 않는다.

[0159] 세 번째로 상기 제2메모리260의 변경된 패키지를 이용하여 휴대단말기의 동작을 수행한다.

[0160] 그리고 상기 수신시스템의 사용자가 갱신 패키지를 선택하고, 선택된 갱신 패키지에 의해 시스템의 버전을 갱신할 수 있다. 즉, 수신시스템은 갱신 가능한 갱신패키지들의 리스트를 표시하고, 시스템 사용자에게 의해 선택된 갱신패키지를 상기 제1버전에 갱신하여 시스템을 운용할 수 있다. 또한 상기한 바와 같이 갱신패키지 다운로드시 갱신패키지들을 각각 할당된 특정 저장영역에 저장하고 상기 저장영역을 설정된 수로 구비하므로, 특정 갱신 패키지에 이상이 발생되더라도 다른 버전의 갱신패키지를 이용하여 갱신 버전의 프로그램을 생성할 수 있다.

발명의 효과

[0161] 상술한 바와 같이 시스템의 갱신 버전을 생성할 때 기준버전과 갱신버전의 차에 따른 갱신패키지를 생성할 수 있으며, 이로인해 갱신버전의 프로그램을 신속하게 생성할 수 있다. 또한 상기 갱신버전을 다운로드할 때 수신시스템은 제1메모리에 상기 기준버전의 프로그램 및 갱신버전의 갱신패키지를 독립적으로 인스톨하여 보관하며, 상기 보관된 기준버전 및 갱신버전을 제2메모리에서 머지하여 갱신된 프로그램을 생성함으로써 안정되게 시스템의 프로그램을 갱신할 수 있는 이점이 있다. 또한 다수개의 갱신패키지들을 보관하고 있으므로, 시스템의 갱신 히스토리를 유지할 수 있으며, 사용자는 필요시 원하는 버전의 갱신 프로그램으로 시스템 프로그램을 변경할 수 있는 이점이 있다. 또한 상기 제1버전 프로그램의 자체 갱신은 제1메모리에서 실행되지 않으므로 갱신 실패시 콘트롤하는 제어기능인 폴트 톨러런트 제어(fault tolerant control)가 없어도 폴트 톨러런트 제어(fault tolerant control)기능을 가진다. 즉, 본 발명은 갱신이 실패하더라도, 제1 버전 프로그램이 제1메모리에 유효하여 갱신에 실패하여도 언제든지 다시 할 수 있다. 갱신 절차를 수행하는 중에 어떤 폴트가 발생하여도 다시 실행할 수 있는 이점이 있다.

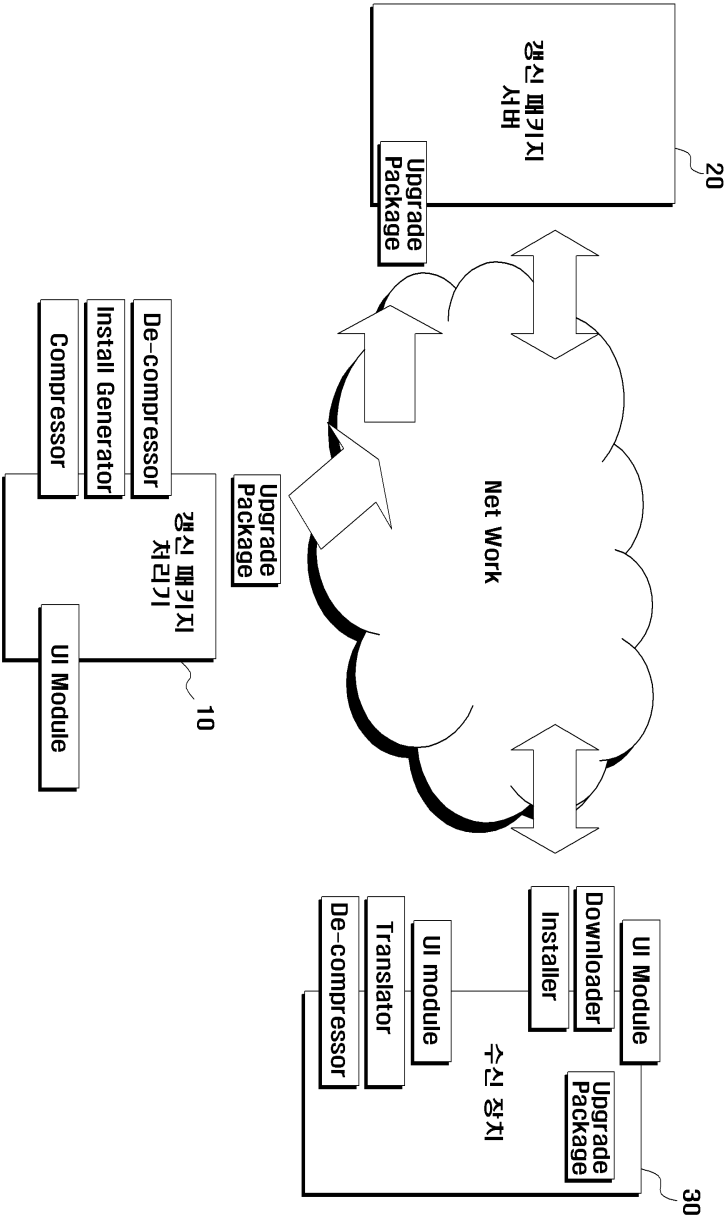
도면의 간단한 설명

- [0001] 도 1은 본 발명의 실시예에 따라 갱신패키지를 생성하여 전송하고, 상기 전송되는 갱신패키지를 다운로드하여 시스템 갱신 동작을 수행하는 시스템의 구성을 도시하는 도면
- [0002] 도 2는 본 발명의 실시예에 따른 갱신패키지 생성기와 주변 구성의 관계를 설명하기 위한 도면
- [0003] 도 3a 및 도 3b는 상기 갱신패키지처리기10에서 갱신패키지를 생성하는 방법을 설명하기 위한 도면
- [0004] 도 4는 도 2의 갱신패키지 생성기의 구성예를 도시하는 도면
- [0005] 도 5는 도 2의 갱신패키지 생성기의 다른 구성예를 도시하는 도면
- [0006] 도 6a는 도 4와 같은 갱신패키지생성기에서 발생하는 갱신패키지의 구성을 도시하는 도면이며, 도 6b는 도 5와 같은 갱신패키지생성기에서 발생하는 갱신패키지의 구성을 도시하는 도면
- [0007] 도 7은 도 2의 갱신패키지 생성기의 또 다른 구성예를 도시하는 도면
- [0008] 도 8은 본 발명의 실시예에 따른 수신시스템의 구성을 도시하는 도면
- [0009] 도 9는 본 발명의 실시예에 따른 수신시스템의 제1메모리의 구조를 도시하는 도면
- [0010] 도 10a는 제1메모리에 저장되는 갱신패키지들의 구조를 도시하는 도면이며, 도 10b는 도 10a와 같은 갱신패키지에서 히스토리데이터의 구조를 도시하는 도면

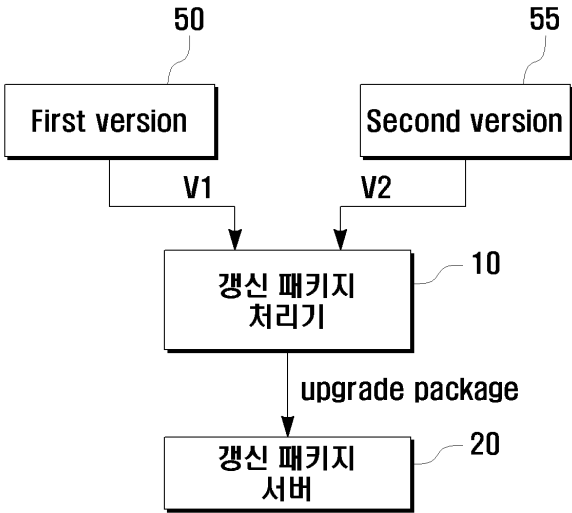
- [0011] 도 11a 및 도 11b는 상기 제1메모리 및 제2메모리의 구조를 도시하는 도면
- [0012] 도 12는 제2버전정보가 제1메모리에 저장되지 않은 경우의 동작을 설명하기 위한 도면
- [0013] 도 13은 제2버전정보가 제1메모리에 저장된 경우의 동작을 설명하기 위한 도면
- [0014] 도 14는 도 12 및 도 13과 같이 수신시스템에서 버전 갱신방법을 설명하기 위한 도면
- [0015] 도 15는 본 발명의 실시예에 따른 갱신패키지 생성 및 갱신 절차를 설명하는 흐름도
- [0016] 도 16a - 도 16c는 상기 갱신패키지를 생성하는 절차를 도시하는 흐름도
- [0017] 도 17은 본 발명의 실시예에 따라 갱신패키지처리기10에서 갱신패키지를 생성하는 절차를 도시하는 흐름도
- [0018] 도 18은 도 17에서 제1 및 제2버전정보를 압축 및 검증하는 절차를 도시하는 흐름도
- [0019] 도 19는 도 17에서 인스톨 데이터를 생성하는 절차를 도시하는 흐름도
- [0020] 도 20은 도 17에서 갱신패키지를 생성하는 절차를 도시하는 흐름도
- [0021] 도 21은 본 발명의 실시예에 따라 갱신패키지를 전달받는 갱신패키지서버가 갱신패키지를 상기 수신시스템들에 다운로드하는 절차를 도시하는 흐름도
- [0022] 도 22는 본 발명의 실시예에 따라 상기 수신시스템이 갱신패키지를 다운로드하여 인스톨하는 절차 및 상기 인스톨된 갱신패키지를 갱신하여 운용하는 절차 등을 도시하는 흐름도
- [0023] 도 23은 본 발명의 실시예에 따라 상기 수신시스템에서 다운로드되는 갱신패키지를 제1메모리에 인스톨하는 절차를 도시하는 흐름도
- [0024] 도 24는 본 발명의 실시예에 따라 수신시스템에서 제1메모리에 인스톨된 갱신패키지를 제1버전정보에 갱신하여 제2메모리에 저장한 후, 시스템을 운용하는 절차를 도시하는 흐름도
- [0025] 도 25a-도 25d는 도 24와 같이 제1메모리에 인스톨된 갱신패키지를 기준버전 정보에 갱신하여 제2메모리에 저장하는 절차를 상세하게 도시하는 흐름도

