【명세서】

발명의 명칭

역델타 파일 생성 방법 및 이를 이용한 펌웨어 복구 방법{Method For Creating Inverse Delta File And Method For Recovering Firmware Using The Same}

발명의 상세한 설명

기술분야

본 발명은 역델타 파일 생성 방법 및 이를 이용한 펌웨어 복구 방법에 관한 것으로, 보다 상세하게는 펌웨어 업데이트을 중단한 경우 업데이트된 펌웨어를 원래대로 복구할 수 있는 역델타 파일 생성 방법 및 이를 이용한 펌웨어 복구 방법에 관한 것이다.

배경기술

최근들어, 휴대전화, PDA(Personal Digital Assistants), PMP(Portable Multimedia Player) 등과 같은 휴대용 단말기는 프로세서 기술의 발전과 메모리의 대용량화에 따라 다양한 기능을 포함하고 있고, 휴대용 단말기가 제조된 후에도 사용자의 다양한 요구에 부응하기 위해 기능이 지속적으로 향상되고 있다.

상기와 같이 휴대용 단말기가 제조된 이후의 기능 향상은 펌웨어(Firmware) 업데이트(update)를 통해 달성되는데, 상기 펌웨어는 휴대용 단말기를 운영하기 위한 운영체제와, 시스템 또는 응용프로그램과 데이터들이 저장된 비휘발성 메모리(예를 들면, ROM 또는 Flash Memory)로 구성된다.

일반적으로 펌웨어는 업데이트 가능하도록 구성되어 휴대용 단말기의 제조 후에도 업데이트 파일의 설치를 통해 각종 프로그램 또는 하드웨어의 개발로 인하여 추가된 새로운 기능들을 업데이트 파일을 설치함으로써 사용할 수 있도록 한다.

펌웨어를 업데이트 하기 위한 업데이트 파일은 일반적으로 용량이 크기 때문에 휴대용 단말기의 펌웨어를 업데이트 하기 위해서는 해당 서비스센터를 찾거나, 컴퓨터를 통해 업데이트 파일을 다운로드한 후 컴퓨터와 휴대단말기를 유선 또는 무선으로 연결하여 펌웨어를 업데이트 하였다.

도 1은 종래의 휴대용 단말기의 펌웨어 업데이트를 위한 시스템의 구성을 나타내는 블록도이다.

도 1을 참조하면, 종래의 펌웨어 업데이트 시스템은 펌웨어 업데이트가 가능하도록 설계된 휴대용 단말기 (10)와, 인터넷(30) 또는 보조 저장매체(40) 등을 통해 새로운 버전의 펌웨어 업데이트 파일을 제공받고 이를 해당 휴대용 단말 기(10)에 제공하는 컴퓨터(20)와, 상기 컴퓨터(20)와 휴대용 단말기(10)를 유선 또는 무선으로 연결할 수 있는 연결 수단(50)(여기서는 유선 케이블)으로 구성된다.

도 1에 도시된바와 같은 종래의 펌웨어 업데이트 시스템에서 휴대용 단말기(10)의 펌웨어는, 컴퓨터(20)에서 펌웨어 업데이트 전용 프로그램을 실행하여 연결된 휴대용 단말기(10)를 원격 제어하면서 펌웨어 업데이트를 수행한다.

또는, 휴대용 단말기(10)의 운영체제에서 동작하는 전용 펌웨어 업데이트 프로그램 및 업데이트할 펌웨어 업데이트 파일을 컴퓨터(20)에서 휴대용 단말기(10)로 다운로드한 후 휴대용 단말기(10)에서 해당 전용 펌웨어 업데이트 프로그램을 동작시켜 펌웨어의 버전을 체크한 후 기존 버전을 소거하고 새로운 버전의 펌웨어 업데이트 파일을 복사함으로써 휴대용 단말기(10)의 펌웨어 업데이트가 이루어진다.

그러나, 상기한 바와 같은 종래의 펌웨어 업데이트 방법은 펌웨어를 구성하는 비휘발성 메모리에 펌웨어 업데

이트를 위한 파일 전체가 기록되므로 사용자의 실수나 운영체제의 오류, 시스템 재부팅, 전원 부족, 전원 케이블 접촉 불량 등과 같은 비정상적인 상황이 발생되면 펌웨어 업데이트 과정을 처음부터 다시 수행하여야 하므로 시간이 낭비되는 문제점이 있었다.

상기와 같은 문제점을 해결하기 위한 기술로 본원 출원인이 선출원한 대한민국 특허출원 10-2006-0081824 호에는 이전 버전의 기존 파일과 신규 버전의 새 파일을 차이 알고리즘(Difference Algorithm)을 통해 블록 단위로 비교하여 차이가 있는 블록을 압축 알고리즘을 통해서 선택적으로 압축을 수행(즉, 압축 후 사이즈가 압축전 사이즈보다 적을 경우에만 압축을 수행)하며, 차이가 있는 블록의 옵셋(Offset) 정보와 압축 여부 및 압축 전/후 사이즈, 신규버전의 패치 데이터 등을 포함한 델타 블록을 생성하고, 생성된 델타 블록들로 구성된 하나의 델타파일로 생성한 후, 상기 블록 옵셋 정보를 이용하여 휴대용 단말기의 해당 블록을 업데이트하는 휴대 단말기의 펌웨어 업데이트용 델타 파일 생성 방법과 델타 파일 및 이를 이용한 펌웨어 업데이트 방법을 실시하여 펌웨어 업데이트 속도를 향상시킨 기술을 개시하고 있다.

