



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2009년07월23일  
(11) 등록번호 10-0909137  
(24) 등록일자 2009년07월16일

(51) Int. Cl.  
G06K 17/00 (2006.01) G06F 12/00 (2006.01)  
G06F 12/16 (2006.01) B42D 15/10 (2006.01)  
(21) 출원번호 10-2003-7012433  
(22) 출원일자 2002년12월11일  
심사청구일자 2007년08월31일  
(85) 번역문제출일자 2003년09월24일  
(65) 공개번호 10-2004-0071585  
(43) 공개일자 2004년08월12일  
(86) 국제출원번호 PCT/JP2002/012953  
(87) 국제공개번호 WO 2003/065298  
국제공개일자 2003년08월07일  
(30) 우선권주장  
JP-P-2002-00017764 2002년01월25일 일본(JP)  
(56) 선행기술조사문헌  
US05590306 A1\*  
\*는 심사관에 의하여 인용된 문헌

(73) 특허권자  
소니 가부시끼 가이샤  
일본국 도쿄도 미나토쿠 코난 1-7-1  
(72) 발명자  
노다, 다꾸로  
일본 141-0001 도쿄도 시나가와쑤 기따시나가와  
6쑤메 7-35 소니 가부시끼 가이샤 내  
사토, 마꼬토  
일본 141-0001 도쿄도 시나가와쑤 기따시나가와  
6쑤메 7-35 소니 가부시끼 가이샤 내  
(74) 대리인  
구영창, 이중희, 장수길

전체 청구항 수 : 총 36 항

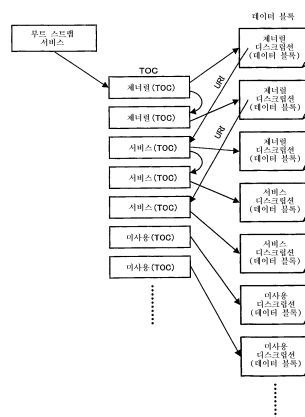
심사관 : 김창주

(54) 근접 통신 시스템 및 근접 통신 방법, 데이터 관리 장치 및 데이터 관리 방법, 기억 매체, 및 컴퓨터 판독 가능 매체

(57) 요약

비접촉 IC 카드에서는, 정보의 기입 및 판독의 단위로서 16 바이트를 1 블록으로 하고, 최대 동시 기입 사이즈인 8 블록까지의 정보의 기입은 보증되어 있다. 카드의 메모리 공간상에 TOC(Table Of Contents)의 구조를 구축한다. 최대 동시 기입 사이즈를 초과하는 정보를 기입할 때, 이것에 수반하는 TOC의 재기입을 1회의 트랜잭션으로 끝나도록 하여, 기입 도중에 카드가 기입 기계로부터 멀어지게 되더라도 메모리의 내용이 파괴되지 않도록 한다. 소정 이상의 크기의 데이터를 기입할 때에 임의의 타이밍에서 디바이스간 통신이 두절되어도, 데이터의 정합성을 적절히 보증할 수 있다.

대표도 - 도8



## 특허청구의 범위

### 청구항 1

근접하는 외부 장치로부터의 요구에 따라서 데이터의 재기입을 행하는 근접 통신 시스템으로서,

외부 장치로부터의 데이터 재기입 요구에 따른 데이터의 재기입을 행하기 위한 1 또는 그 이상의 데이터 블록과 각 데이터 블록마다 설치된 제어 정보로 구성되는 메모리 영역과,

외부 장치로부터의 데이터 재기입 요구에 따라서 해당하는 데이터 블록 및 그 제어 정보의 재기입을 제어하는 데이터 재기입 제어부를 구비하며,

상기 데이터 재기입 제어부는,

데이터 블록 재기입에 관련된 제어 정보의 합계가 허용가능한 재기입 데이터 사이즈 이하이면 그 재기입 동작을 1회의 트랜잭션으로 통합하여 행하며,

데이터 블록 재기입에 관련된 제어 정보의 합계가 허용가능한 재기입 데이터 사이즈를 초과할 때에는, 제어 정보를 링크된 순서에 따라서 각각이 1회의 트랜잭션으로 재기입 가능한 복수의 그룹으로 분할함과 아울러, 각 트랜잭션에서는 재기입 동작이 두절되어도 다음의 트랜잭션을 계속 가능하게 하기 위한 정보를 최후미의 제어 정보에 기입함으로써 재기입 동작을 1회의 트랜잭션으로 행하는

근접 통신 시스템.