도면

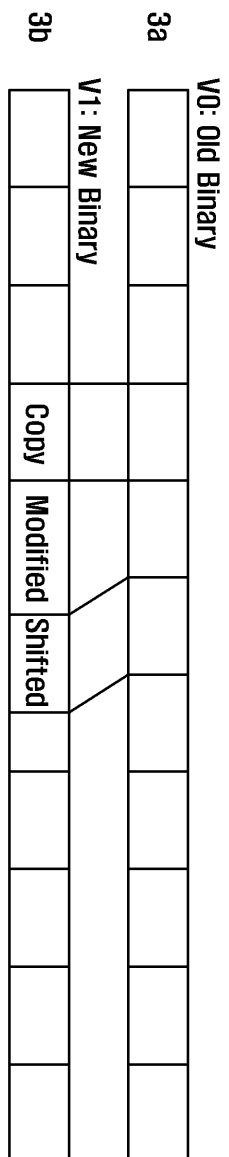
도면1



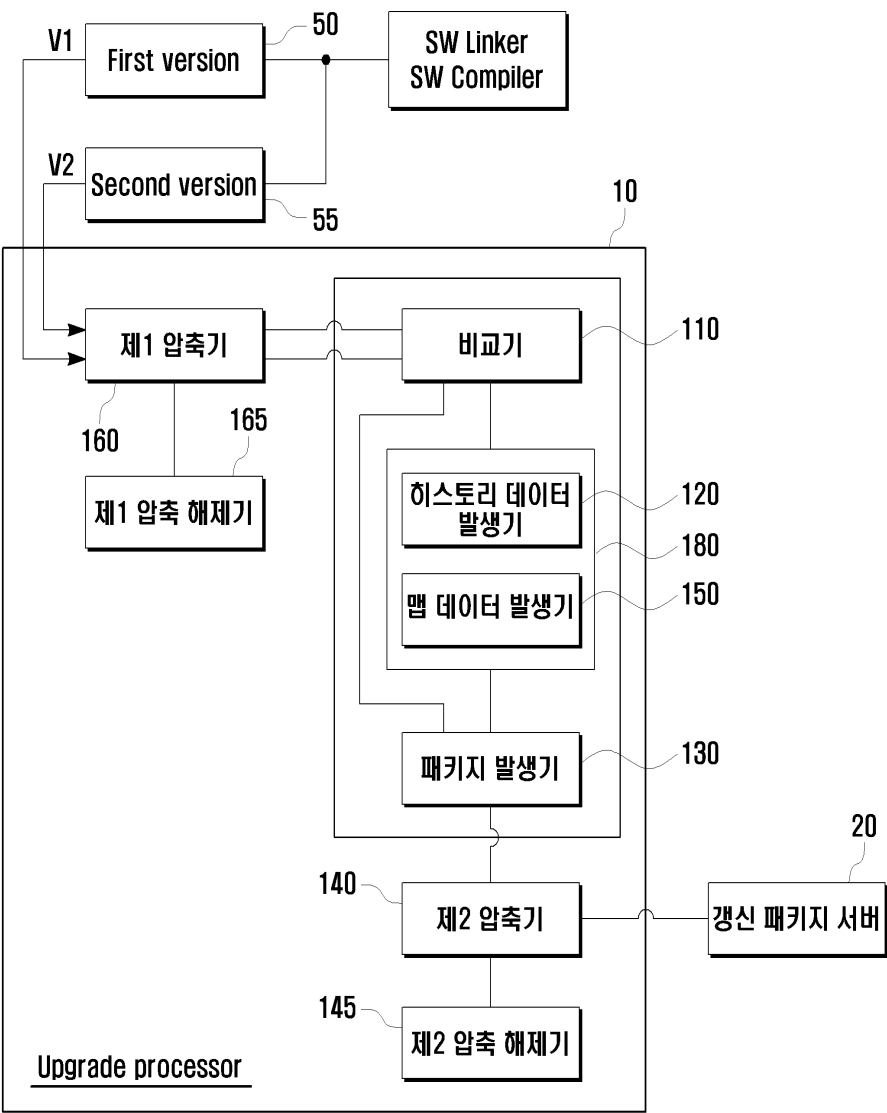
도면2



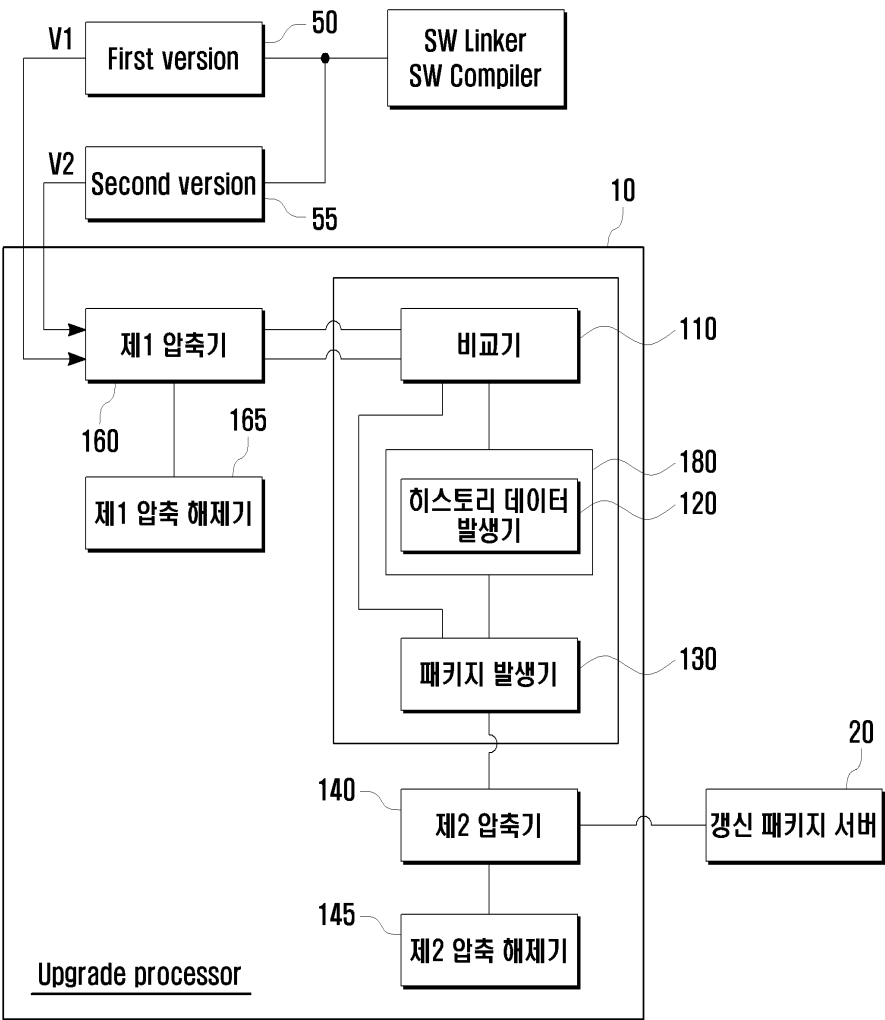
도면3



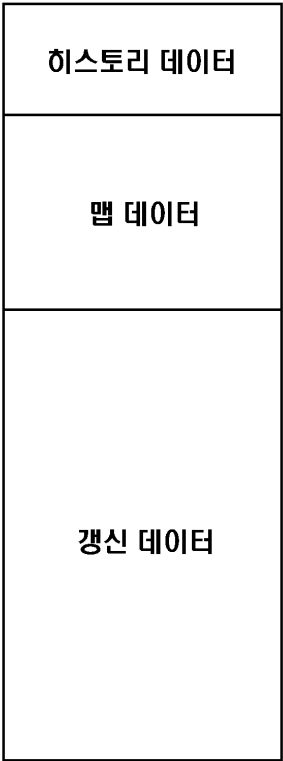
도면4



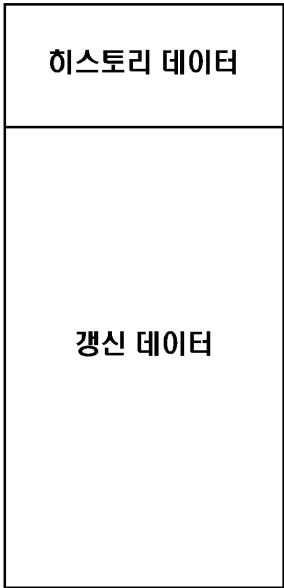
도면5



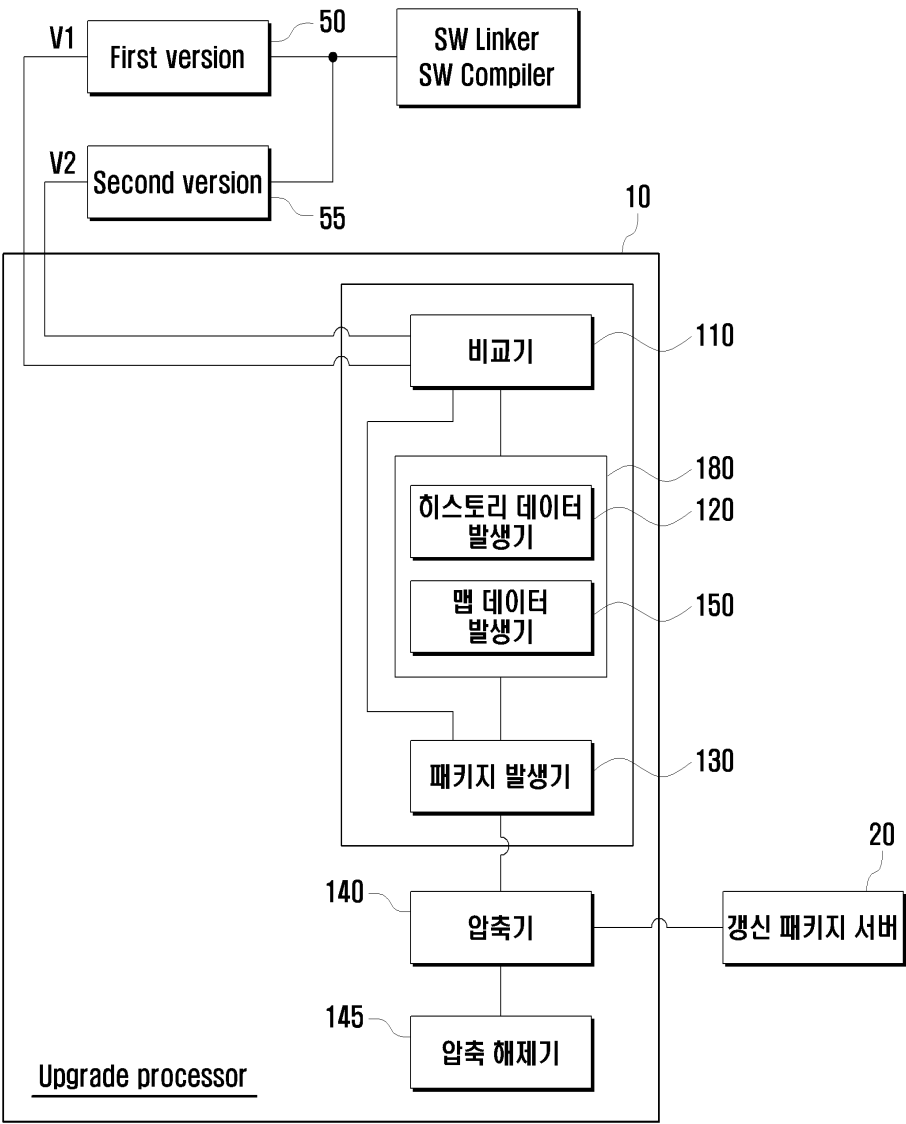
도면6a



도면6b

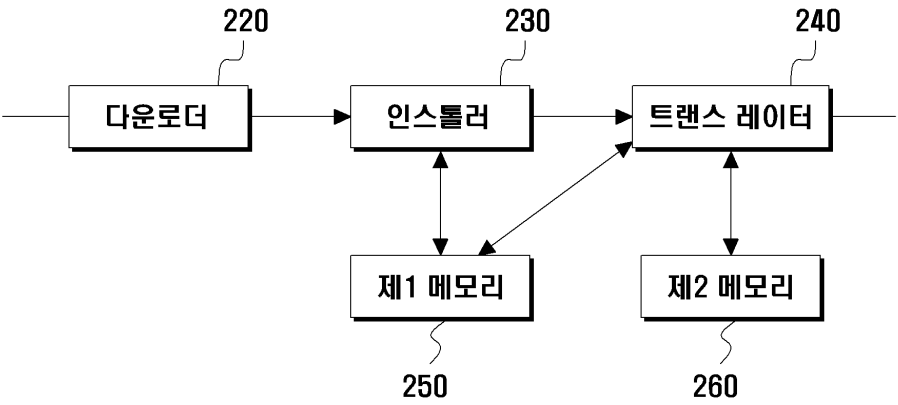