그러나, 상기 델타 파일을 이용한 펌웨어 업데이트 방법에서는 펌웨어의 원본 파일을 새로운 펌웨어 업데이트 파일로 덮어쓰기 때문에, 외부환경 또는 사용자의 요구, 바이너리 데이터의 오류 등에 의하여 업데이트가 정 상적으로 종료되지 않고 실패 또는 취소되는 경우 펌웨어가 새로운 업데이트 파일과 원본 파일이 혼재되어 있어서 복구가 불가능하게 되고, 이로 인해 프로그램을 처음부터 새로 설치한 후 펌웨어 업데이트를 다시 수행하여야 하기 때문에 펌웨어 업데이트 시간이 길어지는 단점이 있다.

발명의 내용

해결 하고자하는 과제

상기와 같은 문제점을 해결하기 위한 본 발명의 제1 목적은 펌웨어 업데이트가 실패 또는 취소된 경우 펌웨어 를 원래의 상태로 용이하게 복구할 수 있도록 하는 역델타 파일 생성 방법을 제공하는 것이다.

또한, 본 발명의 제2 목적은 펌웨어 업데이트가 실패 또는 취소된 경우 펌웨어를 원래의 상태로 용이하게 복구할 수 있는 역델타 파일을 이용한 펌웨어 복구 방법을 제공하는 것이다.

과제 해결수단

상기한 본 발명의 제1 목적을 달성하기 위한 본 발명의 일 실시예에 따른 역델타 파일 생성 방법은, 대응되는 원본 파일 블록에 대한 수정 작업을 지시하는 적어도 하나의 명령어 코드와 명령어 코드에 대응되는 원본 파일 블록의 옵셋 및/또는 패치 데이터를 포함하는 적어도 하나의 델타 블록으로 구성된 델타 파일을 이용한 펌웨어 업데이트 방법에서, 상기 델타 블록에 포함된 명령어 코드에 대응되는 역 명령어 코드를 파악하고, 상기 명령어 코드에 의한 원본파일 블록에 대한 수정작업을 복원할 수 있도록 상기 역 명령어 코드에 대응되는 원본 파일 블록의 옵셋 및/또는 복원 데이터를 파악하여 역델타 블록을 생성하는 단계와, 상기 델타 블록을 이용하여 대응되는 원본 파일 블록에 대한 수정작업을 수행하는 단계 및 상기 델타 블록의 위치에 상기 역델타 블록을 기록하는 단계를 포함한다.

상기 역델타 블록을 생성하는 단계는 상기 생성된 역델타 블록을 버퍼에 임시 저장하는 단계를 포함할 수 있다. 상기 역델타 블록을 생성하는 단계는 상기 명령어 코드, 상기 역명령어 코드, 상기 델타 블록에 포함된 수정 데이터, 상기 델타 블록에 의해 수정되는 원본 파일 블록의 데이터 중 적어도 하나에 기초하여 상기 복원데이터가 결정될 수 있다.

또한, 본 발명의 제1 목적을 달성하기 위한 본 발명의 다른 실시예에 따른 역델타 파일을 이용한 펌웨어 복구방법은, 적어도 하나의 델타 블록을 포함한 델타 파일을 이용한 펌웨어 업데이트 과정에서 펌웨어 업데이트가성공적으로 수행되지 못한 경우-업데이트 취소 또는 실패-에 수정된 원본 파일 블록을 복구하기 위한 역델타파일을 생성하는 방법에 있어서, 소정의 델타 블록에 대응되는 원본 파일 블록을 버퍼에 저장하는 임시 저장단계와, 상기 델타 블록을 적용하여 상기 델타 블록에 대응되는 원본 파일 블록을 수정하는 패치 단계와, 상기수정된 원본 파일 블록을 상기 버퍼에 저장된 원본 파일 블록 형태로 복원하는 역델타 블록을 생성하는 역델

타 블록 생성 단계 및 상기 패치 단계 및 역델타 블록 생성 단계 완료 후 상기 역델타 블록을 상기 델타 파일의 대응되는 델타 블록의 위치에 기록하여 역델타 파일을 생성하는 역델타 파일 생성 단계를 포함한다. 상기 역 델타 블록 생성 단계는, 상기 수정된 원본 파일 블록과 상기 버퍼에 저장된 원본 파일 블록에 대해 차이 알고리즘을 적용하고, 상기 차이 알고리즘에 의하여, 상기 수정된 원본 파일 블록을 상기 버퍼에 저장된 원본 파일 블록의 형태로 복원할 수 있는 역델타 블록을 생성할 수 있다.

또한, 본 발명의 제2 목적을 달성하기 위한 본 발명의 일 실시예에 따른 역델타 파일을 이용한 펌웨어 복구 방법은, 역델타 파일 생성 방법에 의하여 역델타 파일을 생성하면서, 델타 파일을 이용하여 펌웨어를 업데이트 하는 과정에서, 펌웨어 업데이트의 진행 중 또는 펌웨어 업데이트의 종료의 경우에, 펌웨어 업데이트 취소 또는 실패 여부를 판단하는 단계 및 펌웨어 업데이트가 취소 또는 실패인 경우, 상기 역델타 파일을 이용하여 원본 파일을 복구하는 단계를 포함한다. 상기 펌웨어 업데이트 취소 또는 실패 여부를 판단하는 단계는, 업데이트된 펌웨어 유효성을 검증하여 상기 펌웨어 업데이트의 취소 또는 실패 여부를 판단할 수 있다.

