(19)대한민국특허청(KR)
(12) 공개특허공보(A)

(51) 。Int. Cl.[7]
H04L 12/24
H04L 12/28
H04L 29/06

(11) 공개번호 10-2005-0062766
(43) 공개일자 2005년06월27일

(21) 출원번호 10-2005-7002135
(22) 출원일자 2005년02월04일
번역문 제출일자 2005년02월04일
(86) 국제출원번호 PCT/IB2003/003308
국제출원일자 2003년07월24일
(87) 국제공개번호 WO 2004/015929
국제공개일자 2004년02월19일

(30) 우선권주장 0218174.1 2002년08월06일 영국(GB)
0309400.0 2003년04월25일 영국(GB)

(71) 출원인 코닌클리케 필립스 일렉트로닉스 엔.브이.
네델란드왕국, 아인드호펜, 그로네보드스베그 1

(72) 발명자 블락웰 로빈 제이.
영국, 수레이 알에치1 5에치에이, 레드힐, 크로스 오크 레인, 필립스 인
텔렉추얼 프로퍼티 앤 스탠다즈
한킨 네일 에이.
영국, 수레이 알에치1 5에치에이, 레드힐, 크로스 오크 레인, 필립스 인
텔렉추얼 프로퍼티 앤 스탠다즈
라니간 피터 제이.
영국, 수레이 알에치1 5에치에이, 레드힐, 크로스 오크 레인, 필립스 인
텔렉추얼 프로퍼티 앤 스탠다즈
쉬펄드 니콜 비.
영국, 수레이 알에치1 5에치에이, 레드힐, 크로스 오크 레인, 필립스 인
텔렉추얼 프로퍼티 앤 스탠다즈
루드랜드 필립 에이.
영국, 수레이 알에치1 5에치에이, 레드힐, 크로스 오크 레인, 필립스 인
텔렉추얼 프로퍼티 앤 스탠다즈

(74) 대리인 정상구
신현문
이범래

심사청구 : 없음

(54) 네트워크 설정 및 관리 프로토콜

명세서

기술분야

본 발명은 네트워크 프로토콜에 관한 것으로, 특히 프로토콜의 구현들에 관한 것이다.

배경기술

네트워크 관리에 대한 종래 기술 프로토콜은 대역폭, 배터리 소비, 및 비용의 정도가 문제가 되지 않는, 인터넷 애플리케이션들에 유용한 유니버설 플러그 앤 플레이(UPnP)이다. 소비자 전자제품들(CE)에서의 프로토콜의 구현들은 존재하지만, 프로토콜의 한도 때문에, 그러한 구현들은 그렇지 않다면 최소 처리 능력만을 요구하는 가장 간단한 디바이스들에 특히 과부하(heavy load)를 부과한다.

따라서, 라이트들, 서모스탯들, 및 CE 장비(TV, DVD 및 PVR을 위한 원격 제어)와 같은 간단한 디바이스들에 내장하기에 적절한 프로토콜, 즉 구현하기에 심플하고 비용효율이 높고, 최소 대역폭을 요구하면서, 능력 변경으로 디바이스들의 범위에 걸쳐 스케일러블(scalable)한 프로토콜에 대한 필요성이 존재한다.

이 필요성은 무선 애플리케이션에 대해 제한되지 않고, 유선 애플리케이션들로 확장한다.

발명의 상세한 설명

본 발명의 제 1 양상에 따라, 네트워크된 디바이스를 작동하는 방법이 제공되고, 상기 방법은, 토큰-압축된(token-compression) 메시지들을 수신하는 단계; 네트워크된 디바이스로부터 간단한 디바이스 기술 응답(simple device description response)을 요구하는 인입 간단한 디바이스 기술 질의 메시지들을, 인입 메시지들을 압축해제(decompress)하지 않고 수신된 토큰-압축된 메시지들에서 인식하는 단계; 및 간단한 디바이스 기술 응답을 요구하는 인입 디바이스 질의 메시지에 대한 응답으로서 디바이스 타입을 포함하는 간단한 디바이스 기술을 전송하는 단계를 포함한다.

이러한 방법은 이 특허 출원의 목적인 프로토콜을 구현한다. 프로토콜 자체는 홈 유니폼 제어 언어(home uniform control language; HUCL)라 호칭될 것이다.

간단한 디바이스들이 압축된 토큰-코딩된 메시지들상에서 직접적으로 작동 가능하도록 하는 어빌리티는 간단한 디바이스들이 과도한 하드웨어 또는 소프트웨어 요구조건들 없이 이루어지는 것이 가능하도록 허용한다. 더 복잡한 디바이스들은 메시지들을 압축해제할 수 있고 더 복잡한 방식들로 그것들상에서 작동할 수 있다. 따라서, 본 발명에 따른 접근법은 간단한 디바이스들이 어려움이 없이 간단한 디바이스들을 오버로딩하지 않고 복잡한 디바이스들의 전체 기능성을 허용하면서, 통합된 네트워크에서 복잡한 디바이스들과 결합되도록 허용한다. 간단한 디바이스들은 확장된 디바이스 기술 질의들을 간단하게 무시할 수 있다.

간단한 디바이스 기술은 각각의 필드가 고정된 길이로 이루어진 미리 결정된 필드들의 적은 또는 적당한 수를 포함한다. 일반적으로, 동일한 필드들은 일부 변 화가 존재할 수 있음에도, 각각의 메시지들에 대해 사용될 것이다. 예컨대, 합성 디바이스는 아래에 설명된 서브-디바이스들의 수를 포함하는 부가적인 정수 필드(integer field)를 포함할 수 있다.

간단한 디바이스 기술 메시지는 인간-판독가능한 메시지 포맷으로부터 압축된 토큰-압축된 메시지 형태로 이루어진다. 바람직하게, 메시지는 다른 디바이스의 타입을 나타내는 디바이스 타입 값을 포함하고; 디바이스 타입 값은 제어기 디바이스 타입 및 기본 디바이스 타입(basic device type)을 포함하는 미리 결정된 최상위 레벨 요소(top level element)들을 갖고, 기본 디바이스에 종속되고 보조(subsidiary) 디바이스 타입이 종속된 상위 레벨 디바이스 타입들의 성질들을 승계하는 보조 디바이스 타입들의 적어도 하나의 다른 레벨을 갖지만, 제어기 디바이스 타입에 종속되는 보조 디바이스 타입들의 어떠한 다른 레벨도 포함하지 않는 디바이스 타입 계층구조로부터 선택된다.

HUCL 프로토콜에 따라, 아래 놓인 메시지 포맷은 XML과 같은 인간 판독가능한 포맷이다. 그러나, 대역폭을 절약하기 위해, 메시지들은 네트워크된 디바이스들간에서 압축된 형태로 전달된다. 사용된 압축 방법이 공통 스트링들을 토큰들로 대체하는 토큰 압축이기 때문에, 네트워크된 디바이스들은 그럼에도 불구하고, 이러한 압축된 메시지들을 처리 가능하다. 따라서, 네트워크된 디바이스는 압축해제 없이 압축된 토큰들을 인식할 수 있고, 적어도 간단한 디바이스 기술의 응답을 요구하는 질의를 인식하기에 충분하며, 그 다음에 간단한 디바이스 기술로 응답한다.