### 청구항 2

제1항에 있어서,

상기 제어 정보는,

대응하는 데이터 블록에의 링크 정보 및 후속하는 데이터 블록에 대응하는 제어 정보에의 링크 정보를 가짐과 아울러,

대응하는 데이터 블록이 사용 상태, 미사용 상태, 기입 중, 소거가능 중에서 어느 상태인지를 나타내는 것을 특징으로 하는 근접 통신 시스템.

### 청구항 3

제2항에 있어서,

상기 메모리 영역은 선두의 제어 정보에 대한 링크 정보를 포함한 부트스트랩을 더 구비하는 것을 특징으로 하는 근접 통신 시스템.

### 청구항 4

제3항에 있어서,

상기 데이터 재기입 제어부는, 상기 부트스트랩으로부터 링크되어 있는 데이터 블록군에 대한 데이터 재기입 요구에 응답하여,

추가로 기입을 행할 수 있는 용량을 갖는 미사용 상태의 데이터 블록군을 탐색하고,

상기 탐색된 데이터 블록군의 선두에 해당하는 제어 정보를 미사용 상태에서 기입 중으로 변경하고, 그 외의 데이터 블록의 제어 정보를 사용 상태로 변경함과 아울러, 선두의 데이터 블록으로부터 순차적으로 데이터 블록을 찾아갈 수 있도록 각 제어 정보 사이에 링크를 형성하기 위해 상기 제어 정보의 데이터 재기입 동작을 실행하고,

재기입 요구되어 있는 데이터를 각 제어 정보 사이에 형성된 링크를 찾아가면서 상기 탐색된 데이터 블록군에 기입하고,

상기 탐색된 데이터 블록군의 선두의 제어 정보를 기입 중에서 사용 상태로 변경하고, 이전의 데이터 블록군의 선두의 제어 정보를 사용 상태에서 소거가능으로 변경함과 아울러, 상기 부트스트랩으로부터의 링크를 상기 탐

색된 데이터 블록군의 선두의 제어 정보를 가리키도록 변경함으로써, 일련의 제어 정보의 재기입을, 1회의 트랜잭션에 의해서 실행하고,

소거가능을 나타내는 제어 정보 및 거기에서부터 링크된 모든 제어 정보를 미사용 상태로 변경하기 위해 상기 제어 정보의 데이터 재기입 동작을 실행하는 것을 특징으로 하는 근접 통신 시스템.

#### 청구항 5

제4항에 있어서,

상기 데이터 재기입 제어부는, 상기 탐색된 데이터 블록군의 선두에 해당하는 제어 정보를 미사용 상태에서부터 기입 중으로 변경하고, 그 외의 데이터 블록의 제어 정보를 사용 상태로 변경함과 함께, 최선두의 데이터 블록으로부터 순차적으로 데이터 블록을 찾아갈 수 있도록 각 제어 정보 사이에 링크를 형성함으로써 일련의 제어 정보의 재기입을 행할 때,

제어 정보의 합계가 K(K는 1회의 트랜잭션으로 재기입할 수 있는 상기 허용가능한 재기입 데이터 사이즈에 해당) 이하일 때 그 재기입 동작을 1회의 트랜잭션으로 통합하여 행하며,

데이터 블록 재기입에 관련된 제어 정보의 합계가 K 를 초과할 때에는, 제어 정보를 링크된 순서에 따라서 각각 이 1회의 트랜잭션으로 재기입 가능한 복수의 그룹으로 분할하고, n회째의 트랜잭션에서는  $\{(K-1) \times n - (K-2)\}$  번째로부터  $(K-1) \times n$  번째까지의 제어 정보의 링크가 각각 다음의 제어 정보를 가리키도록 함과 함께  $\{(K-1)n+1\}$  번째의 제어 정보의 링크를 0(또는 널)으로 하는 재기입 동작을 반복하여 행하는 것을 특징으로 하는 근접 통신 시스템.