효과

상기와 같은 역델타 파일 생성 방법 및 이를 이용한 펌웨어 복구 방법에 따르면, 외부 장치로부터 제공된 델타파일에 기초하여 원본 파일 블록을 패치하는 과정에서 각 델타 블록에 기초하여 역델타 블록을 생성하거나, 또는 원본 파일 블록과 이에 대응되는 패치된 파일 블록에 차이 알고리즘을 적용하여 역델타 파일을 생성한후 델타 블록이 기록된 위치에 저장한다. 그리고, 사용자의 업데이트 취소 명령 또는 업데이트 환경에 의해 펌웨어 업데이트가 성공적으로 수행되지 않은 경우 생성된 역델타 파일을 이용하여 패치된 원본 파일을 원래의파일로 복원한다.

따라서, 원본 파일이 성공적으로 수행되지 못한 경우에도 용이하게 패치된 파일을 원본 파일로 복구할 수 있다. 또한, 생성된 역델타 블록은 대응되는 델타 블록과 동일한 사이즈를 가지기 때문에 역델타 블록을 원본 블록 파일의 패치가 종료된 후 해당 델타 파일이 기록된 위치에 기록함으로써 별도의 메모리 공간 필요로 하지않게 되어 메모리 사용 효율을 높일 수 있다.

발명의 실시를 위한 구체적인 내용

본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 상세하게 설명하고자 한다.

그러나, 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다.

제1, 제2 등의 용어는 다양한 구성요소들을 설명하는데 사용될 수 있지만, 상기 구성요소들은 상기 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 사용된다. 예를 들어, 본 발명의 권리 범위를 벗어나지 않으면서 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소도 제1 구성요소로 명명될 수 있다. 및/또는 이라는 용어는 복수의 관련된 기재된 항목들의 조합 또는 복수의 관련된 기재된 항목들 중의 어느 항목을 포함한다.

어떤 구성요소가 다른 구성요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다.

본 출원에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 출원에서, "포함하다" 또는 "가지다" 등의 용어는 명세서상에 기재된 특징, 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가지고 있다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥 상 가지는 의미와 일치하는 의미를 가진 것으로 해석되어야 하며, 본 출원에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식 적인 의미로 해석되지 않는다.

이하, 첨부한 도면들을 참조하여, 본 발명의 바람직한 실시예를 보다 상세하게 설명하고자 한다. 본 발명을 설명함에 있어 전체적인 이해를 용이하게 하기 위하여 도면상의 동일한 구성요소에 대해서는 동일한 참조부호를 사용하고 동일한 구성요소에 대해서 중복된 설명은 생략한다.

이하, 본 발명의 실시예에서 역델타 파일은 델타 파일에 의해 업데이트된 펌웨어의 소정 데이터를 원래의 데이터로 복구하기 위한 정보를 포함하는 파일을 의미한다. 또한, 본 발명의 실시예에 기재된 원본 파일은 델타파일에 의해 업데이트 되기 이전의 파일을 의미하고, 신규 파일은 델타파일에 의해 업데이트된 파일을 의미하며, 복원 파일은 역델타 파일에 의해 업데이트된 원본 파일이 업데이트 되기 이전의 파일로 복원된 파일을 의미하다.

도 2는 본 발명의 일 실시예에 따른 역델타 파일을 이용한 펌웨어 복구를 수행하는 장치의 구성을 나타내는 블록도로서, 휴대용 단말기를 예를 들어 설명한다.

도 2를 참조하면, 휴대용 단말기(100)는 업데이트/복원부(110), 펌웨어(120), 저장부(130)를 포함하여 구성 될 수 있고, 업데이트/복원부(110)는 역델타 파일 생성 모듈(111)을 포함할 수 있다.

업데이트/복원부(110)는 유선 또는 무선 인터페이스를 통해 연결된 외부 장치(예를 들면, 컴퓨터)로부터 펌웨어의 업데이트를 위한 델타 파일을 제공받고, 제공받은 델타 파일을 펌웨어(120)의 여유 공간 또는 저장부 (130)에 저장한다.

또한, 업데이트/복원부(110)는 검증값(Checksum)을 이용하여 파일의 유효성을 검증하고, 델타 파일의 압축 여부를 판단하여 압축된 경우에는 압축을 해제하고, 델타 파일을 구성하는 각각의 델타 블록에 포함된 블록 옵셋 정보를 이용하여 펌웨어(120)에 기록된 해당 원본 파일 블록에 대한 수정 작업(patch)을 수행한다.

또한, 업데이트/복원부(110)의 역델타 파일 생성 모듈(111)은 상기와 같이 원본 파일 블록에 대한 패치가 수행되는 과정에서 펌웨어 업데이트가 성공적으로 수행되지 않은 경우 업데이트된 원본 파일 블록 즉, 신규 파일 블록을 원래의 파일 블록으로 복원하기 위한 역델타 파일을 생성하고 소정의 원본 파일 블록에 대한 패치가 끝나면 생성된 역델타 파일을 패치가 끝난 원본 파일 블록에 대응되는 델타 블록이 기록된 위치에 기록한다.

업데이트/복원부(110)는 펌웨어의 업데이트가 종료되면 검증값(Checksum)을 이용하여 업데이트의 성공 여부를 판단하고, 업데이트가 성공적으로 종료되지 않은 것으로 판단되면 기록된 역델타 블록을 이용하여 패치가 수행된 원본 파일 블록을 원래대로 복원한다.