도면7

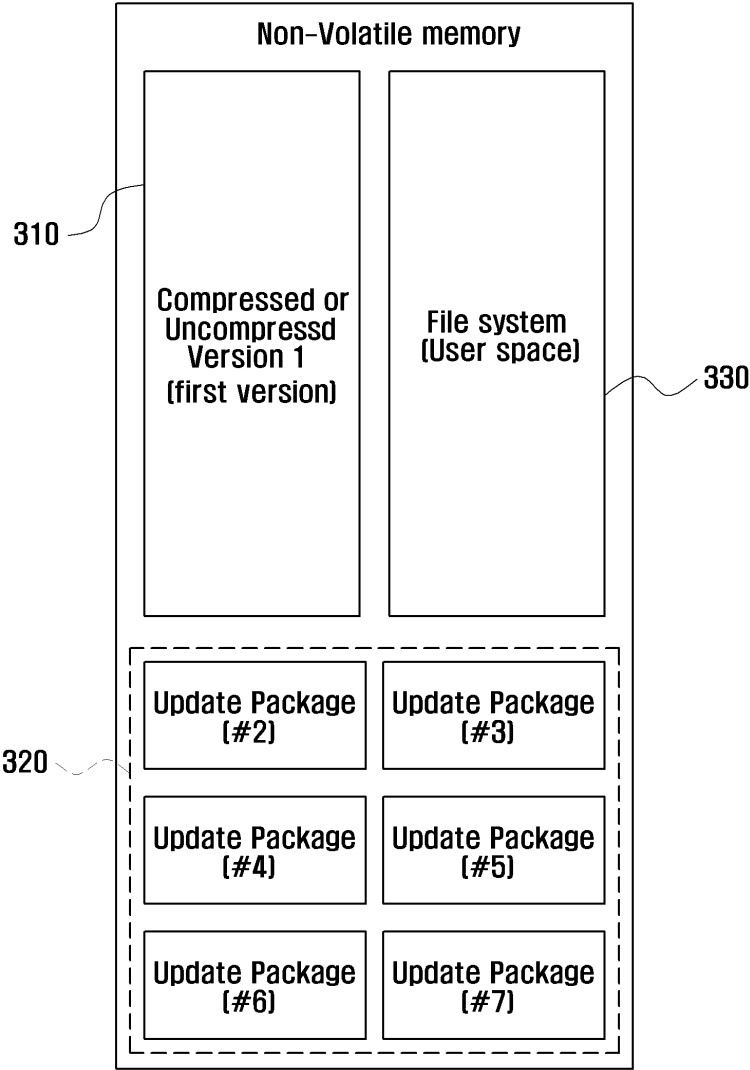


도면8

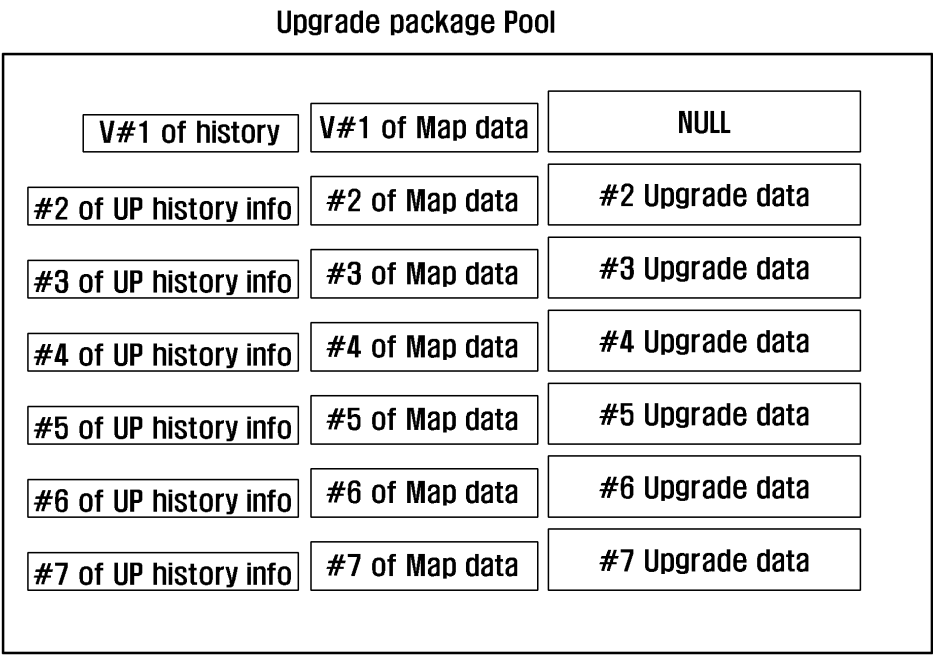
30



도면9



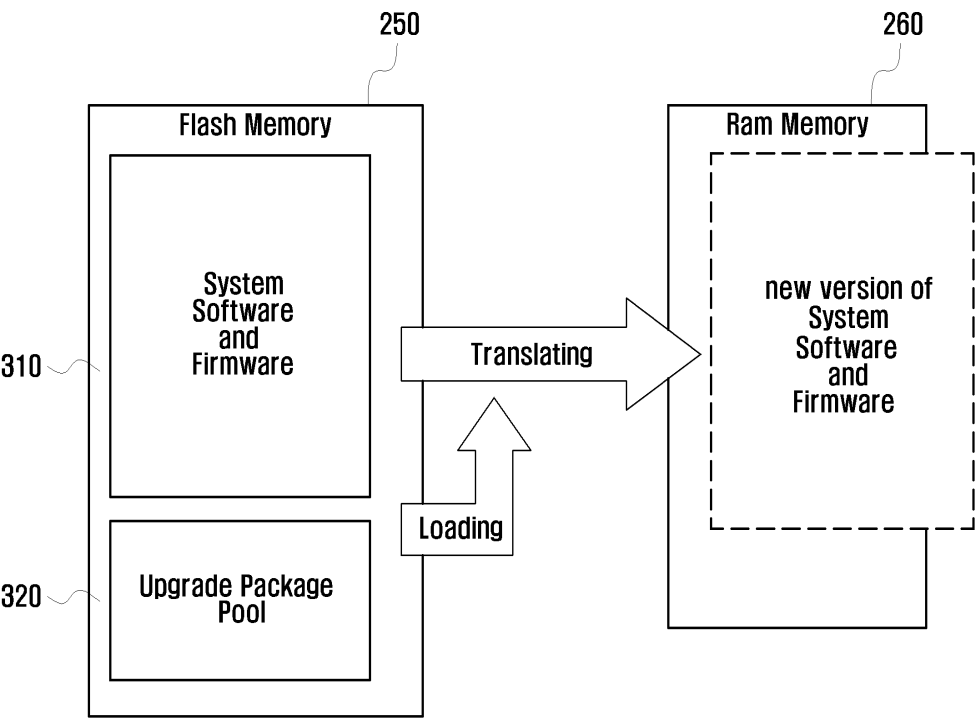
도면10a



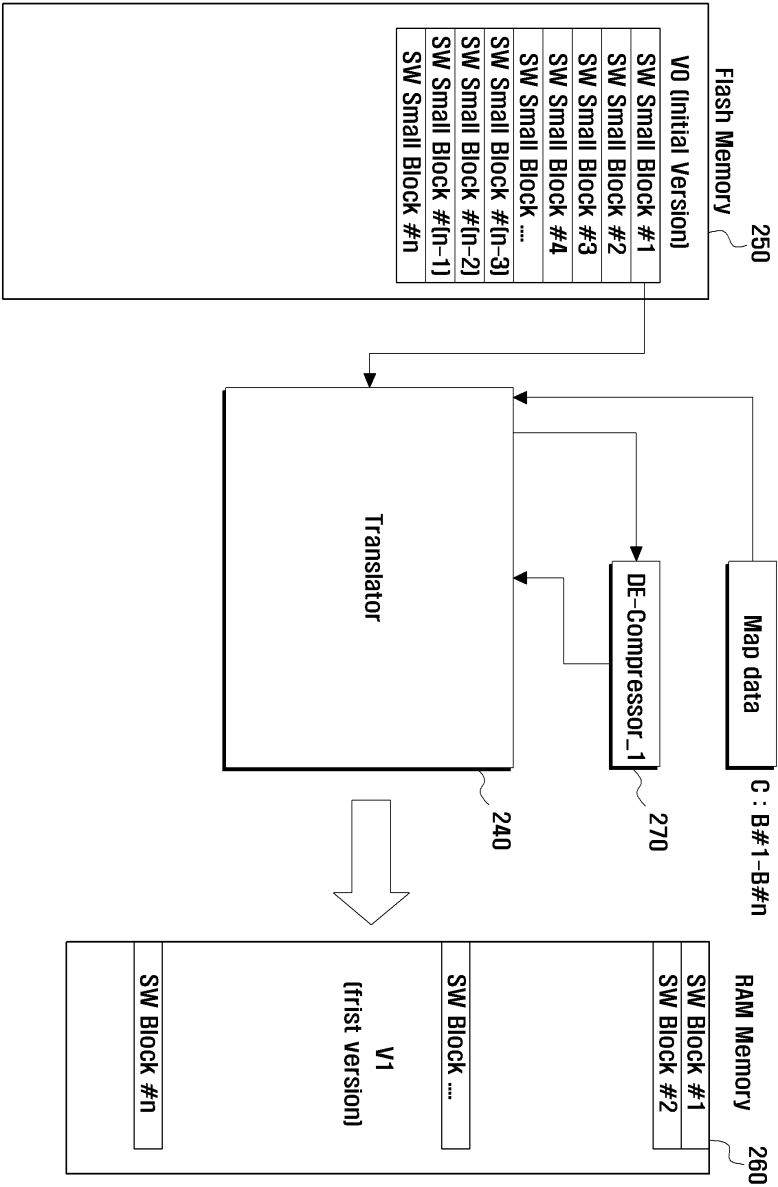
도면10b

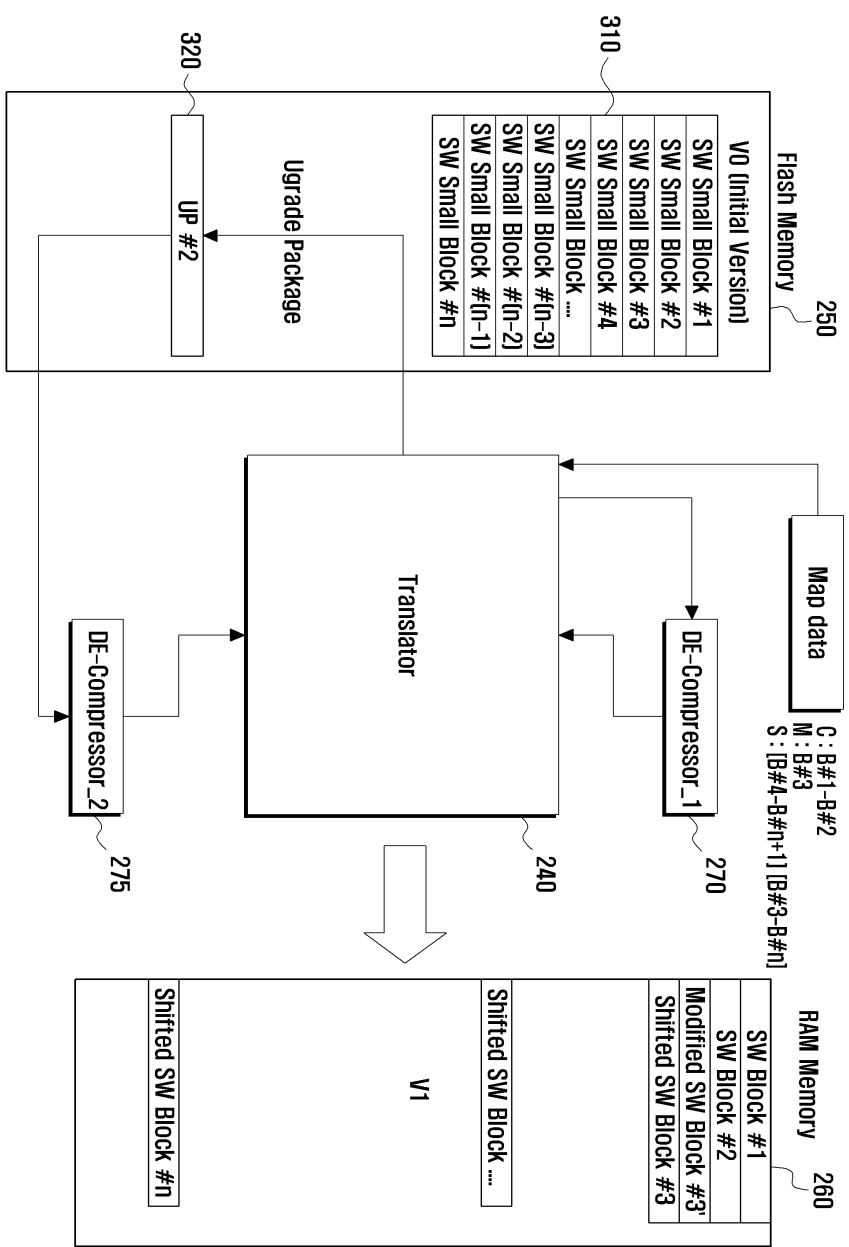
History structure
Version
Size
Combined Flag
Fail flag