#### 청구항 6

제4항에 있어서,

상기 데이터 재기입 제어부는, 소거 가능을 나타내는 제어 정보 및 거기에서부터 링크된 모든 제어 정보를 미사용 상태로 변경함으로써 일련의 제어 정보의 재기입을 행할 때,

제어 정보의 합계가 K(K는 1회의 트랜잭션으로 재기입할 수 있는 상기 허용가능한 재기입 데이터 사이즈에 해당) 이하일 때 그 재기입 동작을 1회의 트랜잭션으로 통합하여 행하며,

데이터 블록 재기입에 관련된 제어 정보의 합계가 K 를 초과할 때에는, 제어 정보를 링크된 순서에 따라서 각각 이 1회의 트랜잭션으로 재기입 가능한 복수의 그룹으로 분할하고, n회째의 기입으로는  $(K-1) \times n$  번째까지의 제어 정보를 미사용 상태로 함과 함께  $\{(K-1) \times n + 1\}$  번째의 제어 정보를 소거가능으로 하는 재기입 동작을 반복하여 행하는 것을 특징으로 하는 근접 통신 시스템.

#### 청구항 7

제2항에 있어서,

상기 데이터 블록은 관련된 데이터 블록에 대한 제어 정보에의 링크 정보를 가질 수 있는 것을 특징으로 하는 근접 통신 시스템.

#### 청구항 8

제7항에 있어서,

상기 데이터 재기입 제어부는, 다른 데이터 블록으로부터 링크되어 있는 데이터 블록군에 대한 데이터 재기입 요구에 응답하여,

추가로 기입을 행할 수 있는 용량을 갖는 미사용 상태의 데이터 블록군을 탐색하고,

상기 탐색된 데이터 블록군의 선두에 해당하는 제어 정보를 미사용 상태에서부터 기입 중으로 변경하고, 그 외의 데이터 블록의 제어 정보를 사용 상태로 변경함과 함께, 최선두의 데이터 블록으로부터 순차적으로 데이터 블록을 찾아갈 수 있도록 각 제어 정보 사이에 링크를 형성하기 위해 상기 제어 정보의 데이터 재기입 동작을 실행하고,

재기입 요구되어 있는 데이터를 제어 정보 사이에 형성된 링크를 찾아가면서 상기 탐색된 데이터 블록군에 기입

하고,

상기 탐색된 데이터 블록군의 선두의 제어 정보를 이전의 데이터 블록군의 선두의 제어 정보로 교체함과 함께, 상기 탐색된 데이터 블록군의 선두의 제어 정보를 기입 중으로부터 사용 상태로 변경하고, 이전의 데이터 블록군의 선두의 제어 정보를 사용 상태에서 소거가능으로 변경함으로써, 일련의 제어 정보의 재기입을, 1회의 트랜잭션에 의해서 실행하고,

소거가능을 나타내는 제어 정보 및 거기에서부터 링크된 모든 제어 정보를 미사용 상태로 변경하기 위해 상기 제어 정보의 데이터 재기입 동작을 실행하는 것을 특징으로 하는 근접 통신 시스템.

#### 청구항 9

근접하는 외부 장치로부터의 요구에 따라서 데이터의 재기입을 행하는 근접 통신 방법으로서,

메모리 영역은, 외부 장치로부터의 요구에 따른 데이터의 재기입을 행하는 1 또는 그 이상의 데이터 블록과, 각 데이터 블록마다 설치된 제어 정보를 포함하고,

데이터 블록 재기입에 관련된 제어 정보의 합계가 허용가능한 재기입 데이터 사이즈 이하이면 그 재기입 동작을 1회의 트랜잭션으로 통합하여 행하는 단계와,

데이터 블록 재기입에 관련된 제어 정보의 합계가 허용가능한 재기입 데이터 사이즈를 초과할 때에는, 제어 정보를 링크된 순서에 따라서 각각이 1회의 트랜잭션으로 재기입 가능한 복수의 그룹으로 분할하는 단계와,

재기입 동작이 두절되어도 다음의 트랜잭션을 계속 가능하게 하기 위한 정보를 최후미의 제어 정보에 기입하는 단계

를 포함하는 근접 통신 방법.

#### 청구항 10

제9항에 있어서,

상기 제어 정보는,

대응하는 데이터 블록에의 링크 정보 및 후속하는 데이터 블록에 대응하는 제어 정보에의 링크 정보를 가짐과 함께,

대응하는 데이터 블록이 사용 상태, 미사용 상태, 기입 중, 소거가능 중에서 어느 상태인지를 나타내는 것을 특징으로 하는 근접 통신 방법.

#### 청구항 11

제10항에 있어서,

상기 메모리 영역은 선두의 제어 정보에 대한 링크 정보를 포함한 부트스트랩을 더 구비하는 것을 특징으로 하는 근접 통신 방법.