펌웨어(120)는 실질적으로 플래시 롬(Flash-ROM)으로 구성될 수 있고, 소정 크기를 가지는 블록 단위로 데이터의 기록이 이루어진다.

저장부(130)는 비휘발성 메모리(예를 들면, EEPROM 또는 플래시 메모리) 또는 휘발성 메모리(RAM)로 구성될 수 있고, 소정 크기의 버퍼(131) 공간을 포함하여 역델타 파일의 생성과정에서 원본 파일 블록 또는 생성된 역델타 파일이 임시 저장된다.

도 3은 본 발명의 일 실시예에 따른 델타 파일의 구성을 나타내는 블록도이고, 도 4는 도 3에 도시된 델타 데이터의 포맷을 나타내는 블록도이다.

도 3 및 도 4를 참조하면, 본 발명의 일 실시예에 따른 델타 파일(200)은 크게 델타 헤더(Delta Header)(210)

와 델타 바디(Delta Body)(220)로 구성된다.

델타 헤더(210)는 식별자(ID), 검증값(Checksum), 사이즈(Size) 및 파일 정보를 포함한다.

식별자(ID)는 5바이트로 구성될 수 있고, 처음 1바이트는 델타 파일(200)의 다운로드 시작과 종료를 의미하는 다운로드 플래그로서 "1" 또는 "0"을 가질 수 있다. 이후의 4바이트는 바이너리 업데이트를 나타내는 예약된 문자열을 가진다.

검증값(Checksum)은 5바이트로 구성될 수 있고 델타 파일(200)의 유효성 검증을 위해 사용된다. 사이즈 (Size)는 4바이트로 구성될 수 있고 델타 파일(200)의 전체 크기를 나타낸다.

파일 정보는 원본 파일에 대한 검증값(Ori_Checksum)과, 원본 파일의 사이즈(Ori_Filesize)와, 신규 파일에 대한 검증값(New_Checksum)과, 신규 파일에 대한 사이즈(New_Filesize)를 포함할 수 있고, 상기 원본 파일에 대한 검증값(Ori_Checksum), 원본 파일의 사이즈(Ori_Filesize), 신규 파일에 대한 검증값 (New_Checksum) 및 신규 파일에 대한 사이즈(New_Filesize)는 각각 4바이트로 구성될 수 있다.

델타 바디(220)는 소정 개수의 델타 블록(221)을 포함할 수 있고, 각각의 델타 블록(221)은 원본 파일의 해당 블록과 이에 대응되는 신규 파일의 해당 블록에 대해 차이 알고리즘(Difference Algorithm)을 적용하여 생성 된다.

또한, 각각의 델타 블록(221)은 해당 원본 블록의 식별자인 블록 옵셋(JumpCount), 압축 여부를 나타내는 압축 상태(CompressState), 압축된 파일에 대한 압축 데이터 사이즈(CompressSize), 원본 데이터 사이즈 (OriginalSize) 및 델타 데이터(Delta Data) 정보를 포함한다.

도 4에 도시된 바와 같이 상기 델타 데이터(Delta Data)(223)는 명령어의 시작을 나타내는 시작 플래그 (StartOperand)와, MOD, INS, DEL, EQL, BKT 등과 같은 명령어 코드(OPCODE)와, 명령어 코드가 실행되어야 할 데이터 옵셋(Offset)과, 패치 길이 또는 패치 데이터(Length|Data) 정보를 포함한다.

예를 들어, 상기 시작 플래그(StartOperand)는 'esc'가 될 수 있고, 명령어 코드가 INS 또는 MOD인 경우 상기 델타 데이터는 명령어 코드(OPCODE), 옵셋(Offset), 패치 데이터(Data) 정보를 포함할 수 있고, 명령어 코드가 DEL 또는 BKT인 경우에는 명령어 코드(OPCODE), 옵셋(Offset), 패치 길이(Length) 정보를 포함할 수 있다.

도 5는 본 발명의 일 실시예에 따른 역델타 파일 생성 과정을 나타내는 흐름도이다.

도 5를 참조하면, 먼저 펌웨어 업데이트를 수행할 휴대용 단말기는 소정 원본 파일 블록에 대응되는 델타 블록에 포함된 델타 데이터를 획득한다(단계 510). 여기서, 상기 델타 블록은 휴대용 단말기와 유선 또는 무선으로 연결된 외부 장치(예를 들면, 컴퓨터)로부터 제공될 수 있다.

이후, 휴대용 단말기는 상기 델타 데이터에 포함된 명령어 코드, 옵셋, 패치 길이 또는 패치 데이터를 파악하고(단계 520), 파악된 델타 데이터의 내용에 기초하여 역델타 블록을 생성한다(단계 530).

구체적으로, 휴대용 단말기는 상기 델타 블록에 포함된 명령어 코드에 상응하는 역 명령어 코드를 파악한다 (단계 531). 예를 들어, 델타 블록에 포함된 명령어 코드가 'MOD'인 경우 역 명령어 코드는 'MOD'가 될 수 있고, 델타 블록에 포함된 명령어 코드가 'INS'인 경우 역 명령어 코드는 'DEL'이 될 수 있다. 또한, 델타 블록에 포함된 명령어 코드가 'DEL'인 경우 역 명령어 코드는 'INS'가 될 수 있다.

또한, 휴대용 단말기는 델타 블록에 포함된 옵셋 정보에 기초하여 상기 역명령 코드에 대응되는 원본 파일의 옵셋을 파악하고(단계 533), 델타 블록에 포함된 패치 길이 또는 패치 데이터에 기초하여 복구 길이 또는 복구 데이터를 판단한다(단계 535).