토큰 코딩의 적절한 형태는 http://www.w3.org/TR/wbxml에서 이용가능한, 1999년 6월 24일자 "wap binary XML content format"에서 설명된다.

기본 디바이스 타입에 종속된 적어도 하나의 계층, 즉 제어되는 디바이스들의 계층이 존재하지만, 제어기 디바이스들의 계층에 대응하는 것은 존재하지 않는다. 이는 간단한 디바이스 기술 메시지들을 최대한 짧고 간단하게 유지하기 위한 것이고 - 유니버설 원격 제어와 같은 많은 제어기들은 다수의 상이한 디바이스 타입들을 제어가능하다.

제 2 양상에서, 본 발명은 네트워크된 디바이스에 관한 것이고, 네트워크된 디바이스는 : 토큰-압축된 인간 판독가능한 메시지들을 전송하고 수신하는 트랜시버; 및 네트워크된 디바이스로부터 간단한 디바이스 기술을 요구하는 수신된 디바이스 질의 메시지들을, 인입 메시지들을 압축해제하지 않고 인식하는 단계, 및 간단한 디바이스 기술 응답을 요구하는 인입 디바이스 질의 메시지에 대한 응답으로서 디바이스 타입을 포함하는 간단한 디바이스 기술을 트랜시버를 통해 전송하는 단계의, 인입 토큰-압축된 인간 판독가능한 메시지들에 관한 단계들을 수행하도록 배열된 메시지 핸들러를 포함한다.

네트워크된 디바이스는 인간 판독가능한 포맷으로부터 미리 압축된 미리 결정된 간단한 디바이스 기술 메시지를 더 포함할 수 있고, 여기서 메시지 핸들러는 메모리로부터 미리 결정된 간단한 디바이스 기술 메시지를 판독하고, 인입 디바이스 질의 메시지에 대응하여 그것을 트랜시버를 통해 전송하도록 배열된다.

그 이상의 양상에서, 본 발명은 시스템에 관한 것이다. 시스템은:

네트워크 메시지들을 전송 및 수신하는 트랜시버를 각각 갖는 복수의 네트워 크된 디바이스들; 간단한 디바이스 질의 메시지를 다른 디바이스들에 전송하고, 다른 디바이스들로부터 이후 수신된 간단한 디바이스 기술 메시지들을 수신 및 해석하도록 배열된 적어도 하나의 네트워크된 디바이스; 디바이스의 타입을 나타내는 디바이스 타입 값을 포함하는 규정된 길이의 간단한 디바이스 기술 메시지를 전송함으로써, 디바이스들 중 다른 것들로부터의 인입 간단한 디바이스 질의 메시지에 응답하도록 배열된 네트워크된 디바이스들 각각을 포함하고; 복수의 네트워크된 디바이스들은 메시지들을 압축해제하

기 위한 능력이 없고 압축된 메시지들을 직접적으로 해석하는 적어도 하나의 간단한 디바이스, 및 메시지들을 압축해제하기 위한 메시지 압축해제 배열 및 압축해제된 메시지들을 해석하기 위한 메시지 해석기를 포함하는 적어도 하나의 복잡한 디바이스를 포함한다.

도면의 간단한 설명

도 1은 본 발명에 다른 디바이스들의 쌍을 포함하는 시스템을 도시하는 도.

도 2는 1개의 디바이스내의 소프트웨어의 개략도.

도 3은 디바이스 디스커버리 프로세스의 흐름도.

도 4는 디바이스 타입 계층구조(hierarchy)의 개략도.

도 5는 제어되는 디바이스에 그 디바이스의 제어기의 제어 능력(capability)을 제어기가 통지하기 위해 수행하는 단계들을 도시하는 도.

도 6은 제어되는 디바이스의 제어기의 제어 능력을 제어기가 결정하기 위해 수행하는 단계들을 도시하는 도.

도 7은 소프트웨어의 구조를 도시하는 도.

도 8은 HUCL 프로토콜을 예시하는 도.

도 9는 간단한 디바이스 기술 메시지를 예시하는 도.

실시예

본 발명의 더 나은 이해를 위해, 실시예들이 첨부 도면들을 참조하여 전적으로 예로서만 이제 기술될 것이다.

프로토콜 HUCL은 무선 시스템들을 위해 본래 설계된 낮은 대역폭 제어 프로토콜인 라이트웨이트(lightweight)다. 메시징 포맷은 XML에 기초하고, 메시지들은 전송 이전에 압축된다. XML의 사용은 전송되는 데이터를 감소시키는 압축을 갖는 확장가능하고 스케일러블한 솔루션을 제공하여, 송신기가 온(on)되고 전력을 소비하는 시간량을 감소시킨다.

HUCL 프로토콜의 일반적인 원리들 및 디바이스상에서 어떻게 작동하는지는 간단한 예를 참조하여 이제 기술될 것이다.

도 1을 참조하여, 라이트 스위치(2) 및 라이트 설비(light fitting)(4)가 제공되고, 후자는 본 발명의 실시예이다. 라이트 스위치(2)는 제어 회로(12) 및 메모리(14)와 함께, RF 트랜시버(8) 및 배터리(10)와 함께, 사용자에 의해 작동되는 물리적인 락커 스위치(6)를 갖는다. 라이트 설비 또한 RF 트랜시버(8) 및 메모리(14)를 갖지만, 라이트 설비는 메인 전원 공급되고, 전구(22)에 전원 공급하기 위해 제어 회로(20)를 갖는다. 따라서, 라이트 스위치(2)는 제어 입력(6)(스위치)을 갖는 제어기의 예이고, 반면에 라이트 설비는 제어되는 디바이스(4)의 예이다. 제어기내의 메모리(14)는 제어기가 제어할 수 있는 디바이스 타입들, 및 디바이스 타 입들에 속하는 제어 기능들의 목록(24)을 포함한다. 제어되는 디바이스(4) 및 제어기(2) 둘 다내의 메모리(14)는 제어 회로가 아래 더 상세히 기술될 방법들을 수행하도록 하는 코드(26)를 또한 포함한다.

도 2는 각각의 디바이스들상의 메모리(14)내에 존재하는 소프트웨어의 리프리젠테이션(representation)이다. 제어 애플리케이션(30)은 일정 이벤트들이 발생할 때 HUCL 소프트웨어 스택(32)과 통신한다.

유사한 방식으로, HUCL 소프트웨어 스택(32)은 RF 소프트웨어 스택(34)과 통신하고, RF 소프트웨어 스택(34)은 예컨대, 데이터의 수신과 같은 일정 이벤트들이 발생할 때 HUCL 소프트웨어 스택(32)과 되돌려 통신할 것이다.

메시지들(36)은 전송되고 수신된다. 메시지들은 간단한 디바이스 기술 질의 메시지, 또는 다수의 어떤 다른 메시지 타입들을 포함하는 다수의 타입들일 수 있다.

라이트 설비(4)의 메모리(14)는 간단한 디바이스 기술을 라이트 설비(4)로부터 요청하는 질의가 수신될 때, 간단히 전송될 수 있는 미리 압축된 간단한 디바이스 기술 메시지를 포함한다. 라이트 설비내의 제어 애플리케이션은 트랜시버(8)로부터 수신된 그러한 인입 간단한 디바이스 기술 질의 메시지들을 그것들을 압축해제하지 않고 인식 가능하다. 이는 전송되는 메시지들을 압축하기 위한 토큰-압축의 사용 때문에 가능하다.