도면11



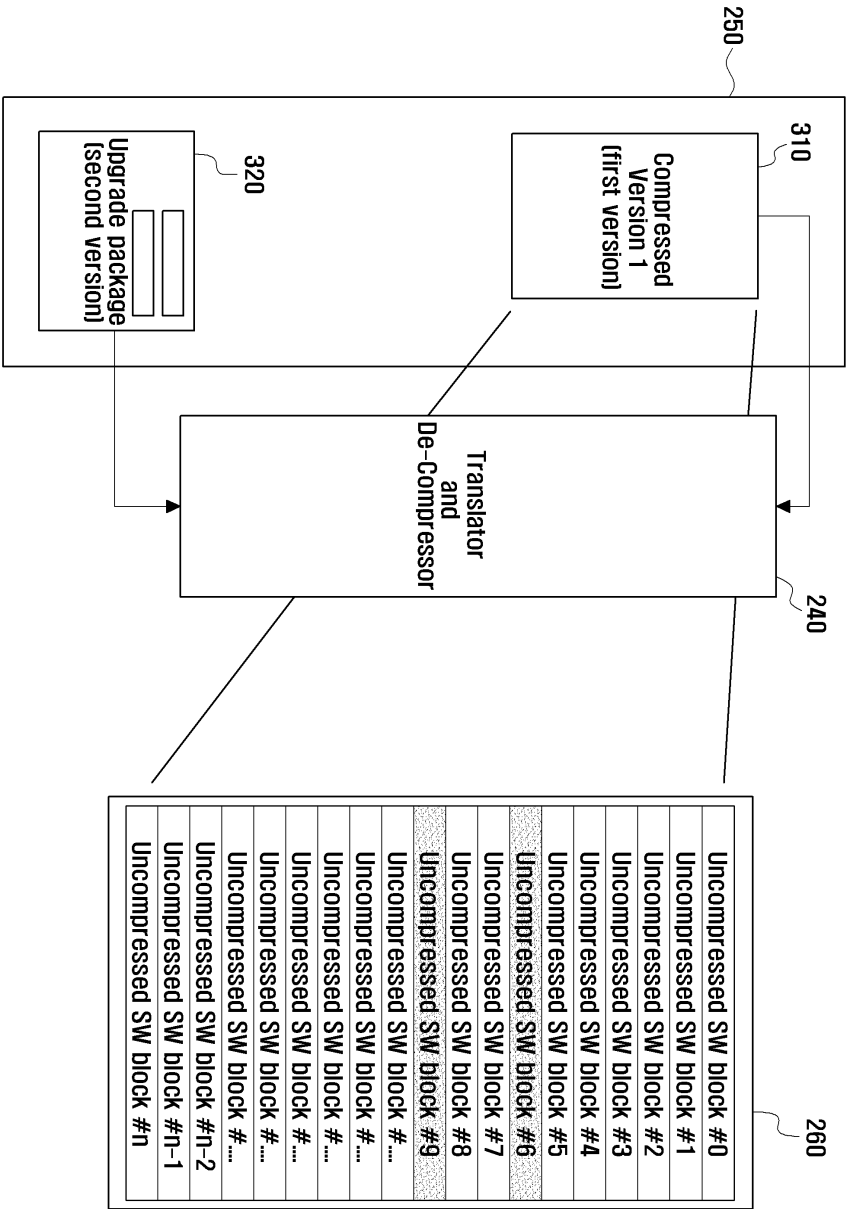
도면12



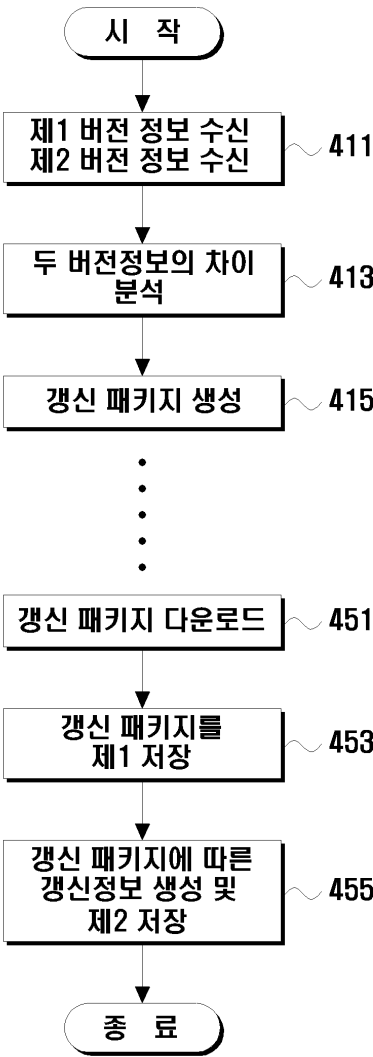


도면13

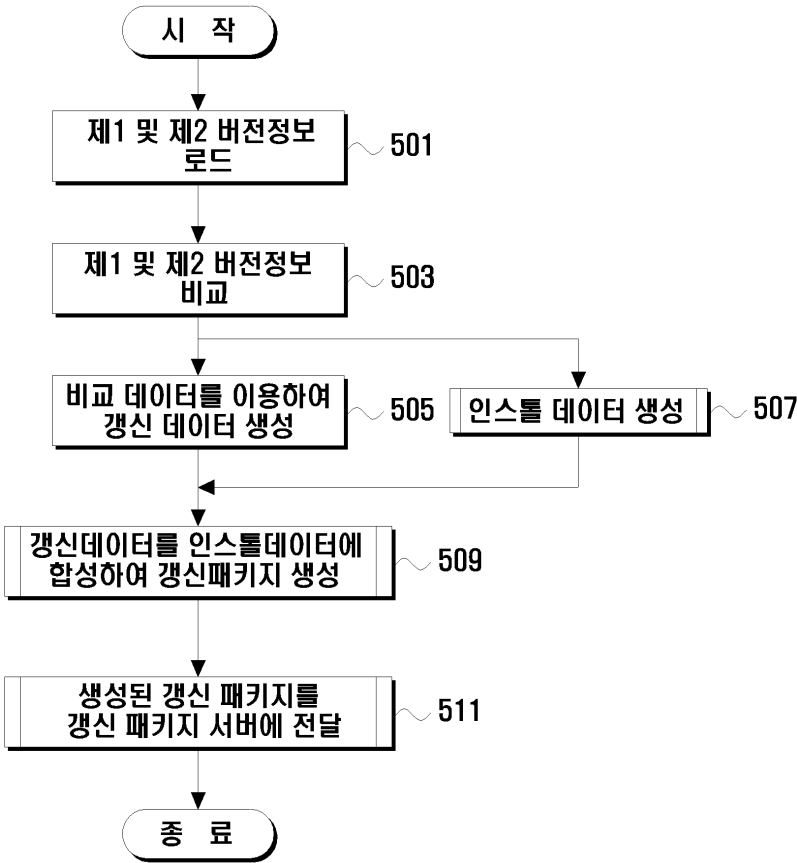
도면14