이후, 휴대용 단말기는 생성된 역델타 블록을 버퍼에 임시 저장하고(단계 540), 상기 델타 블록에 기초하여 상기 델타 블록에 대응되는 원본 파일의 해당 블록을 패치한 후(단계 550), 버퍼에 임시 저장된 역델타 블록을 상기 델타 블록이 기록된 위치에 기록한다(단계 560).

도 6a 내지 6c은 본 발명의 일 실시예에 따른 역델타 파일 생성 과정을 설명하기 위한 개념도이다.

도 6a를 참조하면, 델타 파일의 델타 블록 1은 명령어 코드 'MOD'와, 델타 블록 1에 대응되는 원본 파일 블록에 대한 옵셋 및 패치 데이터'11 34'로 구성되고, 휴대용 단말기는 상기 델타 블록 1에 기초하여 역델타 블록 1을 생성하게 된다.

상기 역델타 블록 1은 상기 델타 블록 1의 명령어 코드에 상응하는 역 명령어 코드 'MOD'와, 상기 델타 블록 1의 옵셋과 동일한 옵셋 및 복원 데이터 '12 23'으로 구성된다. 여기서, 상기 복원 데이터는 상기 패치 데이터에 대응되는 원본 파일 블록의 데이터가 될 수 있다.

원본 파일 블록 1은 이에 상응하는 델타 블록 1에 기초하여 신규 파일 블록 1로 패치될 수 있고, 신규 파일 블록 1은 역델타 블록 1에 기초하여 복원 파일 블록 1로 복원될 수 있다. 여기서 원본 파일 블록 1과 복원 파일 블록 1은 서로 동일한 데이터를 가진다.

도 6b를 참조하면, 델타 파일의 델타 블록 2는 명령어 코드 'INS'와, 델타 블록 2에 대응되는 원본 파일 블록에 대한 옵셋 및 패치 데이터'11 34'로 구성되고, 휴대용 단말기는 상기 델타 블록 2에 기초하여 역델타 블록 2를 생성하게 된다.

상기 역델타 블록 2는 상기 델타 블록 2의 명령어 코드(INS)에 상응하는 역 명령어 코드 'DEL'과, 상기 델타 블록 2의 옵셋과 동일한 옵셋 및 상기 패치 데이터 '11 23'에 상응하는 복원 길이 '2'로 구성된다. 여기서, 상기 복원 길이는 상기 명령어 또는 역 명령어 및 상기 패치 데이터에 상응하여 결정될 수 있다.

원본 파일 블록 2는 이에 상응하는 델타 블록 2에 기초하여 신규 파일 블록 2로 패치될 수 있고, 신규 파일 블록 2는 역델타 블록 2에 기초하여 복원 파일 블록 2로 복원될 수 있다. 여기서 원본 파일 블록 2와 복원 파일 블록 2는 서로 동일한 데이터를 가진다.

도 6c를 참조하면, 델타 파일의 델타 블록 3은 명령어 코드 'DEL'과, 델타 블록 3에 대응되는 원본 파일 블록에 대한 옵셋 및 패치 길이'2'로 구성되고, 휴대용 단말기는 상기 델타 블록 3에 기초하여 역델타 블록 3을 생성하게 된다.

상기 역델타 블록 3은 상기 델타 블록 3의 명령어 코드(DEL)에 상응하는 역 명령어 코드 'INS'와, 상기 델타 블록 3의 옵셋과 동일한 옵셋 및 상기 패치 길이 '2'에 상응하는 복원 데이터 '12 23'으로 구성된다. 여기서, 상기 복원 데이터는 상기 명령어 또는 역 명령어 및 상기 패치 길이에 상응하는 원본 파일 블록 3의 데이터에 기초하여 결정될 수 있다.

도 5 및 도 6a 내지 6c에 도시된 바와 같이 본 발명의 일 실시예에 따른 역델타 파일의 생성 방법에서는 델타파일을 구성하는 각각의 델타 블록에 포함된 명령어 코드, 옵셋 및 패치 길이 및 또는 패치 데이터에 기초하여역 명령어 코드, 옵셋 및 복구 길이 또는 복구 데이터로 구성된 역델타 파일을 생성한다.

도 7은 본 발명의 다른 실시예에 따른 역델타 파일 생성 과정을 나타내는 흐름도로서, 휴대용 단말기가 유선 또는 무선 인터페이스를 통해 연결된 외부 장치로부터 펌웨어의 업데이터를 위한 델타 블록을 제공받은 것으로 가정한다.

도 7을 참조하면, 먼저 휴대용 단말기는 소정의 델타 블록에 대응되는 소정의 원본 파일 블록을 버퍼에 임시 저장한 후(단계 710), 상기 소정의 델타 블록에 기초하여 원본 파일 블록을 패치한다(단계 720).

이후, 휴대용 단말기는 패치된 원본 파일 블록(즉, 신규 파일 블록)을 버퍼에 저장된 원본 파일 블록으로 복원

하기 위한 역델타 파일을 생성한다(단계 730).

구체적으로, 휴대용 단말기는 먼저 패치된 원본 파일 블록과 버퍼에 저장된 원본 파일 블록을 선택한 후(단계 731), 선택된 두 파일 블록에 대해 차이 알고리즘(Difference Algorithm)을 적용하여(단계 732), 두 파일 블록 간의 차이에 해당하는 데이터를 델타 버퍼의 해시 테이블에 저장한다(단계 733).