디바이스들의 작동이 도 3을 참조하여 이제 기술될 것이다. 디바이스들의 상기 쌍의 작동의 제 1 페이스(phase)는 스위치(2)(제어기)가 라이트 설비(4)(제어 되는 디바이스)의 어드레스를 디스커버(discover)하는 것에 대한 것이다. 이는 디바이스 디스커버리(device discovery)로 알려진 것이고, 이는 디바이스 디스커버리가 제공되거나(RF 소프트웨어 스택내에), 또는 RF 전송 스택의 상부에(HUCL 소프트웨어 스택의 하위 계층에) 디바이스 디스커버리가 구현 가능한, 아래 놓인 RF 전송 스택의 요구조건이다.

디스커버리 프로세스는 알려진 디바이스들의 수를 처음 요청하고, 그 디바이스들의 네트워크 어드레스들을 그 다음에 요청하는 호출(call)을 HUCL 소프트웨어 스택에 행함으로써 제어기내의 제어 애플리케이션에 의해(일부 사용자 상호작용의 결과로서 가능한) 개시된다(100). 이들 디바이스 어드레스들은 리턴된다.

아래 놓인 RF 프로토콜에 따라, 네트워크 어드레스들은 일부 다른 방식으로 설정될 수 있다.

디바이스 디스커버리 페이스의 최종 결과는 제어 애플리케이션에 RF 스택에 알려진 모든 디바이스들의 어드레스들의 목록이 공급되는 것이다(102). 프로세스의 이 포인트에서, 제어 애플리케이션은 각각의 다른 디바이스의 어드레스 외에 각각의 다른 디바이스에 관해 아는 것이 없다.

쌍을 이루는 프로세스(pairing process)내의 제 2 페이스는 제어 애플리케이션이 어드레스들을 가지고 있는 디바이스들에 관한 정보를 모으는 것에 대한 것이다. 이 정보는 디바이스 기술이라 호칭된다. 제어 애플리케이션은 HUCL 소프트웨어 스택에 호출을 행하고, 디바이스 기술이 요구되는 디바이스의 어드레스를 전달함으로써 이를 이루어낸다.

그 다음에, 디바이스 기술에 대한 요청이 RF 링크를 통해 목적 디바이스에 전달되어(104), 상술된 스위치/설비 예에서, 요청이 스위치로부터 설비로 전송된다. 요청을 수신할 시에, 목적 디바이스의 HUCL 소프트웨어 스택은 제어 애플리케이션에 디바이스 기술을 요청하는 호출을 행한다. 기술의 포맷은 규정되어 있다. 미리 압축된 형태가 아니라면, 기술은 요청의 발신자에 되돌려 전송되기 전에 압축된다.

요청하는 디바이스상의 HUCL 소프트웨어 스택이 디바이스 기술을 수신할 때(106), 디바이스 기술은 제어 애플리케이션까지 전달된다. 이 포인트에서, 애플리케이션은 디바이스에 관한 일정 기본 정보를 가지고 있고, 이 디바이스와 그 이상의 통신을 해야하는가에 대한 결정을 이루어낼 수 있다.

HUCL의 설계 목표는 간단한 디바이스들상에서 작동하기에 적절한 것이지만, 디바이스를 완전히 기술하기 위해 필요한 정보는 꽤나 복잡할 가능성이 있다. 아래의 목록은 디바이스가 그것의 기술의 부분으로서 제공하기를 원할 수 있는 정보의 소트(sort)를 도시한다.

Device Type e.g. DVD

Vendor Name e.g. Philips

Model Number e.g. DVD1010/002

Serial Number e.g. AH6848032345

Vendor URL e.g. www.philips.com

이 섹션 전체를 통한 예에서 사용된 스위치와 같은, 제어 디바이스들의 가장 심플함을 위해, 이 정보의 대부분은 대체로 불필요하다. 그러나, 이는 이러한 정보가 사용자에 디스플레이될 수 있는 스크린을 갖는 상위 엔드 'PDA' 타입 원격 제어기에서 사용될 수 있다.

라이트 설비(4)와 같은 하위-엔드 디바이스들상에서의 이러한 기술들의 처리는 기술이 수신되면서 완전한 메시지를 캐시하기 위한 저장매체가 필요할 수 있으므로, 문제점을 보일 수 있다. 상기 도시된 기술 데이터의 전체 크기가 불명확하고, 대부분의 정보는 '프리 텍스트(free text)'이고; 벤더 이름이 길 수 있으며, URL이 예컨대,

http://www.consumer.philips.com/global/b2c/ce/catalog/subcategory.jhtml?groupId=VIDEO&divId=0&catId=DVD&subCatId=DVDPLAYER와 같이 파라미터들을 갖는 정확한 페이지를 명기할 수 있으므로, 문제점은 처음 짐작할 수 있는 것보다 안좋다.

HUCL에서 이를 극복하는 방식은 디바이스 정보가 정보의 2개의 타이어들(tiers)로 분리되는 것이다. 제 1 타이어는 디바이스의 심플화한 기술이지만, 그 이상의 정보가 이용가능하면 식별한다. 제 1 타이어는 어떤 프리 텍스트 필드들을 포함하지 않아서, 그것의 전체 길이는 명확하다. 확장된 정보의 제 2 타이어는 선택적이지만, 부가 정보를 제공한다.

도 9를 참조하여, 간단한 디바이스 기술 메시지(230)는 디바이스 타입(232), 확장된 디바이스 기술이 이용가능한지 여부를 표시하기 위한 필드(238), 및 예컨대, 이벤트 서브스크립션(event subscription)이 이용가능한지 여부를 표시하기 위한 플래그(flag)와 같은 키 정보를 식별하는 다른 필드들(236)을 필드들로서 포함한다. 선택적인 정수 필드(234)는 합성 디바이스의 서브-디바이스들의 수를 나타낸다. 당업자는 메시지(230)가 심플화를 위해 생략된 헤더(header) 및 푸터(footer)를 또한 포함할 수 있다는 것을 알 것이다. 메시지는 명료함을 위해 마찬가지로 생략된 압축된 XML 토큰들을 포함할 것이다. 간단한 디바이스 기술의 필드들은 모두 고정된 길이여서, 압축해제 없이 쉽게 다루어질 수 있다.

간단한 디바이스 기술(230)을 수신한 이후에(도 3), 간단한 디바이스 기술(230)은 HUCL 스택으로 되돌려 전달된다.

현재 예에서, 라이트 설비(4)는 이용가능한 확장된 디바이스 기술을 가지고 있지 않고, 이는 간단한 디바이스 기술(230)에서 플래그에 의해 표시된다.

그러나, 다른 디바이스가 확장된 디바이스 기술이 이용가능하다고 조사되고(polled), 제어기 디바이스가 그것을 요구하면, 제어기 디바이스 제어 애플리케이션은 "GetExtendedDescription" 요청(108)을 그 디바이스에 되돌려 송신(issue)할 수 있다.