#### 청구항 12

제11항에 있어서,

상기 메모리 영역의 데이터 재기입 동작을 행하는 단계는,

상기 부트스트랩으로부터 링크되어 있는 데이터 블록군에 대한 데이터 재기입 요구에 응답하여, 추가로 기입을 행할 수 있는 용량을 갖는 미사용 상태의 데이터 블록군을 탐색하는 서브 단계와,

상기 탐색된 데이터 블록군의 선두에 상당하는 제어 정보를 미사용 상태에서 기입 중으로 변경하고, 그 외의 데이터 블록의 제어 정보를 사용 상태로 변경함과 함께, 선두의 데이터 블록으로부터 순차적으로 데이터 블록을 찾아갈 수 있도록 각 제어 정보 사이에 링크를 형성하기 위해 상기 제어 정보의 데이터 재기입 동작을 실행하는 서브 단계와,

재기입 요구되어 있는 데이터를 제어 정보 사이에 형성된 링크를 찾아가면서 상기 탐색된 데이터 블록군에 기입하는 서브 단계와,

상기 탐색된 데이터 블록군의 선두의 제어 정보를 기입 중으로부터 사용 상태로 변경하고, 이전의 데이터 블록군의 선두의 제어 정보를 사용 상태에서 소거가능으로 변경함과 함께, 상기 부트스트랩으로부터의 링크를 상기 탐색된 데이터 블록군의 선두의 제어 정보를 가리키도록 변경함으로써, 일련의 제어 정보의 재기입을, 1회의 트랜잭션에 의해서 실행하는 서브 단계와,

소거 가능을 나타내는 제어 정보 및 거기에서 링크된 모든 제어 정보를 미사용 상태로 변경하기 위해 상기 제어 정보의 데이터 재기입 동작을 실행하는 서브 단계를 포함하는 것을 특징으로 하는 근접 통신 방법.

### 청구항 13

제12항에 있어서,

상기 메모리 영역의 데이터 재기입 동작을 행하는 단계는, 상기 탐색된 데이터 블록군의 선두에 해당하는 제어 정보를 미사용 상태에서 기입 중으로 변경하고, 그 외의 데이터 블록의 제어 정보를 사용 상태로 변경함과 함께, 최선두의 데이터 블록으로부터 순차 데이터 블록을 찾아갈 수 있도록 각 제어 정보 사이에 링크를 형성함으로써, 일련의 제어 정보의 재기입을 행할 때,

재기입될 제어 정보의 합계가  $K$  ( $K$ 는 1회의 트랜잭션으로 재기입할 수 있는 상기 허용가능한 재기입 데이터 사이즈에 해당) 이하일 때 그 재기입 동작을 1회의 트랜잭션으로 통합하여 행하는 서브 단계와,

데이터 블록 재기입에 관련된 제어 정보의 합계가  $K$  를 초과할 때에는, 제어 정보를 링크된 순서에 따라서 각각이 1회의 트랜잭션으로 재기입 가능한 복수의 그룹으로 분할하고,  $n$ 회째의 트랜잭션에서는  $\{(K-1) \times n - (K-2)\}$  번째로부터  $(K-1) \times n$  번째까지의 제어 정보의 링크가 각각 다음의 제어 정보를 가리키도록 함과 함께  $\{(K-1)n+1\}$  번째의 제어 정보의 링크를 0(또는 널)으로 하는 재기입 동작을 반복하여 행하는 서브 단계를 포함하는 것을 특징으로 하는 근접 통신 방법.

### 청구항 14

제12항에 있어서,

상기 메모리 영역의 데이터 재기입 동작을 행하는 단계는, 소거가능을 나타내는 제어 정보 및 거기에서부터 링크된 모든 제어 정보를 미사용 상태로 변경함으로써 일련의 제어 정보의 재기입을 행할 때,

재기입될 제어 정보의 합계가  $K$  ( $K$ 는 1회의 트랜잭션으로 재기입할 수 있는 상기 허용가능한 재기입 데이터 사이즈에 해당) 이하일 때 그 재기입 동작을 1회의 트랜잭션으로 통합하여 행하는 서브 단계와,

데이터 블록 재기입에 관련된 제어 정보의 합계가  $K$  를 초과할 때에는, 제어 정보를 링크된 순서에 따라서 각각이 1회의 트랜잭션으로 재기입 가능한 복수의 그룹으로 분할하고,  $n$ 회째의 기입으로는  $(K-1) \times n$  번째까지의 제어 정보를 미사용 상태로 함과 함께  $\{(K-1) \times n + 1\}$  번째의 제어 정보를 소거가능으로 하는 재기입 동작을 반복하여 행하는 서브 단계를 포함하는 것을 특징으로 하는 근접 통신 방법.