이후, 휴대용 단말기는 압축 여부를 판단한다(단계 734). 즉, 휴대용 단말기는 압축한 경우의 파일 사이즈를 연산하여 압축하지 않은 경우의 파일 사이즈와 비교하고, 압축한 경우의 파일 사이즈가 압축하지 않은 경우의 파일 사이즈보다 작은 것으로 판단되는 경우에는 상기 델타 버퍼의 해시 테이블에 저장된 데이터를 압축하여 (단계 735) 역델타 블록을 생성한다(단계 736).

여기서, 상기 생성된 역델타 블록은 도 3 및 도 4에 도시된 델타 블록과 동일한 구성 즉, 블록 옵셋 (JumpCount), 압축 상태(CompressState), 압축 데이터 사이즈(CompressSize), 원본 데이터 사이즈 (OriginalSize) 및 델타 데이터(Delta Data) 정보로 구성되고, 상기 델타 데이터는 시작 플래그 (StartOperand)와, 패치된 파일 블록을 원본 파일 블록으로 복원하기 위한 명령어 코드(OPCODE), 데이터 옵셋(Offset)과, 패치 길이 또는 패치 데이터(Length|Data) 정보를 포함한다.

이후, 생성된 역델타 블록은 상기 델타 블록이 기록된 위치에 기록된다(단계 740).

도 8은 본 발명의 일 실시예에 따른 역델타 파일을 이용한 펌웨어 복구 방법을 나타내는 흐름도이다. 도 8에 도시된 역델타 파일을 이용한 펌웨어 복구 방법에서는 외부 장치에서 델타 파일을 지정된 패킷 단위로 휴대용 단말기의 지정된 영역에 다운로드한 후 압축 상태(CompressState) 정보에 기초하여 압축된 경우에는 압축을 해제한 것으로 가정한다.

도 8을 참조하면, 먼저 휴대용 단말기는 소정의 델타 블록에 저장된 옵셋 정보에 기초하여 해당 원본 파일 블록에 대한 패치를 수행하여 신규 파일 블록으로 변환한다(단계 810).

이후, 휴대용 단말기는 업데이트가 종료되었는지를 판단하고(단계 820), 업데이트가 종료되지 않은 것으로 판단되면 단계 810으로 되돌아가서 다음 델타 블록의 옵셋 정보에 기초하여 해당 원본 파일 블록에 대한 패치를 수행한다.

단계 820에서 휴대용 단말기는 패치된 원본 파일 블록(즉, 신규 파일 블록)을 포함하는 신규 파일의 사이즈를 델타 파일의 헤더에 포함된 신규 파일 사이즈(New_Filesize)와 비교하여 업데이트 종료 여부를 판단할 수 있다.

또는, 휴대용 단말기는 업데이트가 진행 중인 동안 업데이트 취소를 지시하는 이벤트 신호가 생성되면 업데이트가 종료된 것으로 판단할 수도 있다.

단계 820에서 업데이트가 종료된 것으로 판단되면 휴대용 단말기는 업데이트가 성공적으로 종료되었는지를 판단하고(단계 830), 업데이트가 성공적으로 종료된 것으로 판단되면 본 발명의 일 실시예에 따른 역델타 파일을 이용한 펌웨어 복구 방법을 종료한다.

여기서, 휴대용 단말기는 업데이트된 펌웨어의 유효성(Checksum)을 검증하여 상기 업데이트 성공 여부를 판단할 수 있다.

단계 830에서 업데이트가 성공적으로 수행되지 않은 것으로 판단되면, 휴대용 단말기는 적어도 하나의 역델 타 블록을 포함하는 역델타 파일을 이용하여 패치된 원본 파일 블록(즉, 신규 블록)을 원래대로 복원한다(단 계 840).

단계 840의 역델타 블록에 기초하여 원본 파일 블록으로 복원하는 과정은 단계 810의 델타 블록에 기초하여 원본 파일 블록을 패치하는 과정과 동일한 방법으로 수행될 수 있다.

이후, 휴대용 단말기는 모든 신규 파일 블록에 대한 복원이 종료되었는지를 판단하고(단계 850), 복원되지 않은 신규 파일 블록이 있는 경우에는 단계 840으로 되돌아가서 복원 과정을 계속 수행하고, 모든 신규 파일 블록이 원본 파일 블록으로 복원된 것으로 판단되면 본 발명의 일 실시예에 따른 역델타 파일을 이용한 펌웨어 복구 방법을 종료한다.

도 8에 도시된 역델타 파일을 이용한 펌웨어 복구 방법에서, 펌웨어 업데이트 도중 사용자가 업데이트를 취소하여 펌웨어 업데이트가 성공적으로 수행되지 않은 경우에는 사용자의 복원 지시에 의해 단계 840 및 단계 850을 수행하도록 구성될 수도 있다.

상기한 바와 같은 본 발명의 실시예에서는 휴대용 단말기의 펌웨어를 업데이트하는 것으로 예를 들어 설명하였으나, 휴대용 단말기 이외에도 펌웨어를 구비한 모든 정보 기기에 본 발명의 실시예에 따른 역델타 파일 생성 방법 및 역델타 파이을 이용한 펌웨어 복구 방법이 적용될 수 있음은 물론이다.

이상 실시예를 참조하여 설명하였지만, 해당 기술 분야의 숙련된 당업자는 하기의 특허 청구 범위에 기재된 본 발명의 사상 및 영역으로부터 벗어나지 않는 범위 내에서 본 발명을 다양하게 수정 및 변경시킬 수 있음을 이해할 수 있을 것이다.