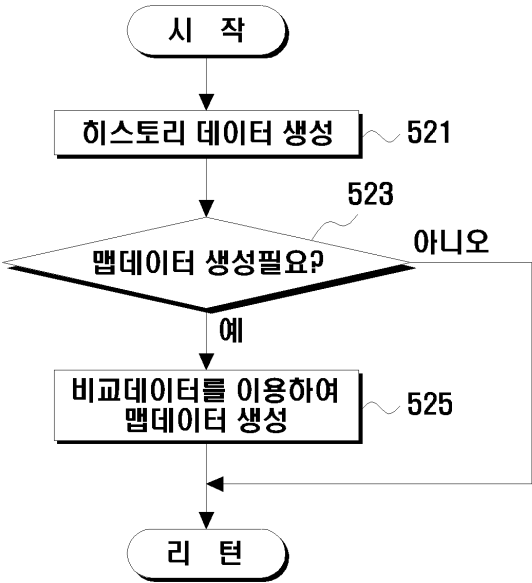
도면15



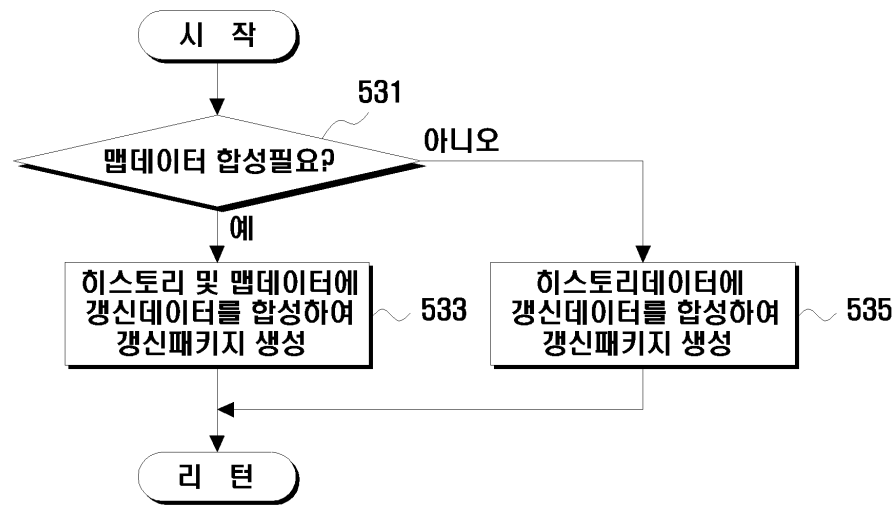
도면16a

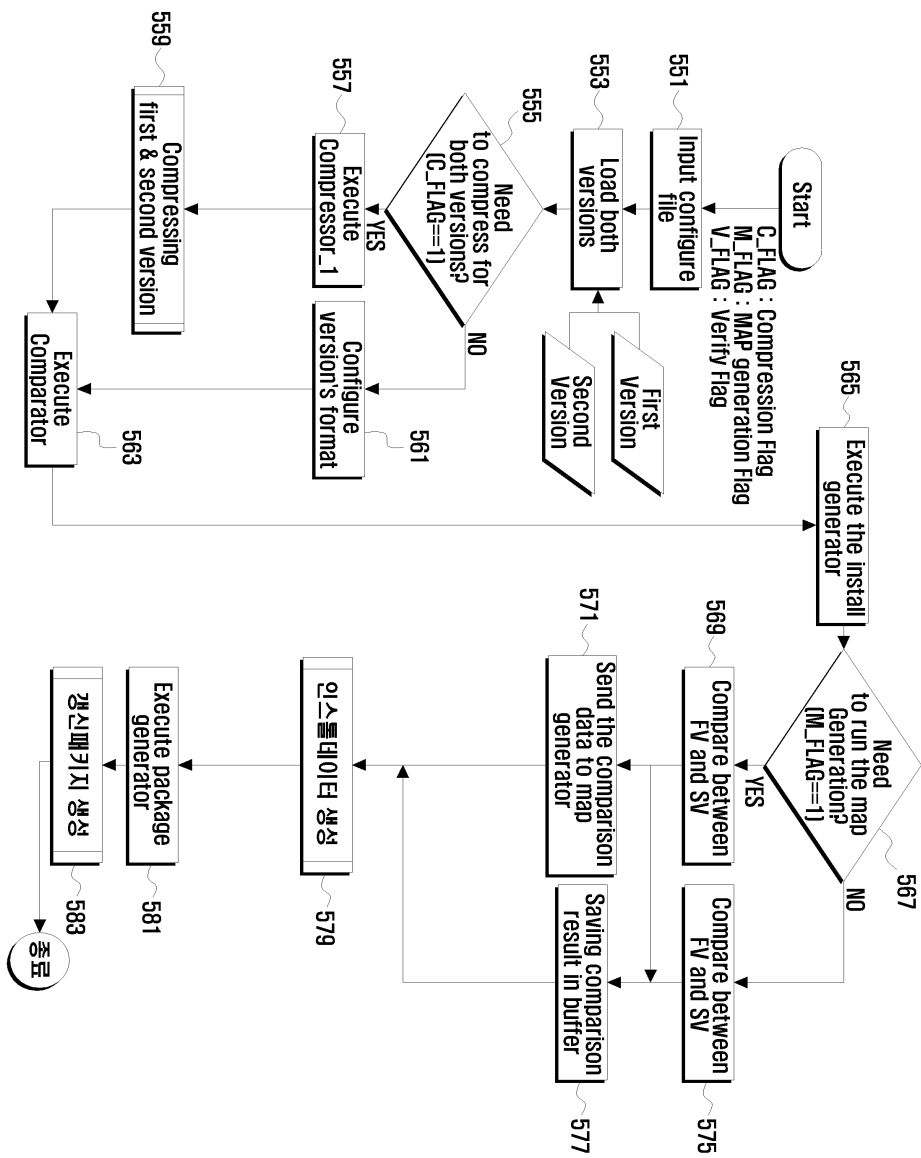


도면16b



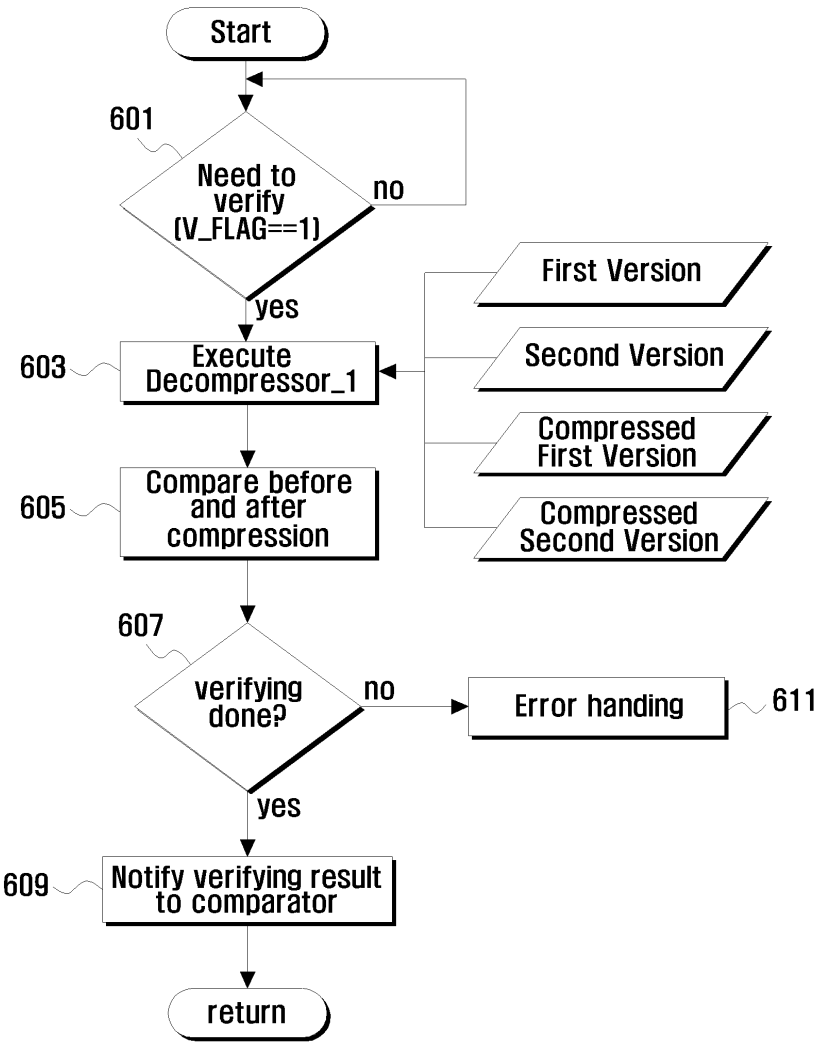