### 청구항 15

제10항에 있어서,

데이터 블록은 관련된 데이터 블록에 대한 제어 정보에의 링크 정보를 가질 수 있는 것을 특징으로 하는 근접 통신 방법.

### 청구항 16

제15항에 있어서,

상기 메모리 영역의 데이터 재기입 동작을 행하는 단계는,

다른 데이터 블록으로부터 링크되어 있는 데이터 블록군에 대한 데이터 재기입 요구에 응답하여, 추가로 기입을 행할 수 있는 용량을 갖는 미사용 상태의 데이터 블록군을 탐색하는 서브 단계와,

상기 탐색된 데이터 블록군의 선두에 해당하는 제어 정보를 미사용 상태에서 기입 중으로 변경하고, 그 외의 데이터 블록의 제어 정보를 사용 상태로 변경함과 함께, 선두의 데이터 블록으로부터 순차적으로 데이터 블록을 찾아갈 수 있도록 각 제어 정보 사이에 링크를 형성하기 위해 상기 제어 정보의 데이터 재기입 동작을 실행하는

서브 단계와,

재기입 요구되어 있는 데이터를 제어 정보 사이에 형성된 링크를 찾아가면서 상기 탐색된 데이터 블록군에 기입하는 서브 단계와,

상기 탐색된 데이터 블록군의 선두의 제어 정보를 이전의 데이터 블록군의 선두의 제어 정보로 교체함과 아울러, 상기 탐색된 데이터 블록군의 선두의 제어 정보를 기입 중으로부터 사용 상태로 변경하고, 이전의 데이터 블록군의 선두의 제어 정보를 사용 상태로부터 소거 가능으로 변경함으로써, 일련의 제어 정보의 재기입을, 1회의 트랜잭션에 의해서 실행하는 서브 단계와,

소거 가능을 나타내는 제어 정보 및 거기에서 링크된 모든 제어 정보를 미사용 상태로 변경하기 위해 상기 제어 정보의 데이터 재기입 동작을 실행하는 서브 단계를 포함하는 것을 특징으로 하는 근접 통신 방법.

#### 청구항 17

1회의 트랜잭션으로 데이터가 재기입되는 시스템에 있어서 데이터를 관리하기 위한 데이터 관리 장치로서,

사용자 데이터를 재기입하기 위한 1 또는 그 이상의 데이터 블록과, 각각의 데이터 블록을 관리하기 위한 제어 정보를 구비한 시스템의 메모리 영역과,

해당하는 데이터 블록 및 그 제어 정보의 재기입을 제어하는 데이터 재기입 제어부 - 상기 데이터 재기입 제어부는, 데이터 블록 재기입에 관련된 제어 정보의 합계가 허용가능한 재기입 데이터 사이즈 이하이면 그 재기입 동작을 1회의 트랜잭션으로 통합하여 행하며, 데이터 블록 재기입에 관련된 제어 정보의 합계가 허용가능한 재기입 데이터 사이즈를 초과할 때에는 제어 정보를 링크된 순서에 따라서 각각이 1회의 트랜잭션으로 재기입 가능한 복수의 그룹으로 분할함과 아울러 재기입 동작이 두절되어도 다음의 트랜잭션을 계속 가능하게 하기 위한 정보를 최후미의 제어 정보에 기입함으로써 재기입 동작을 1회의 트랜잭션으로 행함 -

를 포함하는 데이터 관리 장치.

#### 청구항 18

제17항에 있어서,

상기 제어 정보는,

대응하는 데이터 블록에의 링크 정보 및 후속하는 데이터 블록에 대응하는 제어 정보에의 링크 정보를 가짐과 함께,

대응하는 데이터 블록이 사용 상태, 미사용 상태, 기입 중, 소거가능 중에서 어느 상태인지를 나타내는 것을 특징으로 하는 데이터 관리 장치.

#### 청구항 19

제17항에 있어서,

상기 메모리 영역은 최선두의 제어 정보에 대한 링크 정보를 포함한 부트스트랩을 더 구비하는 것을 특징으로 하는 데이터 관리 장치.