이 요청을 수신하는 디바이스상의 HUCL 스택은 확장된 디바이스 기술을 요청하는 GetExtendedDescription 호출을 제어 애플리케이션에 행한다.

확장된 디바이스 기술은 HUCL 스택에 되돌려 전달되고, 그것을 요청하였던 디바이스상의 제어 애플리케이션으로 되돌아간다. 그 다음에, 확장된 기술은 요청하는 디바이스로 리턴된다(110).

GetExtendedDescription 질의가 확장된 디바이스 기술을 제공하지 않는 디바 이스상에서 수신되면, 요청은 간단히 무시된다.

이 섹션 전체를 통해 사용된 스위치/설비 예로 다시 돌아가면, 스위치가 설비의 어드레스만을 아는 포인트에서부터, 스위치는 설비의 간단한 디바이스 기술을 설비에 요청한다. 이를 수신할 시, 간단한 디바이스 기술은, 스위치가 표준 설비 명령 세트에 순응하는 라이트 설비와 토킹하는 중임을 스위치가 알고, 라이트 설비(4)가 어떤 확장된 디바이스 기술을 제공할 수 없음을 또한 스위치가 알도록 충분한 정보를 제공한다.

디바이스 애플리케이션은 간단한 디바이스 기술이 요청되었을 때, HUCL 스택에 간단한 디바이스 기술을 제공하는 것은 의무적이다. 어떤 확장된 디바이스 기술을 제공하지 않는 디바이스는 이러한 정보에 대해 수신한 어떤 요청도 무시할 수 있다.

디바이스에 의해 리턴된 간단한 디바이스 기술내에 포함된 것은(요청되었을 때) 예컨대, TV, DVD, 라이트 설비 등과 같은 디바이스의 타입을 식별하는 디바이스 타입 필드(232)다. 디바이스 타입 필드(232)는 디바이스가 순응하는 명령 세트를 제어기(간단한 디바이스 기술을 요청하는)에 식별할 것이다. HUCL 디바이스들은 그것들의 식별자에 의해 간단하게 자신들을 식별하고, 그 다음에 HUCL 디바이스들은 자신들이 어떻게 제어되는지 기술하기 위한 메시지들을 전송하려 하지 않으며; HUCL에는 '런타임(run time)' 서비스 기술 컨셉트(concept)가 존재하지 않는다. 디바이스가 라이트 설비로서 자신을 식별한다면, 이 디바이스상에서 호출될 수 있는 명령 세트는 라이트 설비 타입 디바이스에 대한 HUCL 명세에서 식별된다.

도 4를 참조하여, 모든 디바이스 타입들은 베이스 디바이스 타입(50)에 종속된다. 이 예에서, 최상위 레벨 요소들(58)은 제어기 디바이스 타입(52), 제어되는 디바이스들에 대한 베이스 디바이스 타입(54), 및 알람 디바이스 타입(56)을 포함한다.

보조 디바이스 타입들(68)은 기본 디바이스 타입에 종속된다. 예에서, 이들은 TV 디바이스 타입(64), 디머블(dimmable) 라이트 디바이스 타입(62), 및 PVR 디바이스(60)을 포함한다.

디바이스 타입 분류는 간단한 제어기가 제어기의 능력의 한도까지 디바이스를 제어할 수 있는지 여부를 식별하도록 허용하는 시스템을 제공하는 것이다.

간단한 스위치는 라이트를 턴 온 및 오프하기 위해 라이트 설비와 한 쌍일 수 있고, 스위치의 제어 기능, 즉 디바이스를 턴 온 또는 오프하는 능력이 예컨대, TV, 히터, 프린터와 같은 온/오프 컨셉트를 수용할 수 있는 어떤 디바이스에 적용가능해야 한다고 할 지 모른다.

이것이 구현될 수 있는 한가지 방식은 어떻게 제어하는지(턴 온 또는 오프) 알고있는 모든 디바이스들의 목록을 스위치가 갖도록 하여, 스위치가 디바이스에 대한 간단한 디바이스 기술을 요청할 때, 스위치는 리턴된 기술내의 디바이스 타입 필드를 보고 스위치가 어떻게 제어하는지 알고있는 디바이스 타입들의 그것의 목록내에 디바이스가 존재하는지 여부를 결정할 수 있다.

이 접근법의 2개의 중요한 결점들이 존재한다. 우선, 스위치는 간단한 디바이스이고, 그것이 제어할 수 있는 방대할 수 있는 모든 가능한 디바이스들의 목록 을 유지해야만 하는 것은 스위치와의 애플리케이션에 바람직하지 않고; 다음으로, 스위치가 생산된 후에(간단한 온 오프 기능을 수용할 수 있는) 디바이스의 새로운 타입이 생성된다면, 스위치는 그것의 목록에 새로운 디바이스 타입을 가지지 않을 것이고, 스위치가 그 디바이스를 제어할 수 없고, 즉 확장가능하지 않다.

HUCL은 도 4에 도시된 계층 방식으로 디바이스들을 분류한다. 디바이스 타입 필드(232)(도 9)는 계층내에서 디바이스를 식별하여서, 새로운 디바이스들이 생성되었다 할지라도, 그것이 계층내의 적절한 포인트로부터 유도되는 한, 간단한 스위치는 그것을 한도까지 제어할 수 있다는 것을 여전히 알 수 있다.

트리내의 하위 위치에 있는 디바이스들은 그 위의 디바이스 타입들의 기능을 승계한다. 트리내의 하위 디바이스들에 적용될 때, 일부 해석을 명령들에 대해 추가할 필요가 있을 수 있고, 이는 예컨대, 라이트에 전송될 때 온/오프 명령은 그것을 명백하게 턴 온 및 오프할 것이지만, TV에 동일한 명령들이 전송될 때, TV를 스탠바이 모드(standby mode)에 진입하거나 또는 진출하도록 할 수 있다.

디바이스 타입 기술의 핵심 이점은, 제어기가 특정 디바이스 타입에 대한 지식이 자체내에 없을지라도, 제어기가 일부 지식을 가질 수 있는 특정 디바이스가 유도된 곳으로부터 제어기는 디바이스를 결정할 수 있고, 따라서 디바이스를 더 적은 한도까지 제어가능할 수 있다(디바이스의 관점(perspective)으로부터).

예컨대, 라이트 스위치가 디바이스의 어드레스를 얻는 경우를 고려하면, 라이트 스위치는 디바이스로부터 간단한 디바이스 기술을 요청하고; 디바이스 타입 필드는 디바이스를 TV로서 식별하지만, 스위치는 그것이 알고있는 디바이스로서 TV를 인식하지 않는다. 그러나, 스위치는 그것이 스위치가 알고있는 '기본 디바이스'의 유도체(derivative)임을 기술로부터

또한 설정할 수 있다. 결론은 디바이스 자체에 관해서는 아는 것이 없더라도, 스위치가 TV를 제어기의 능력, 즉 온 및 오프의 한도까지 제어할 수 있다는 것이다. 디바이스는 스위치가 제조되고나서 오랜 후에 발명된 'XYZ'라 호칭되는 디바이스의 새로운 분류일 수 있지만, 디바이스가 기본 디바이스로부터 유도된 한, 스위치는 여전히 그것을 한도까지 제어할 수 있다.