도면16c



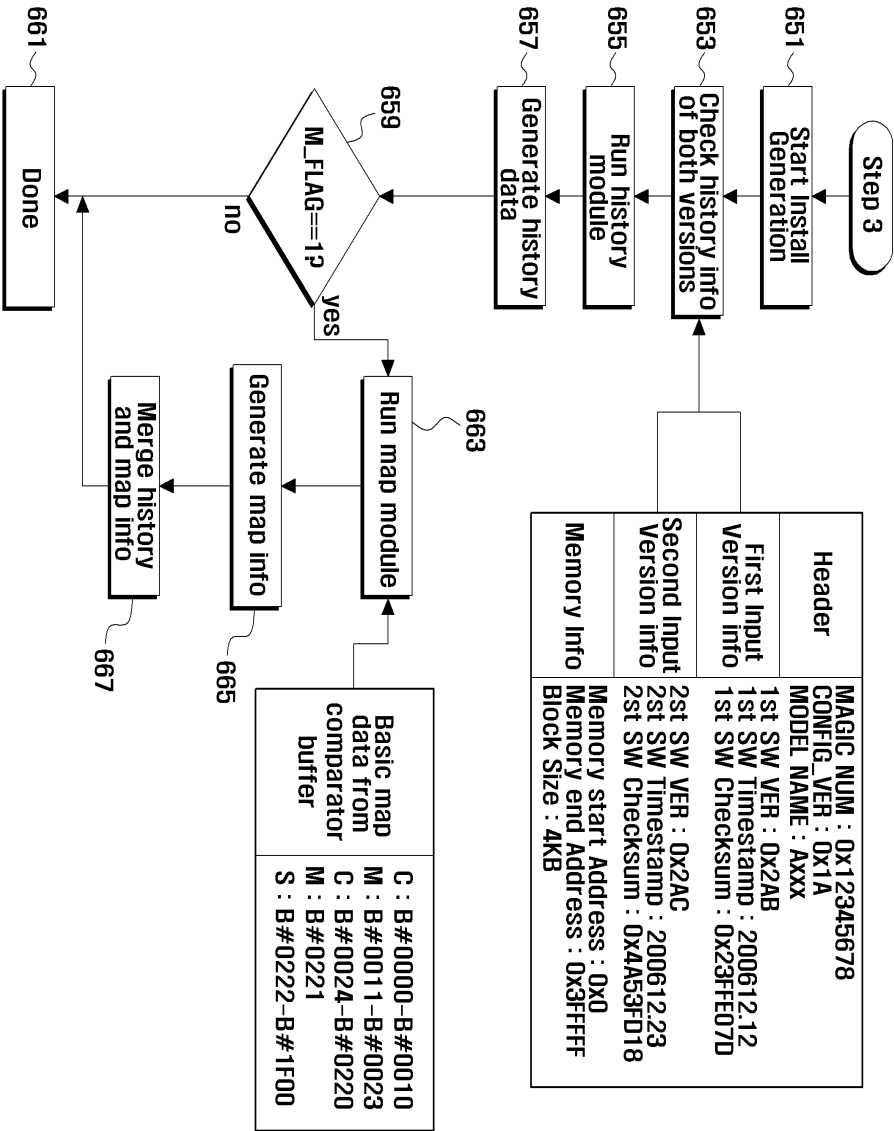


도면17

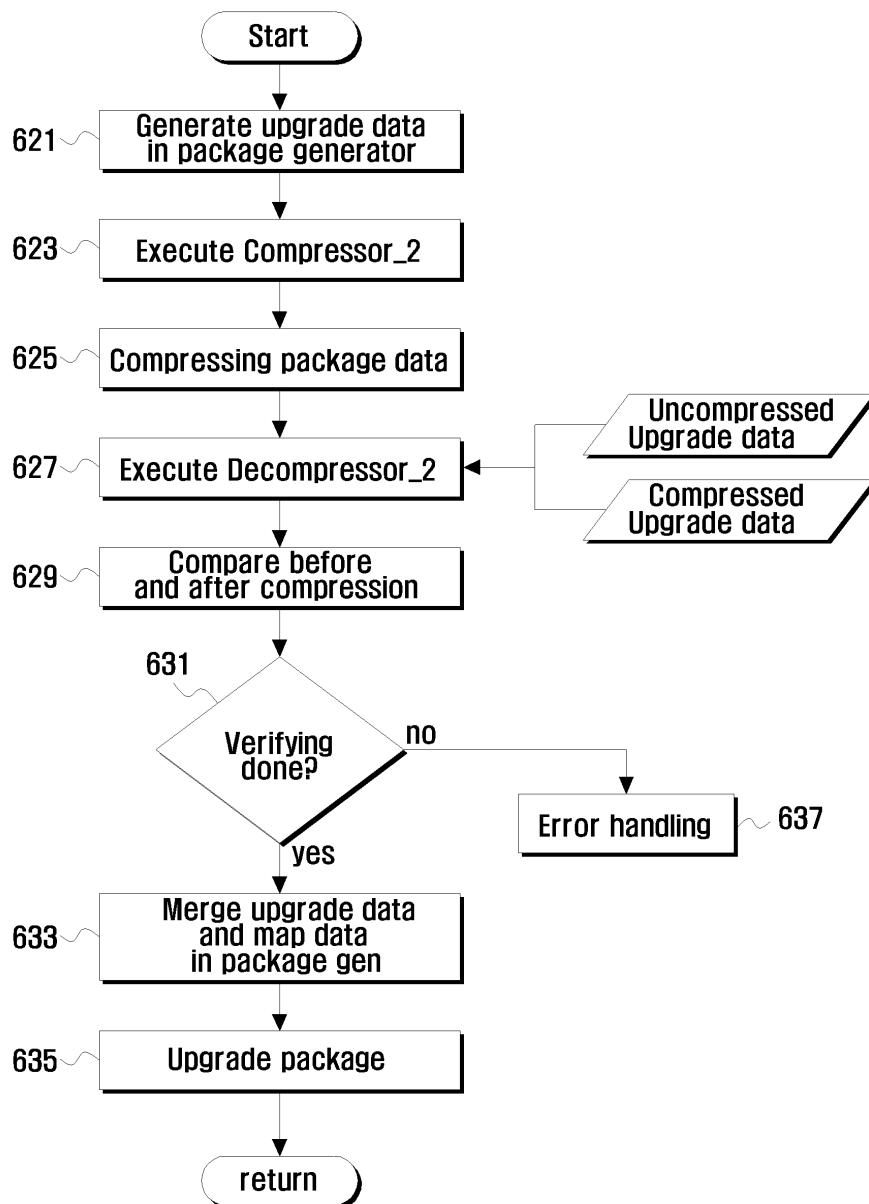
도면18



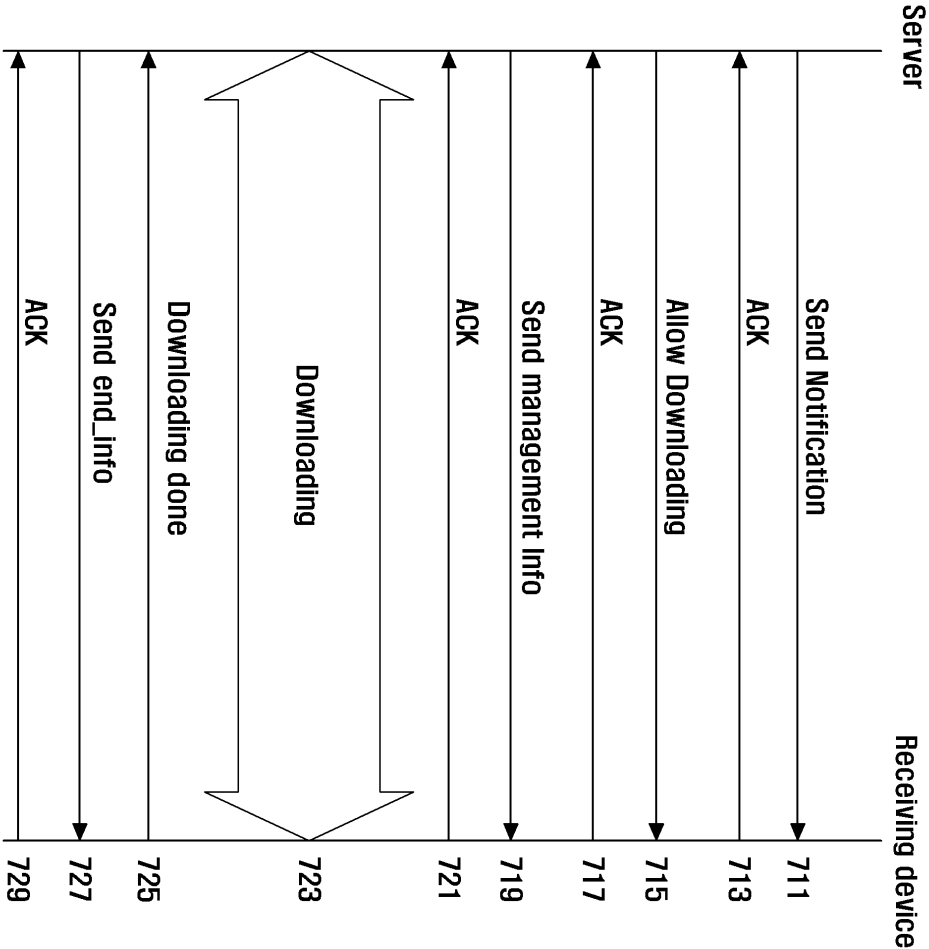
도면19