#### 청구항 20

제19항에 있어서,

상기 부트스트랩으로부터 링크되어 있는 데이터 블록군에 대한 데이터 재기입 요구에 응답하여,

추가로 기입을 행할 수 있는 용량을 갖는 미사용 상태의 데이터 블록군을 탐색하고,

상기 탐색된 데이터 블록군의 선두에 상당하는 제어 정보를 미사용 상태로부터 기입 중으로 변경하고, 그 외의 데이터 블록의 제어 정보를 사용 상태로 변경함과 함께, 최선두의 데이터 블록으로부터 순차적으로 데이터 블록을 찾아가 수 있도록 각 제어 정보 사이에 링크를 형성하기 위해 상기 제어 정보의 데이터 재기입 동작을 실행하고,

재기입 요구되어 있는 데이터를 제어 정보 사이에 형성된 링크를 찾아가면서 상기 탐색된 데이터 블록군에 기입

하고,

상기 탐색된 데이터 블록군의 선두의 제어 정보를 기입 중으로부터 사용 상태로 변경하고, 이전의 데이터 블록군의 선두의 제어 정보를 사용 상태로부터 소거가능으로 변경함으로써, 일련의 제어 정보의 재기입을, 1회의 트랜잭션에 의해서 실행하고,

소거가능을 나타내는 제어 정보 및 거기에서부터 링크된 모든 제어 정보를 미사용 상태로 변경하기 위해 상기 제어 정보의 데이터 재기입 동작을 실행하는 것을 특징으로 하는 데이터 관리 장치.

## 청구항 21

제20항에 있어서,

상기 탐색된 데이터 블록군의 선두에 해당하는 제어 정보를 미사용 상태로부터 기입 중으로 변경하고, 그 외의 데이터 블록의 제어 정보를 사용 상태로 변경함과 함께, 선두의 데이터 블록으로부터 순차적으로 데이터 블록을 찾아갈 수 있도록 각 제어 정보 사이에 링크를 형성함으로써 일련의 제어 정보의 재기입을 행할 때,

재기입될 제어 정보의 합계가  $K$  ( $K$ 는 1회의 트랜잭션으로 재기입할 수 있는 상기 허용가능한 재기입 데이터 사이즈에 해당) 이하일 때 그 재기입 동작을 1회의 트랜잭션으로 통합하여 행하며,

데이터 블록 재기입에 관련된 제어 정보의 합계가  $K$  를 초과할 때에는, 제어 정보를 링크된 순서에 따라서 각각이 1회의 트랜잭션으로 재기입 가능한 복수의 그룹으로 분할하고,  $n$ 회째의 트랜잭션에서는  $\{(K-1) \times n - (K-2)\}$  번째로부터  $(K-1) \times n$  번째까지의 제어 정보의 링크가 각각 다음의 제어 정보를 가리키도록 함과 함께  $\{(K-1)n+1\}$  번째의 제어 정보의 링크를 0(또는 널)으로 하는 재기입 동작을 반복하여 행하는 것을 특징으로 하는 데이터 관리 장치.

## 청구항 22

제20항에 있어서,

소거가능을 나타내는 제어 정보 및 거기에서부터 링크된 모든 제어 정보를 미사용 상태로 변경함으로써 일련의 제어 정보의 재기입을 행할 때,

재기입하는 제어 정보의 합계가 1회의 트랜잭션으로 재기입 가능한 데이터 사이즈 이하이면 그 재기입 동작을 1회의 트랜잭션으로 통합하여 행하며,

재기입하는 제어 정보의 합계가 1회의 트랜잭션으로 재기입 가능한 데이터 사이즈를 초과할 때에는, 제어 정보를 링크된 순서에 따라서 1회의 트랜잭션으로 재기입 가능한 개수의 그룹으로 분할하고,  $n$ 회째의 기입으로는  $(K-1) \times n$  번째까지의 제어 정보를 미사용 상태로 함과 아울러  $\{(K-1) \times n + 1\}$  번째의 제어 정보를 소거가능으로 하는 재기입 동작을 반복하여 행하는 것을 특징으로 하는 데이터 관리 장치.

## 청구항 23

제17항에 있어서,

데이터 블록은 관련된 데이터 블록에 대한 제어 정보에 대한 링크 정보 데이터를 가질 수 있는 것을 특징으로 하는 데이터 관리 장치.