디바이스 타입 계층이 2개의 타이어들, 및 제어기 및 기본 디바이스 최상위 레벨 요소들만을 가질 수 있더라도, 적어도 하나의 그 이상의 타이어 및/또는 최상위 레벨 요소가 요망된다. 위에 도시된 기본 디바이스의 기능에 응하지 않을 수 있는 디바이스들의 이 케이터(cater)들은 알람과 같은 기본 '턴 온''턴 오프' 기능을 갖지 않는 디바이스들이다. 예시 목적을 위해 '알람' 타입 디바이스(56)가 도 4에 도시되어 있고, 이 '알람' 디바이스는 기본 디바이스로부터 유도된 디바이스들이 가지고 있어야만 하는 노멀(normal) 온/오프 기능들을 구현하기를 원하지 않음이 분명하고; 따라서, 그것은 계층내의 기본 디바이스(54) 자체와 동일한 최상위 레벨(58)에 자리잡는다.

계층구조에 대한 제 2 확장이 도 4에 또한 도시되어 있고, 이는 즉, 노멀 TV 디바이스(64) 아래의 향상된 TV 디바이스(66)이다. 여기서, 향상된 TV 디바이스는 기본 디바이스(54)와 TV 디바이스(64) 둘 다의 모든 기능성을 승계하지만, 노멀 TV에 없는 일부 확장된 기능성을 또한 포함한다. 노멀 TV 디바이스를 작동시키도록 설계된 보통의 TV 원격 제어는 향상된 TV 디바이스를 노멀 TV 디바이스 기능성의 레벨까지 작동시킬 수 있지만, 확장된 기능성을 제어할 수는 없다.

HUCL 프로토콜은 따라서, 디바이스 타입 및 그것이 기능성을 승계한 그것의 위의 디바이스들을 기술하는 확장가능한 메카니즘을 제공한다. 다수의 층들의 계층구조의 아이디어(idea)가 매력이 있는 듯 하지만, 3 또는 4 레벨들 너머로 확장하는 것은 간단한 디바이스 기술의 크기에 영향을 주기 시작할 것이다.

HUCL내에서, 제어가능 디바이스 뿐만아니라 제어기로부터 디바이스 기술을 요청하는 것이 가능하다. 1개의 디바이스가 "GetSimpleDescription"을 제어기 디바이스(예컨대, 스위치)에 전송할 때, "제어기"의 디바이스 타입을 포함하는 간단한 디바이스 기술이 리턴된다. 제어기 디바이스는 또한, 제조자, 모델 번호 등과 같은 그 이상의 정보를 제공하는 확장된 디바이스 기술을 이용가능하게 할 수 있다.

제어기에 의해 리턴되는 디바이스 타입은 간단하게, 디바이스 타입 트리내에서 규정된 상이한 제어기 타입 디바이스들의 계층이 없는 "제어기"(52)임을 주지하는 것은 중요하다. 이에 대한 이유는 프로토콜 및 메시지들의 크기들을 작고 간단하게 유지하기 위해 다시 시도하는 것이다. 스위치, TV 원격 제어, PVR 원격 제어 등과 같은 기본 제어기로부터 유도된 상이한 제어기 타입들을 갖는 것이 가능할 수 있다고 여겨질 수 있다. 그러나, 광범위한 디바이스들을 제어 가능한 유니버설 원격 제어기(Universal Remote Controller)와 같은 인텔리전트 제어기(intelligent controller)와 함께 문제점이 발생될 수 있다. 간단한 디바이스 기술내에 모든 가 능한 제어기 타입들을 포함하는 것은 최초의 간단한 디바이스 기술을 간단하게 하기 위해 시도하는 목표에 대립하는 잠재적으로 큰 메시지를 초래할 수 있다. 제어기 디바이스의 정확한 능력을 결정하기 위해 상이한 메카니즘들이 이용된다.

제어기 디바이스의 능력을 결정하는 제 1 수단은 제어기 디바이스에서 허가되고, 예컨대, "유니버설 원격 제어"와 같은 디바이스 명칭과 같은 정보를 포함할 수 있으며, 텍스트 정보이고 애플리케이션 소프트웨어에 의해 직접적으로 해석가능하지 않으면서 제어기에 관한 통지된 선택을 행하도록 돕기 위해 사용자에 제공될 수 있는 확장된 디바이스 기술에 의한 것이다.

디바이스가 제어기에 관해 더 결정하기 위한 제 2 수단은 그것에 질의하는 것이다.

질의의 사용은, 그렇지 않다면, 엔-메스(en-mass)가 공급된다면, 요청자에 오버로드(overload)를 줄 수 있는 디바이스에 관한 정보를 드립-피딩(drip-feeding)하는 강력한 메카니즘이다.

제어 타입의 각각의 디바이스는 다른 디바이스들이 특정 디바이스 타입(도 5)을 제어 가능한지 여부를 질의(120)하도록 하는 수단을 제공한다. 질의에서 전달된 디바이스 타입은 간단한 디바이스 기술에서 사용된, 즉 디바이스 타입 계층에서 규정된 것과 동일한 필드다. 제어기는 질의에서 전달된 디바이스 타입 또는 그 디바이스 타입이 종속되는 곳인, 제어기 메모리(14)에 저장된 목록내의 최하위 디바이스 타입을 리턴함으로써, 제어기가 디바이스를 제어할 수 있는 레벨을 리턴한다(122). 예컨대, 간단한 스위치에 그것이 향상된 TV 디바이스를 제어할 수 있는 지 여부가 질의된다. 도 4에 도시된 상기 계층구조에 기초하여, 응답은 스위치가 향상된 TV 디바이스를 기본 디바이스 레벨까지 제어할 수 있다는 것이다. 스위치는 향상된 TV 디바이스의 디바이스 타입에 관해 자체에서 아는 것이 없지만, 디바이스 타입이 승계된 디바이스들을 또한 포함하므로, 기본 디바이스를 식별가능할 수 있고 이것을 스위치가 제어가능한 최하위 계층적으로 상위 디바이스 타입(lowest hierarchically superior device type)으로서 리턴한다.

제어기는 스위치가 간단한 디바이스 기술(도 6)로 리턴된 디바이스 타입을 제어할 수 있는지 여부를 결정하기 위한 알고리즘을 또한 구현한다. 스위치가 디바이스의 어드레스를 디스커버할 때, 스위치는 디바이스에 그것의 간단한 디바이스 기술을 물어보고(124), 이 정보를 수신할 시(126), 스위치는 질의 프로세스의 결과로서 스위치가 응답할 필요가 있는 동일한 질문인, 스위치가 이 타입의 디바이스를 어떤 정도까지 제어할 수 있는지를 시험한다(128). 결과는 2개의 질의 프로세스들(120, 122, 124, 126, 및 128)이 간단한 스위치 디바이스의 복잡성을 크게 증가시키지 않는다는 것이다. 동일한 것이 다른 간단한 디바. 스들에 적용된다.

가장 간단한 디바이스들상에서 XML, 및 그것의 압축 및 압축해제의 사용은 약간 헤비웨이트(heavyweight)인 것처럼 여겨질 수 있다. 프로토콜을 기술하기 위한 XML의 사용은 추후 향상에 대해 쉽게 확장가능하고, 기술하고 이해하기에 비교적 간단하고, 구축된 정보를 쉽게 다룰 수 있으며, '인터넷 도메인'과 즉시 호환가능한 솔루션을 제공한다.