특허청구의 범위

청구항 1.

소정의 원본 파일 블록에 대한 수정 작업을 지시하는 명령어 코드와, 데이터 옵셋과, 패치 길이 또는 패치 데이터를 포함하는 적어도 하나의 델타 블록으로 구성된 델타 파일을 이용한 펌웨어 업데이트 방법에 있어서,

상기 델타 블록에 포함된 명령어 코드의 역명령어 코드를 파악하고, 상기 델타 블록의 데이터 옵셋에 기초하여 상기 역명령어 코드의 데이터 옵셋을 파악하고, 상기 델타 블록의 패치 길이 또는 패치 데이터에 기초하여 복구 길이 또는 복구 데이터를 파악하여 역델타 블록을 생성하는 단계;

상기 델타 블록을 이용하여 상기 소정의 원본 파일 블록에 대한 수정 작업을 수행하는 단계; 및

상기 델타 블록이 기록된 위치에 상기 생성된 역델타 블록을 기록하는 단계를 포함하는 역델타 파일 생성 방법.

청구항 2.

제1항에 있어서, 상기 역델타 블록을 생성하는 단계는

상기 생성된 역델타 블록을 버퍼에 임시 저장하는 단계를 포함하는 것을 특징으로 하는 역델타 파일 생성 방법.

청구항 3.

제1항에 있어서, 상기 역델타 블록을 생성하는 단계는

상기 명령어 코드, 상기 역명령어 코드, 상기 델타 블록에 포함된 패치 길이 또는 패치 데이터, 상기 델타 블록에 의해 수정되는 원본 파일 블록의 데이터 중 적어도 하나에 기초하여 상기 복구 길이 또는 복구 데이터가 결정되는 것을 특징으로 하는 역델타 파일 생성 방법.

청구항 4.

적어도 하나의 델타 블록을 포함한 델타 파일을 이용한 펌웨어 업데이트 과정에서 펌웨어 업데이트가 성공적으로 수행되지 못한 경우-업데이트 취소 또는 실패-에 수정된 원본 파일 블록을 복구하기 위한 역델타 파일을 생성하는 방법에 있어서.

소정의 델타 블록에 의해 수정될 소정의 원본 파일 블록을 버퍼에 저장하는 임시 저장 단계;

상기 델타 블록을 적용하여 상기 소정의 원본 파일 블록을 수정하는 패치 단계;

상기 수정된 원본 파일 블록을 원본 파일 블록으로 복원하기 위한 역델타 블록을 생성하는 역델타 블록 생성 단계; 및

상기 패치 단계 및 역델타 블록 생성 단계 완료 후 상기 역델타 블록을 상기 델타 블록이 기록된 위치에 기록 하여 역델타 파일을 생성하는 역델타 파일 생성 단계를 포함하는 역델타 파일 생성 방법.

청구항 5.

제4항에 있어서, 상기 역델타 블록 생성 단계는,

상기 수정된 원본 파일 블록과 상기 버퍼에 저장된 원본 파일 블록에 대해 차이 알고리즘을 적용하여 상기 역 델타 블록을 생성하는 것을 특징으로 하는 역델타 파일 생성 방법.

청구항 6.

제1항 또는 5항에 기재된 역델타 파일 생성 방법에 의하여 역델타 파일을 생성하면서 델타 파일을 이용하여 펌웨어를 업데이트 하는 과정에서,

펌웨어 업데이트의 진행 중 또는 펌웨어 업데이트의 종료의 경우에, 펌웨어 업데이트 취소 또는 실패 여부를 판단하는 단계; 및

펌웨어 업데이트가 취소 또는 실패인 경우, 상기 역델타 파일을 이용하여 원본 파일을 복구하는 단계를 포함 한 역델파 파일을 이용한 펌웨어 복구 방법.

청구항 7.

제6항에 있어서, 상기 펌웨어 업데이트 취소 또는 실패 여부를 판단하는 단계에서,

상기 펌웨어 업데이트의 실패 여부 판단은, 업데이트된 펌웨어의 유효성을 검증하여 상기 펌웨어 업데이트의 실패 여부를 판단하는 것을 특징으로 하는 역델타 파일을 이용한 펌웨어 복구 방법.

도면의 간단한 설명

도 1은 종래의 휴대용 단말기의 펌웨어 업데이트를 위한 시스템의 구성을 나타내는 블록도이다.

도 2는 본 발명의 일 실시예에 따른 역델타 파일을 이용한 펌웨어 복구를 수행하는 장치의 구성을 나타내는 블록도이다.

도 3은 본 발명의 일 실시예에 따른 델타 파일의 구성을 나타내는 블록도이다.

도 4는 도 3에 도시된 델타 데이터의 포맷을 나타내는 블록도이다.

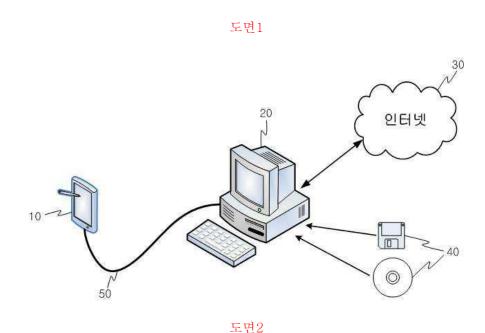
도 5는 본 발명의 일 실시예에 따른 역델타 파일 생성 과정을 나타내는 흐름도이다.