## 청구항 24

제23항에 있어서,

다른 데이터 블록으로부터 링크되어 있는 데이터 블록군에 대한 데이터 재기입 요구에 응답하여,

추가로 기입을 행할 수 있는 용량을 갖는 미사용 상태의 데이터 블록군을 탐색하고,

상기 탐색된 데이터 블록군의 선두에 해당하는 제어 정보를 미사용 상태로부터 기입 중으로 변경하고, 그 외의 데이터 블록의 제어 정보를 사용 상태로 변경함과 함께, 선두의 데이터 블록으로부터 순차 데이터 블록을 찾아갈 수 있도록 각 제어 정보 사이에 링크를 형성하기 위해 상기 제어 정보의 데이터 재기입 동작을 실행하고,

재기입 요구되어 있는 데이터를 각 제어 정보 사이에 형성된 링크를 찾아가면서 상기 탐색된 데이터 블록군에 기입하고,

상기 탐색된 데이터 블록군의 선두의 제어 정보와 이전의 데이터 블록군의 선두의 제어 정보를 교체함과 아울러, 상기 탐색된 데이터 블록군의 선두의 제어 정보를 기입 중으로부터 사용 상태로 변경하고, 이전의 데이터 블록군의 선두의 제어 정보를 사용 상태에서 소거가능으로 변경함으로써 일련의 제어 정보의 재기입을, 1회의 트랜잭션에 의해서 실행하고,

소거가능을 나타내는 제어 정보 및 거기에서부터 링크된 모든 제어 정보를 미사용 상태로 변경하기 위해 상기 제어 정보의 데이터 재기입 동작을 실행하는 것을 특징으로 하는 데이터 관리 장치.

#### 청구항 25

1회의 트랜잭션으로 데이터가 재기입되는 시스템에 있어서 데이터를 관리하기 위한 데이터 관리 방법으로서,

사용자 데이터를 재기입하기 위한 1 또는 그 이상의 데이터 블록과, 각각의 데이터 블록을 관리하기 위한 제어 정보를 구비한 메모리 영역을 제공하는 단계와,

데이터 블록 재기입에 관련된 제어 정보의 합계가 허용가능한 재기입 데이터 사이즈 이하이면 그 재기입 동작을 통합하여 행하며, 데이터 블록 재기입에 관련된 제어 정보의 합계가 허용가능한 재기입 데이터 사이즈를 초과할 때에는 제어 정보를 링크된 순서에 따라서 각각이 1회의 트랜잭션으로 재기입 가능한 복수의 그룹으로 분할함과 아울러 재기입 동작이 두절되어도 다음의 트랜잭션을 계속 가능하게 하기 위한 정보를 최후미의 제어 정보에 기입함으로써 재기입 동작을 1회의 트랜잭션으로 행함으로써, 재기입 동작을 1회의 트랜잭션으로 행하는 단계를 포함하는 데이터 관리 방법.

#### 청구항 26

제25항에 있어서,

상기 제어 정보는,

대응하는 데이터 블록에 대한 링크 정보 및 후속하는 데이터 블록에 대응하는 제어 정보에 대한 링크 정보를 가짐과 함께,

대응하는 데이터 블록이 사용 상태, 미사용 상태, 기입 중, 소거가능 중에서 어느 상태인지를 나타내는 것을 특징으로 하는

데이터 관리 방법.

#### 청구항 27

제25항에 있어서,

상기 메모리 영역은 선두의 제어 정보에 대한 링크 정보를 포함한 부트스트랩을 더 구비하는 것을 특징으로 하는 데이터 관리 방법.

#### 청구항 28

제27항에 있어서,

상기 부트스트랩으로부터 링크되어 있는 데이터 블록군에 대한 데이터 재기입 요구에 응답하여, 추가로 기입을 행할 수 있는 용량을 갖는 미사용 상태의 데이터 블록군을 탐색하는 단계와,

상기 탐색된 데이터 블록군의 선두에 해당하는 제어 정보를 미사용 상태에서 기입 중으로 변경하고, 그 외의 데이터 블록의 제어 정보를 사용 상태로 변경함과 함께, 최선두의 데이터 블록으로부터 순차적으로 데이터 블록을 찾아갈 수 있도록 각 제어 정보 사이에 링크를 형성하기 위해 상기 제어 정보의 데이터 재기입 동작을 실행하는 단계와,