XML상에 태그된 압축 기법을(HUCL내에서 규정된) 사용하는 것은 컨텐트 구조 를 보유하기 위한 일부 부가 오버헤드를 갖는, 종래의 순수 2진-기반 프로토콜의 그것에 대한 비교적 장황한 프로토콜 백 다운(back down)을 취한다.

명령의 압축된 형태로 명령을 제공받아야 한다면, 명령 구조에 관한 정보 및 데이터 값들에 대한 정의 테이블을 사용하여 어떤 다른 2진 기반 프로토콜을 판독할 수 있는 유사한 수단으로 판독될 수 있다. 2진 데이터가 XML 기반 표기법으로부터 발원되었을 수 있는 단일 힌트는 구조를 나타내기 위한 데이터의 존재일 수 있다.

HUCL 명세는 메시지들이 그것의 압축된 형태로 전송 매체를 통해서 항상 전송된다고 규정한다. 그러나, 간단한 디바이스상에서, 애플리케이션은 압축된 메시지들상에서 직접적으로 작동할 수 있어서, HUCL 소프트웨어 스택내에서 압축/압축해제 소프트웨어의 존재에 대한 그 디바이스상의 필요성을 제거한다. 이 경우에서, 애플리케이션은 간단한 디바이스 기술을 그것의 미리압축된 형태로 저장할 수 있고, 어떤 다른 2진 프로토콜 파서(binary protocol parser)와 본질적으로 유사할 수 있는 애플리케이션이 수신한 압축된 프로토콜 메시지들에 대한 파서를 가질 수 있으며; 응답 메시지들은 그것들의 압축된 형태로 또한 저장될 필요가 있을 수 있다.

이 접근법을 사용하여 이 섹션 전체를 통해 사용된 라이트 스위치 및 라이트 설비 예와 같은 가장 간단한 디바이스들이 감소된 소프트웨어 스택으로 구현될 수 있고, 간단한 디바이스가 이해하고 전송해야할 필요가 있을 수 있는 명령들의 수가 비교적 적게 주어지고(턴 라이트 온, 턴 라이트 오프, 토글, 전류 상태 획득, 디바 이스 기술 획득, 등), 애플리케이션 소프트웨어상의 오버헤드는 최소가 된다.

압축된 데이터상에서 작동하도록 애플리케이션을 실제로 구현하는 디바이스들에 대한 스케일러블 솔루션이 제공되고 이루어질 수 있지만, 디바이스가 더 복잡해질 때, 스택내에 압축/압축해제 기능성을 포함하고, 애플리케이션이 전체 XML 표기법에서 프로토콜 메시지들을 사용하도록 하기에 더 용이하게 되는 포인트가 존재할 것이다. 이 컷 오프 포인트는 디바이스 설계자에 전체적으로 일임(down)되고 HUCL에 규정되거나 또는 전용이 되지 않는다.

도 7은 HUCL을 이루는 구성요소들이 어떻게 함께 맞춰지는지를 예시한다. 구성요소들은 메모리내에 레코딩된 소프트웨어 구성요소들인 것이 인지될 것이다.

뒤따르는 섹션들은 HUCL 소프트웨어 스택(32)을 형성하는 층들, 및 그것들이 제공하는 기능성을 더 상세히 설명한다.

이전에 언급된 바와 같이, HUCL은 특정 전송 프로토콜에 따르지 않지만(예컨대, PCT/IP와 다른), 대신, 전송 스택(34)의 상부에 직접적으로 자리잡고 있다. 상이한 전송 스택들(34)은 그것들의 본질에 의해 애플리케이션들에 상이한 API's를 통해 상이한 서비스들을 제공할 것이고; HUCL 전송 적응 층(Transport Adaption Layer)(180)은 특정 전송 층에 대해 버퍼로서 작용한다.

전송 적응 층(180)은 서비스들의 고정된 전송 독립 세트를 HUCL내의 상위층들에 제공한다. 이 층의 요구조건들은 프로토콜 명세에서 상세히 규정된다.

메시징층(messaging layer)(182)은 HUCL 소프트웨어 스택의 기능성의 벌크(bulk)를 제공한다. 애플리케이션들은 HUCL API를 통해 이 층과 통신하고, 그것은 필요할 때(예컨대, 데이터가 수신될 때), 애플리케이션에 호출을 되돌려 행할 것이다.

메시징층(182)은 또한 어떤 최초 오류 보고를 다루고, 필요하다면 긍정응답(acknowledgement)한다. 응답들에 대한 잃어버린 메시지들 및 결합한 메시지들을 검사하기 위해 사용된 메시지 ID's 및 트랜섹션(Transaction) ID's는 또한 이 층에 의해 전체적으로 다루어진다.

메시징층(182)은 메시지가 압축되거나 또는 압축해제될 필요가 있을 때 압축/압축해제 서비스들(184)을 또한 사용한다. 상기된 바와 같이, 애플리케이션은 메시지들의 압축된 형태에서 메시지들과 배타적으로 관계하고, 이들 서비스들에 어떠한 호출도 이루어지지 않고 그것들은 런타임 스택으로부터 제거될 수 있다.

압축 및 압축해제 서비스들은 HUCL 메시지들의 압축된 형태 및 압축해제된 형태간에서 HUCL을 변환하기 위한 수단을 간단하게 메시지층에 제공한다. 시스템의 이 구성요소가 애플리케이션과의 모든 데이터 교환들이 압축된 메시지들로 이루어지는 로우-엔드(low-end) 디바이스들에서 결여되는 것이 가능하다.

애플리케이션 프로그래밍 인터페이스 API(186)는 모든 애플리케이션들이 HUCL 소프트웨어 스택과 통신하기 위해 통하는 인터페이스다. 통신은 HUCL 스택이 예컨대 전송 스택을 통해 수신된 메시지와 같은 하위층들에서 발생하는 일정 이벤트들의 결과로서 애플리케이션에 비동기 호출을 되돌려 행할 것인 양방향성(bi-directional)이다.

HUCL은 독립적인 전송 스택(34)이고, 이는 HUCL 메시징 프로토콜이 유선 및 무선 둘 다의 다양한 전송 스택들의 상부에 구축될 수 있음을 의미한다.

HUCL이 라이트웨이트 프로토콜로서 설계되었으므로, 따라서, 신생 지그비(Zigbee)(802.15.4) 표준과 같은 라이트웨이트 전송 스택들에 가장 적절하지만, 유선(예컨대, 이더넷) 및 무선(예컨대, 802.11b) 둘 다의 다른 프로토콜들의 광범위를 개시하는 TCP & UDP/IP의 상부에 자리잡을 수 있다.

전송 스택(34)상에 구현될 HUCL에 대해, HUCL 스택의 메시징층에 다수의 서비스들을 제공 가능해야만 한다. 이는 이들 서비스들이 전송 스택 자체내에 존재하거나 또는 HUCL 스택의 전송 추상층(Transport Abstraction Layer)에서 어떤 잃어버린 서비스들을 구현 가능해야만 할 수 있다는 의미다. 이들 서비스들은 어드레싱, 메시지 전달 및 디바이스 디스커버리와 같은 양상들을 커버할 수 있다(예컨대, 네트워크상의 다른 디바이스들의 어드레스들을 디스커버링하는).