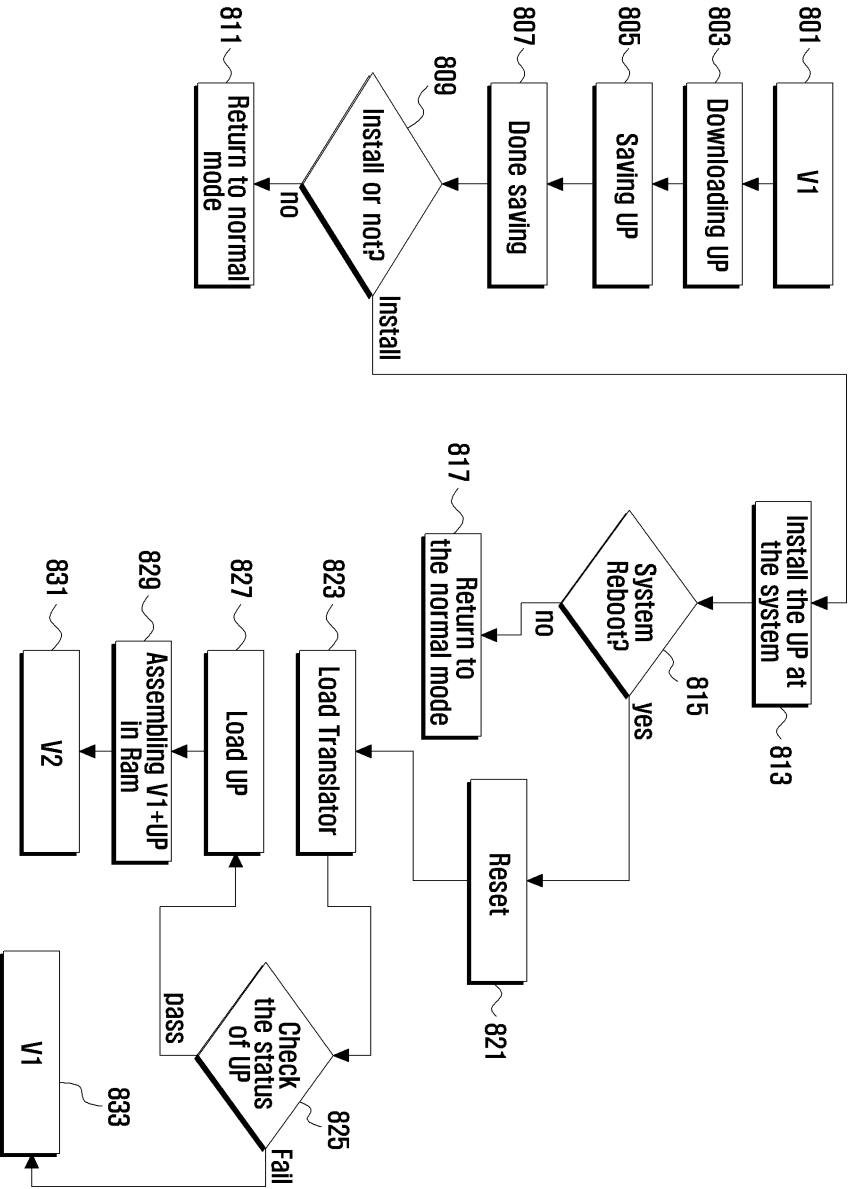
도면20



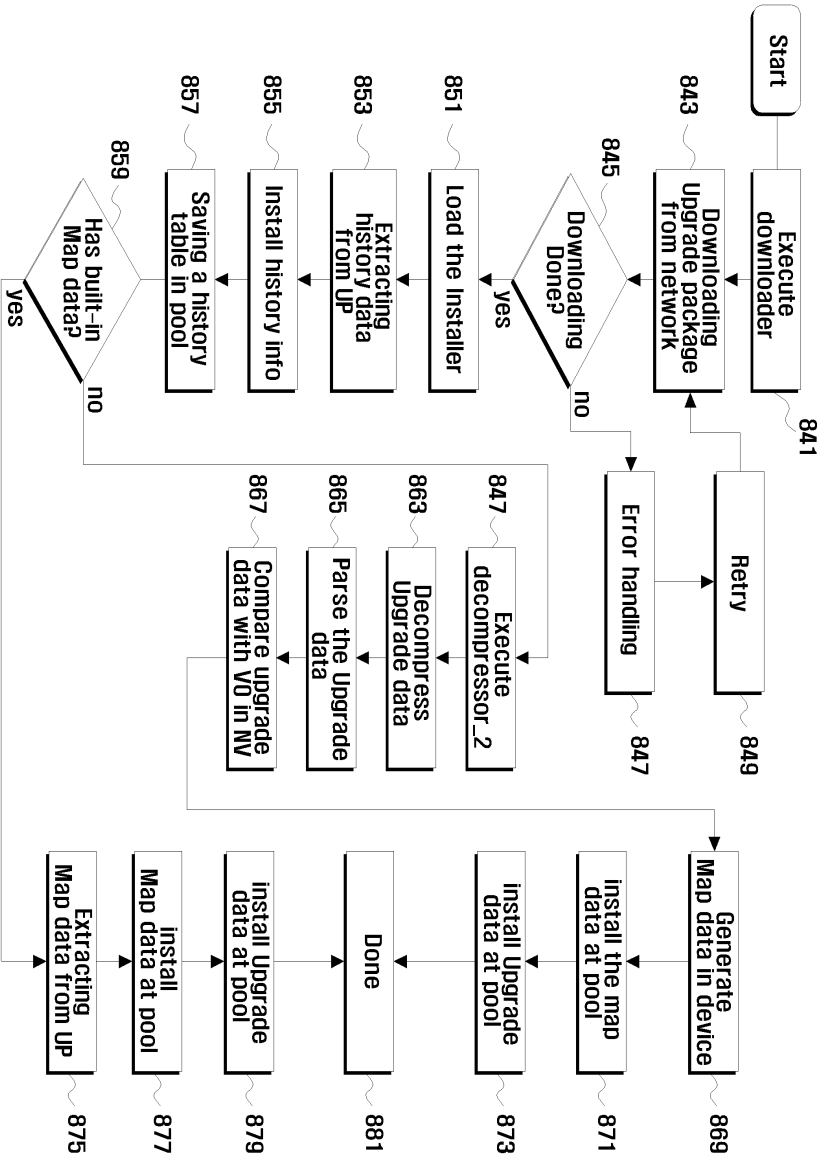
도면21



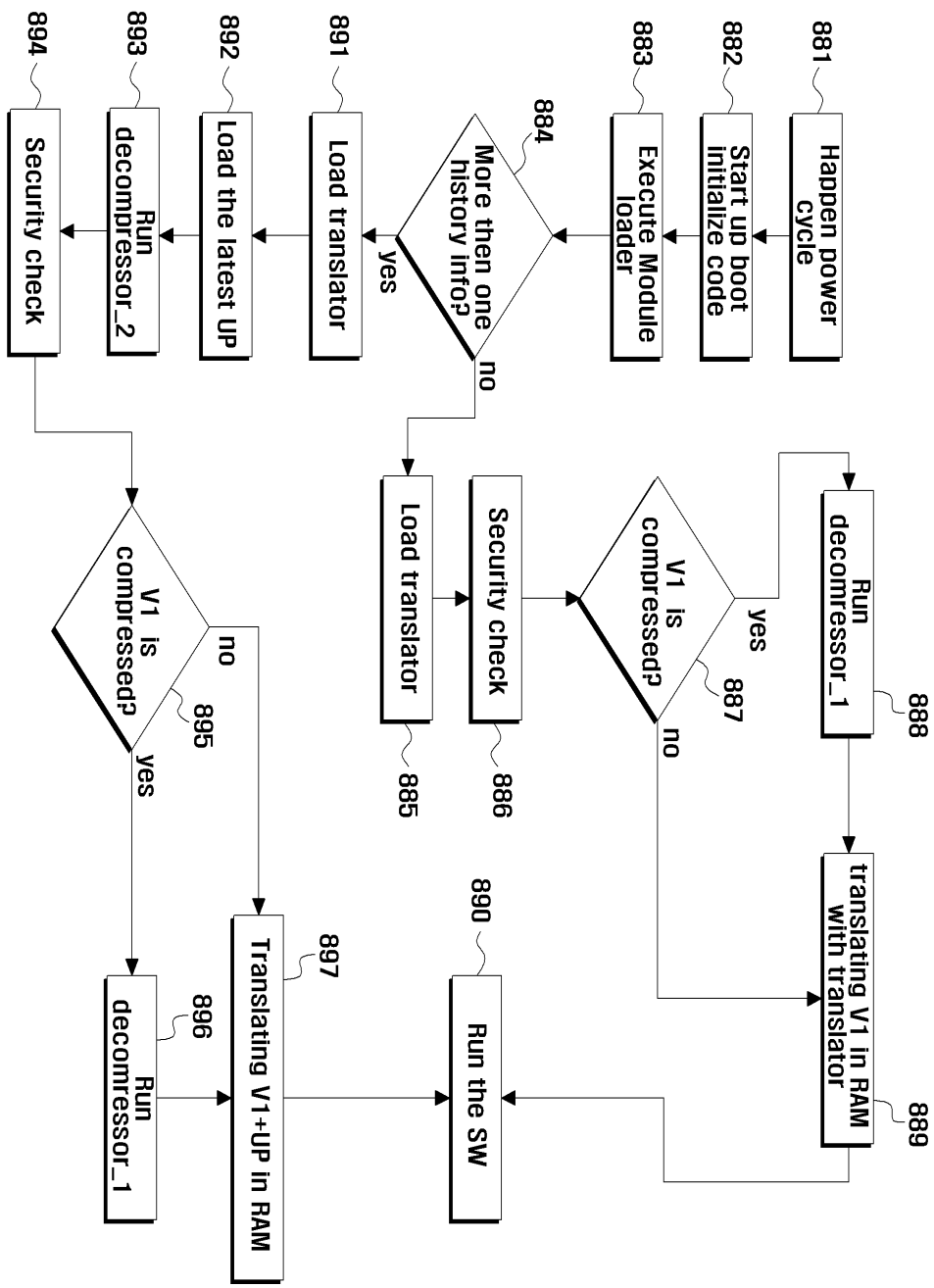
도면22

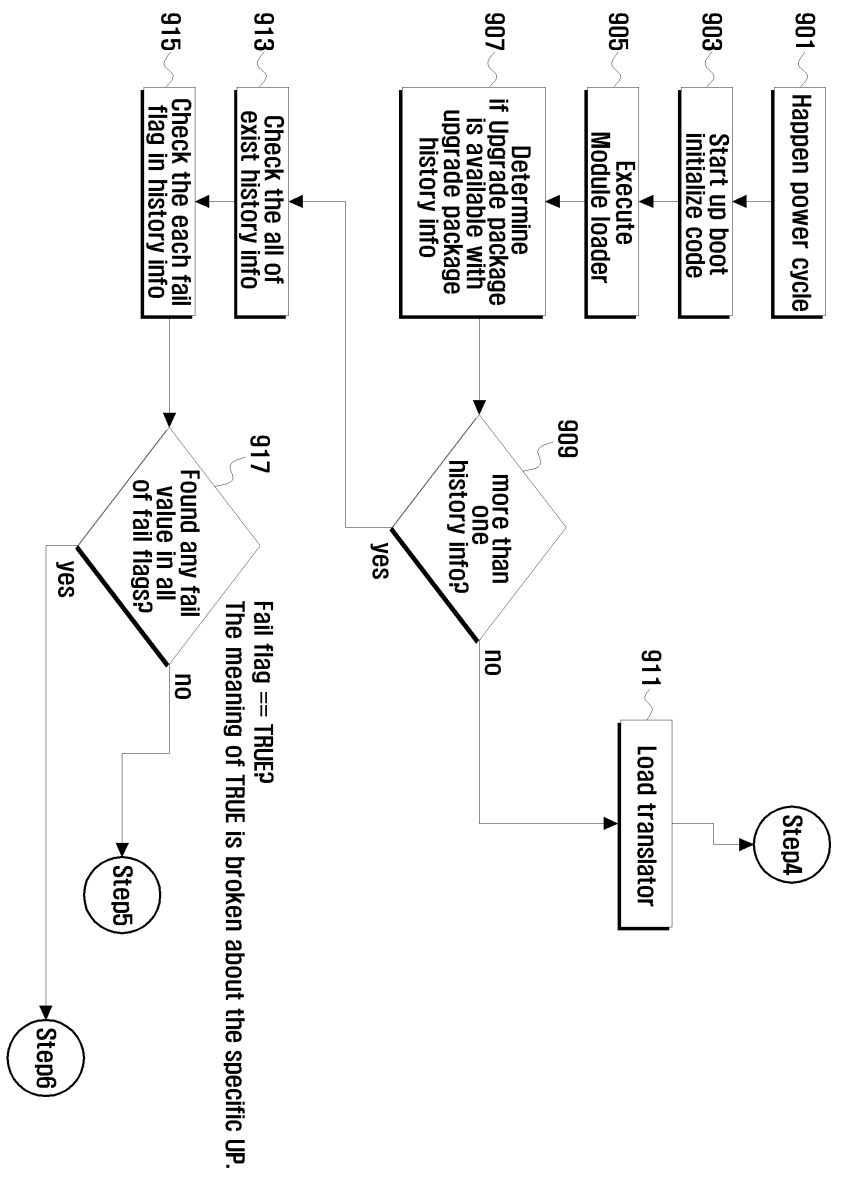