도 6a 내지 6c는 본 발명의 일 실시예에 따른 역델타 파일 생성 과정을 설명하기 위한 개념도이다.

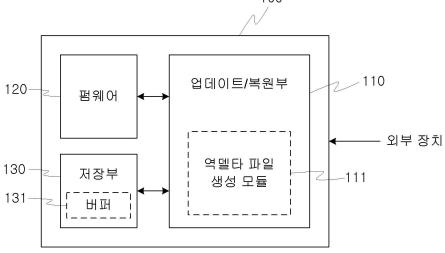
도 7은 본 발명의 다른 실시예에 따른 역델타 파일 생성 과정을 나타내는 흐름도이다.

도 8은 본 발명의 일 실시예에 따른 역델타 파일을 이용한 펌웨어 복구 방법을 나타내는 흐름도이다.

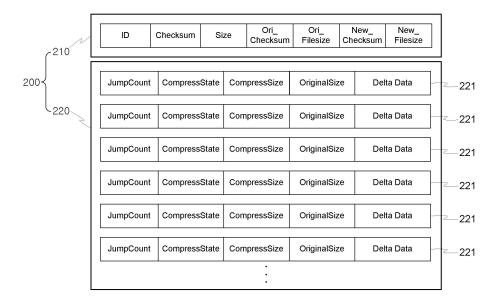
도면



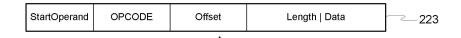
100



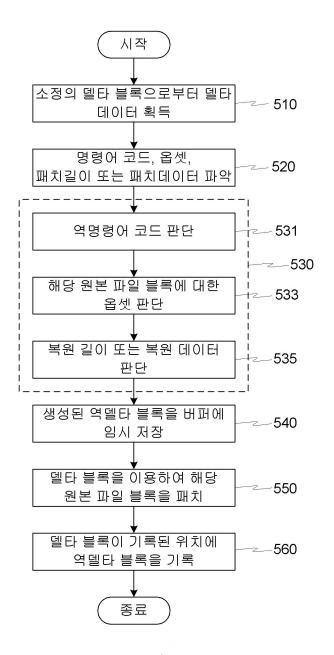
도면3



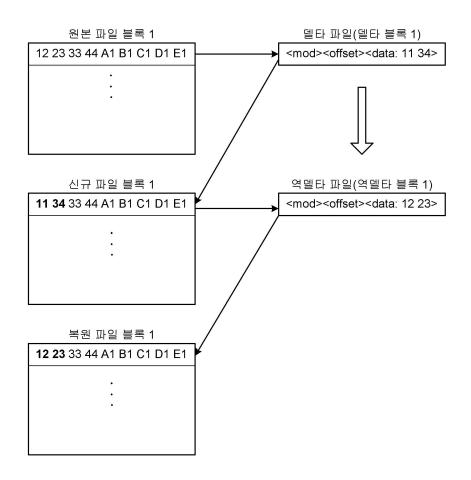
도면4



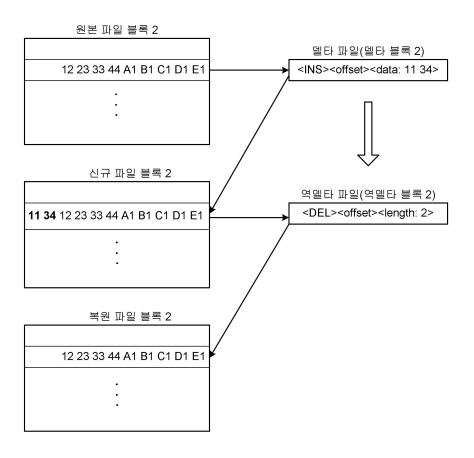
도면5



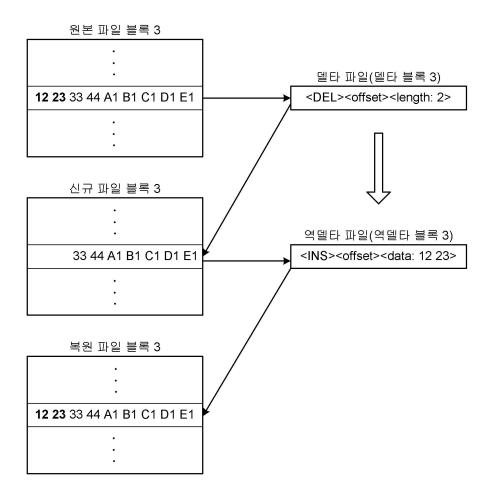
도면6a



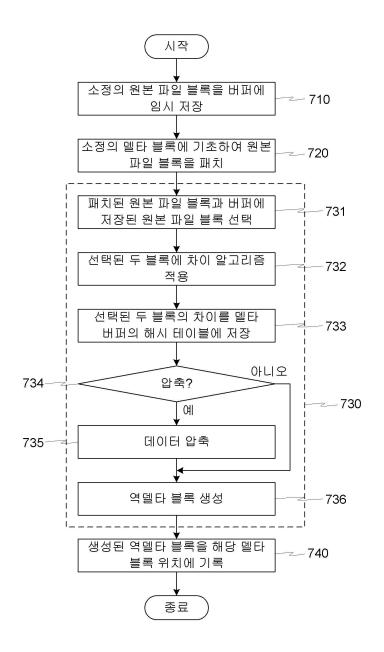
도면6b



도면6c



도면7



도면8