프로토콜은 그자체가 매체(214)에 레코딩된 문서이고, 도 8에 도시된 바와 같은 뒤따르는 정보를 포함한다:

모든 HUCL 메시지들이 순응하는 포맷을 규정하는 범용 HUCL 메시지 포맷(200);

제어 프로토콜을 형성하는 특정 메시지들을 규정하는 메시지 정의(202);

어떤 메시지들이 언제 전송되는지 및 메시지 수신 시 애플리케이션의 요구조건들을 규정하는 메시지 시퀀싱 요구조건들(204);

HUCL과 그것을 사용하는 애플리케이션간의 양방향 인터페이스를 규정하는 HUCL API 정의(206);

HUCL 소프트웨어 스택의 메시징 시스템 요구조건들 및 기능성(208);

HUCL 메시지들의 압축을 위한 메카니즘을 규정하는 압축 알고리즘(210); 및

HUCL 소프트웨어 스택이 전송 시스템과 어떻게 인터페이스되는지를 규정하는 전송 적응 층 정의(transport adaption layer definition)(212)(예컨대, RF 스택).

HUCL은 따라서 간단하게 메시지 포맷 정의가 아니고, 또한 메시지 교환 및 압축을 캡슐화한다(encapsulate). 상기 목록내의 나중의 4개 항목(item)들은 디바이스에 존재할 수 있는 HUCL 소프트웨어 스택을 형성하고, 처음 3개 항목들은 스택 및 애플리케이션이 순응해야만 하는 요구조건들을 규정한다.

본원을 읽으므로서, 다른 변화들 및 변경들이 당업자에 명백할 것이다. 이러한 변화들 및 변경들은 설계, 제조, 및 네트워크 사용에서 이미 알고 있고, 여기서 기술된 특징들 대신에 또는 부가하여 사용될 수 있는, 동등물 및 다른 특징들을 수반할 수 있다. 이 애플리케이션에서, 청구항들이 특징들의 특정 조합들에 명료하게 설명되었지만, 공개의 범위는 본 발명이 행하는 바와 같이 어떤 또는 모든 동일한 기술 문제점들을 덜어내거나 또는 그렇지 않는, 여기서 함축적이거나 또는 명백하게 공개된 어떤 참신한 특징 또는 특징들의 어떤 참신한 조합, 또는 그것들의 어떤 일반화를 또한 포함한다는 것이 이해되어야 한다. 여기서, 출원인은 본 발명 또는 그로부터 유도된 어떤 그이상의 애플리케이션들의 진행 동안, 새로운 청구항들이 어떤 그러한 특징들 및/또는 그러한 특징들의 조합들에 명료하게 설명될 수 있음을 주의시킨다.

특히, 예들에서 사용된 특정 서브루틴 이름들은 쉽게 변경될 수 있다. 디바 이스들을 제어하는 컴퓨터 프로그램이 메모리(14)에 레코딩된 것으로 도시되어 있지만, 당업자는 컴퓨터 프로그램이 CD, 플로피 디스크, 등과 같은 많은 다른 타입의 레코드 캐리어상에 레코딩될 수 있음을 깨달을 것이다.

더욱이, 라이트 설비 및 라이트 스위치의 간단한 예가 광범위하게 전술되었음을 주지해야할 것이다. 당업자는 더 많은 복잡한 제어 시나리오가 또한 가능함을 인지할 것이다.

(57) 청구의 범위

청구항 1.

네트워크된 디바이스를 작동하는 방법에 있어서,

토큰-압축된(token-compressed) 메시지들을 수신하는 단계;

상기 네트워크된 디바이스로부터 간단한 디바이스 기술 응답(simple device description response)을 요구하는 인입 간단한 디바이스 기술 질의 메시지들을, 상기 인입 메시지들을 압축해제(decompress)하지 않고, 상기 수신된 토큰-압축된 메시지들에서 인식하는 단계; 및

간단한 디바이스 기술 응답을 요구하는 인입 디바이스 질의 메시지에 대한 응답으로서 디바이스 타입을 포함하는 간단한 디바이스 기술 메시지(230)를 전송하는 단계를 포함하는 네트워크된 디바이스 작동 방법.

청구항 2.

제 1 항에 있어서,

간단한 디바이스 기술을 전송하는 상기 단계는 상기 네트워크된 디바이스내의 메모리(14)로부터 미리 결정된 간단한 디바이스 기술을 판독하는 단계 및 상기 미리 결정된 간단한 디바이스 기술을 전송하는 단계를 포함하는, 네트워크된 디바이스 작동 방법.

청구항 3.

제 1 항 또는 제 2 항에 있어서,

상기 네트워크된 디바이스는 무선 네트워크의 일부이고, 토큰 압축된 메시지 들을 수신하고 상기 간단한 디바이스 기술을 전송하는 상기 단계들은 무선 신호들을 사용하는, 네트워크된 디바이스 작동 방법.

청구항 4.

네트워크된 디바이스에 있어서,

토큰-압축된 인간 판독가능 메시지들을 전송 및 수신하는 트랜시버(8); 및

상기 네트워크된 디바이스로부터 간단한 디바이스 기술 응답을 요구하는 수신된 디바이스 질의 메시지들을, 상기 인입 메시지들을 압축해제하지 않고 인식하고; 간단한 디바이스 기술 응답을 요구하는 인입 디바이스 질의 메시지에 대한 응답으로서 디바이스 타입을 포함하는 간단한 디바이스 기술을 상기 트랜시버를 통해 전송하는 단계들을 인입 토큰-압축된 인간 판독가능한 메시지들에 대해 수행하도록 배열된 메시지 핸들러(message handler)(26, 182)를 포함하는, 네트워크된 디바이스.

청구항 5.

제 4 항에 있어서,

인간 판독가능한 포맷으로부터 미리 압축된 미리 결정된 간단한 디바이스 기술 메시지를 저장하는 메모리(14)를 더 포함하고, 상기 메시지 핸들러는 상기 메모리로부터 상기 미리 결정된 간단한 디바이스 기술 메시지를 판독하고, 인입 디바이스 질의 메시지에 대한 응답하여 그것을 상기 트랜시버를 통해 전송하도록 배열된, 네트워크된 디바이스.

청구항 6.

네트워크 메시지들을 전송 및 수신하는 트랜시버(8)를 각각 갖는 복수의 네트워크된 디바이스들(2, 4); 및

간단한 디바이스 질의 메시지를 다른 디바이스들에 전송하고, 상기 다른 디바이스들로부터 후속적으로 수신되는 간단한 디바이스 기술 메시지들을 수신 및 해석하도록 배열된 적어도 하나의 네트워크된 디바이스(2)를 포함하고,

상기 네트워크된 디바이스들(2 및 4)의 각각은 상기 디바이스의 타입을 나타내는 디바이스 타입 값을 포함하는 규정된 길이의 간단한 디바이스 기술 메시지(230)를 전송함으로써, 상기 디바이스들 중 다른 것들로부터의 인입 간단한 디바이스 질의 메시지에 응답하도록 배열되고,

상기 복수의 네트워크된 디바이스들은 메시지들을 압축해제하기 위한 상기 능력이 없고 압축된 메시지들을 직접적으로 해석하는 적어도 하나의 간단한 디바이스(4), 및 상기 메시지들을 압축해제하기 위한 메시지 압축해제 배열(184) 및 상기 압축해제된 메시지들을 해석하기 위한 메시지 해석기를 포함하는 적어도 하나의 복잡한 디바이스를 포함하는, 시스템.