도면23



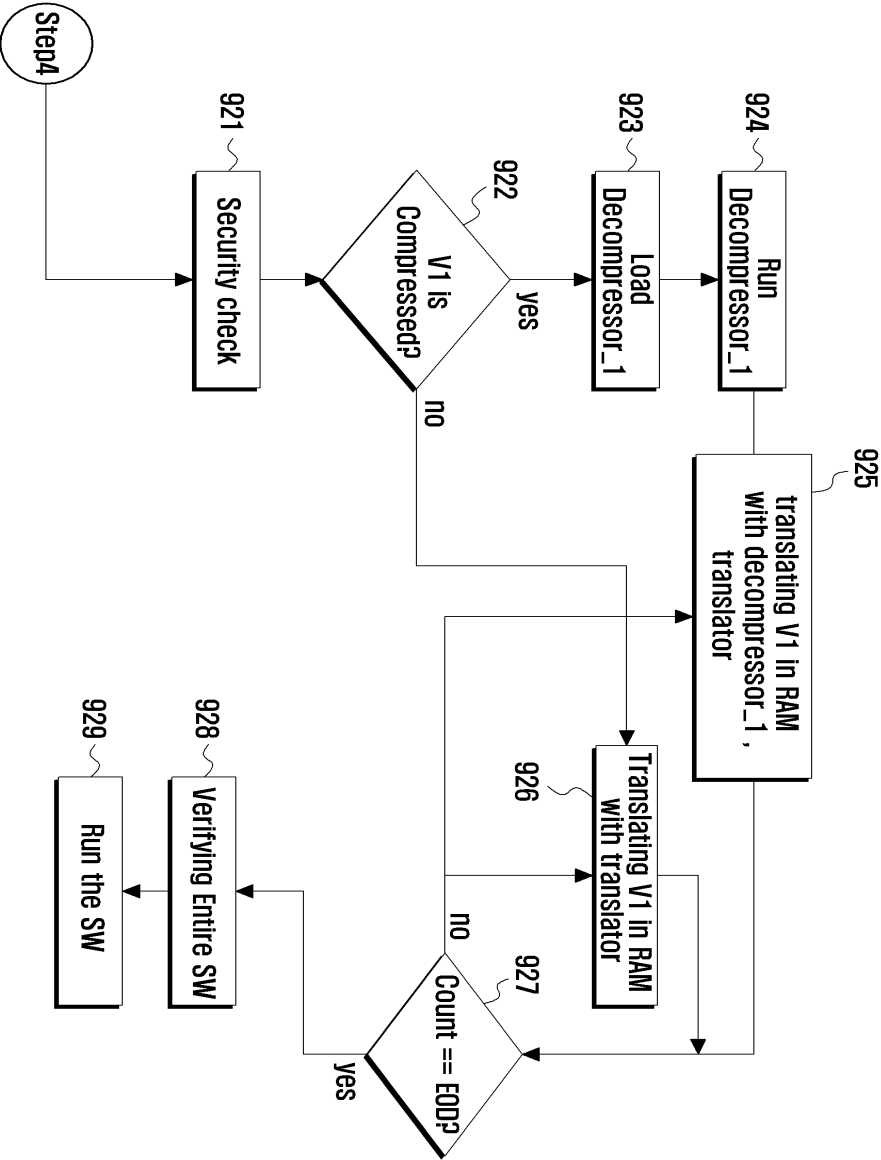
도면24

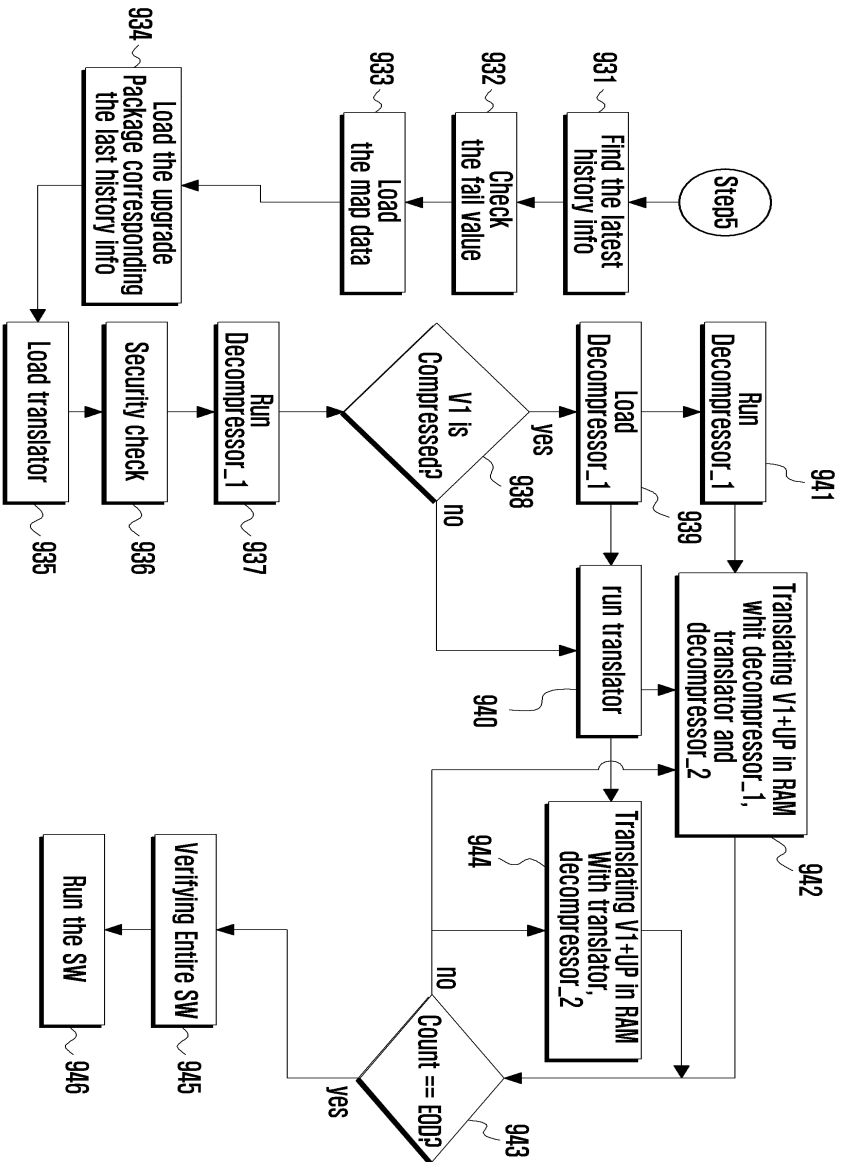




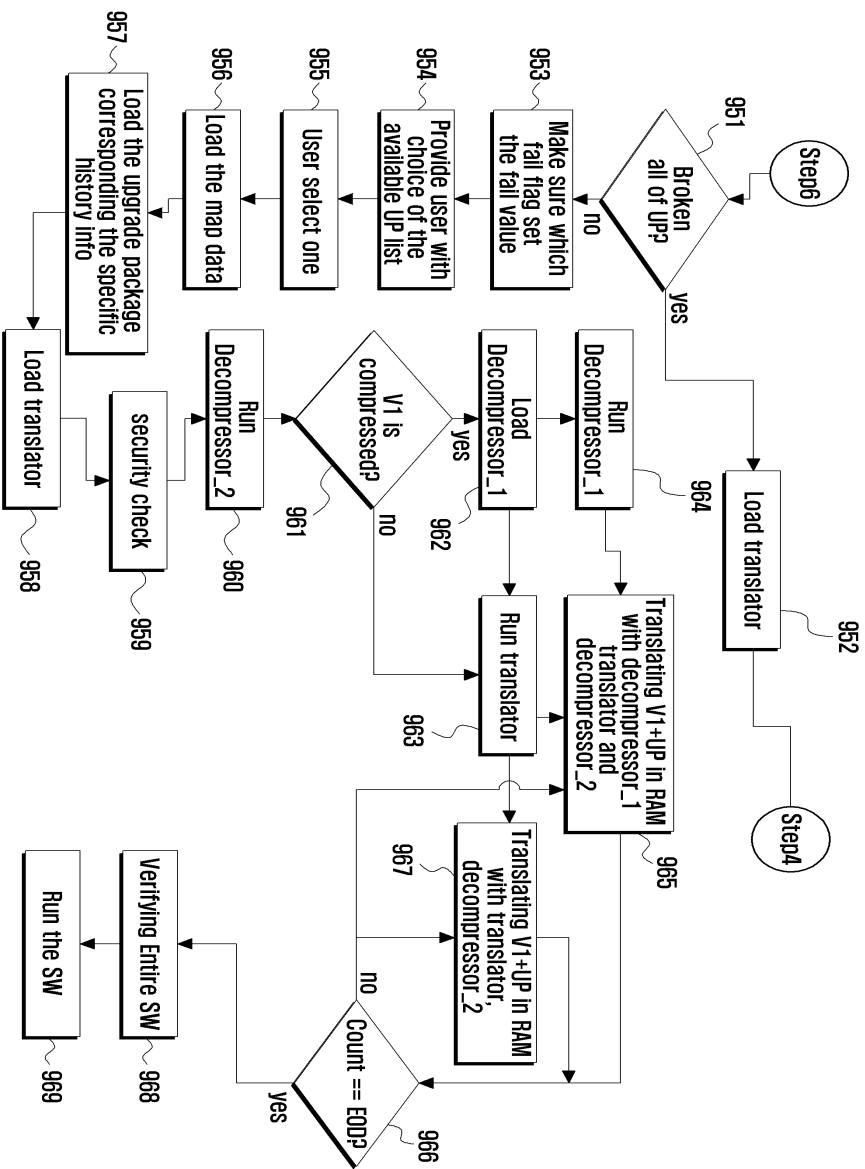
도면25a

도면25b





도면25c



도면25d