청구항 7.

제 6 항에 있어서,

상기 또는 각각의 간단한 디바이스(4)는 인간 판독가능한 포맷으로부터 미리 압축된 미리 결정된 간단한 디바이스 기술 메시지를 저장하는 메모리를 더 포함하고,

상기 메시지 핸들러는 상기 메모리로부터 상기 미리 결정된 간단한 디바이스 기술 메시지를 판독하고 인입 디바이스 질의 메시지에 응답하여 상기 트랜시버를 통해 그것을 전송하는 상기 트랜시버를 통해 전송하도록 배열된, 시스템.

청구항 8.

제 6 항 또는 제 7 항에 있어서,

상기 네트워크된 디바이스들은 메시지들을 디코딩하고 상기 디코딩된 메시지들에 대해 작용하도록 배열된 메시지 압축해제 유닛을 포함하는 적어도 하나의 디바이스를 포함하는, 시스템.

청구항 9.

컴퓨터 프로그램 제품에 있어서,

토큰-압축된 메시지들을 수신하기 위한 코드;

상기 네트워크된 디바이스로부터 간단한 디바이스 기술 응답을 요구하는 인입 간단한 디바이스 기술 질의 메시지들을, 상기 인입 메시지들을 압축해제하지 않고 상기 수신된 토큰-압축된 메시지들에서 인식하기 위한 코드; 및

간단한 디바이스 기술 응답을 요구하는 인입 디바이스 질의 메시지에 대한 응답으로서 디바이스 타입을 포함하는 간단한 디바이스 기술(230)을 전송하기 위한 코드를 포함하는 컴퓨터 프로그램 제품.

요약

본 발명은 네트워크된 디바이스들간의 통신을 위한 프로토콜에 관한 것이다. 디바이스들은, 다른 디바이스 타입이 종속되지 않는 제어기 디바이스 타입(52) 및 다수의 다른 디바이스 타입들이 종속되는 기본 디바이스 타입(54)을 포함하는 디바이스 타입들의 계층구조로서 논리적으로 배열된다. 디바이스들은 디바이스 타입을 포함하는 고정된 길이 및 포맷의 간단한 디바이스 기술 메시지를 구현하고, 일부 디바이스들은 부가 정보를 포함하는 확장된 디바이스 기술 메시지를 더 구현한다.

대표도

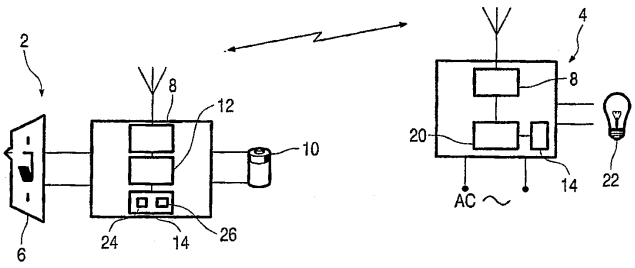
도 4

색인어

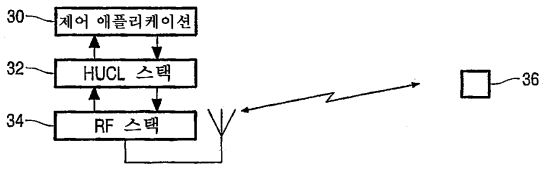
간단한 디바이스 기술 메시지, 디바이스 타입 계층구조, HUCL

도면

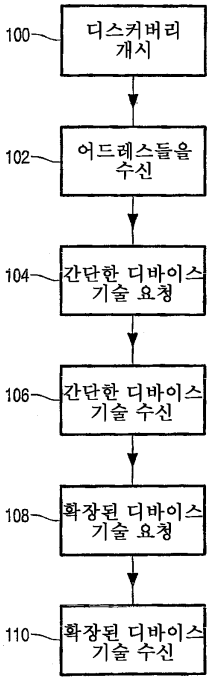
도면1



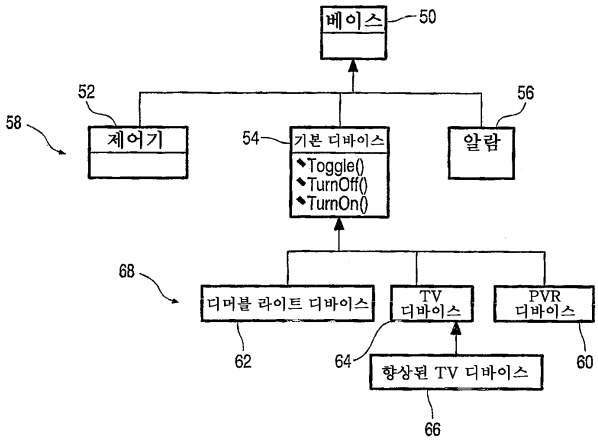
도면2



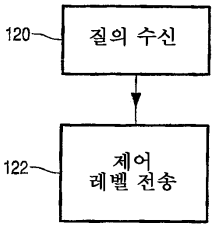
도면3


100
디스커버리 개시
102
어드레스들을 수신
104
간단한 디바이스 기술 요청
106
간단한 디바이스 기술 수신
108
확장된 디바이스 기술 요청
110
확장된 디바이스 기술 수신

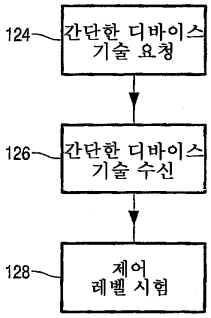
도면4


베이스
50
52
56
58
제어기
54
기본 디바이스
Toggle()
TurnOff()
TurnOn()
알람
68
디머블 라이트 디바이스
TV 디바이스
PVR 디바이스
62
64
60
향상된 TV 디바이스
66

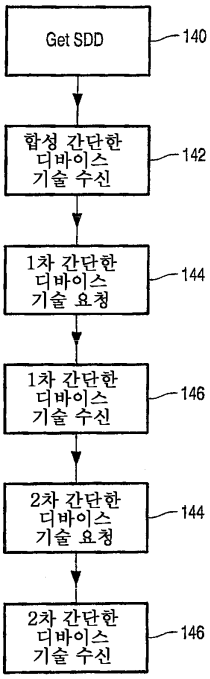
도면5


120
질의 수신
122
제어 레벨 전송

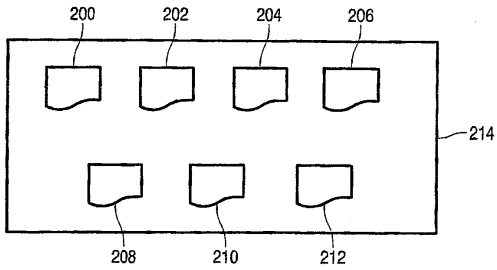
도면6



도면7



도면8



도면9