재기입 요구되어 있는 데이터를 제어 정보 사이에 형성된 링크를 찾아가면서 상기 탐색된 데이터 블록군에 기입하는 단계와,

상기 탐색된 데이터 블록군의 선두의 제어 정보를 기입 중으로부터 사용 상태로 변경하고, 이전의 데이터 블록군의 선두의 제어 정보를 사용 상태에서 소거가능으로 변경하고, 상기 부트스트랩으로부터의 링크를 상기 탐색된 데이터 블록군의 선두의 제어 정보를 가리키도록 변경함으로써, 일련의 제어 정보의 재기입을, 1회의 트랜



잭션에 의해서 실행하는 단계와,

소거 가능을 나타내는 제어 정보 및 거기에서부터 링크된 모든 제어 정보를 미사용 상태로 변경하기 위해 상기 제어 정보의 데이터 재기입 동작을 실행하는 단계

를 포함하는 것을 특징으로 하는 데이터 관리 방법.

#### 청구항 29

제28항에 있어서,

상기 탐색된 데이터 블록군의 선두에 상당하는 제어 정보를 미사용 상태에서부터 기입 중으로 변경하고, 그 외의 데이터 블록의 제어 정보를 사용 상태로 변경함과 함께, 최선두의 데이터 블록으로부터 순차적으로 데이터 블록을 찾아갈 수 있도록 각 제어 정보 사이에 링크를 형성함으로써, 일련의 제어 정보의 재기입을 실행하는 상기 단계는,

재기입될 제어 정보의 합계가  $K$  ( $K$ 는 1회의 트랜잭션으로 재기입할 수 있는 상기 허용가능한 재기입 데이터 사이즈에 해당) 이하일 때 그 재기입 동작을 1회의 트랜잭션으로 통합하여 행하는 서브 단계와,

데이터 블록 재기입에 관련된 제어 정보의 합계가  $K$  를 초과할 때에는, 제어 정보를 링크된 순서에 따라서 각각이 1회의 트랜잭션으로 재기입 가능한 복수의 그룹으로 분할하고,  $n$ 회째의 트랜잭션에서는  $\{(K-1) \times n - (K-2)\}$  번째로부터  $(K-1) \times n$  번째까지의 제어 정보의 링크가 각각 다음의 제어 정보를 가리키도록 함과 함께  $\{(K-1)n+1\}$  번째의 제어 정보의 링크를 0(또는 널)으로 하는 재기입 동작을 반복하여 행하는 서브 단계를 포함하는 것을 특징으로 하는 데이터 관리 방법.

#### 청구항 30

제28항에 있어서,

상기 메모리 영역의 데이터 재기입 동작을 행하는 단계에 있어서, 소거가능을 나타내는 제어 정보 및 거기에서부터 링크된 모든 제어 정보를 미사용 상태로 변경함으로써 일련의 제어 정보의 재기입을 행하는 단계는,

재기입될 제어 정보의 합계가  $K$  ( $K$ 는 1회의 트랜잭션으로 재기입할 수 있는 상기 허용가능한 재기입 데이터 사이즈에 해당) 이하일 때 그 재기입 동작을 1회의 트랜잭션으로 통합하여 행하는 서브 단계와,

재기입하는 제어 정보의 합계가 1회의 트랜잭션으로 재기입 가능한 데이터 사이즈를 초과할 때에는, 제어 정보를 링크된 순서에 따라서 1회의 트랜잭션으로 재기입 가능한 개수의 그룹으로 분할하고,  $n$ 회째의 기입으로는  $(K-1) \times n$  번째까지의 제어 정보를 미사용 상태로 함과 아울러  $\{(K-1) \times n + 1\}$  번째의 제어 정보를 소거가능으로 하는 재기입 동작을 반복하여 행하는 서브 단계

를 더 포함하는 것을 특징으로 하는 데이터 관리 방법.

#### 청구항 31

제25항에 있어서,

데이터 블록은 관련된 데이터 블록에 대한 제어 정보에의 링크 정보를 가질 수 있는 것을 특징으로 하는 데이터 관리 방법.

#### 청구항 32

제31항에 있어서,

다른 데이터 블록으로부터 링크되어 있는 데이터 블록군에 대한 데이터 재기입 요구에 응답하여,

추가로 기입을 행할 수 있는 용량을 갖는 미사용 상태의 데이터 블록군을 탐색하는 단계와,

상기 탐색된 데이터 블록군의 선두에 상당하는 제어 정보를 미사용 상태에서부터 기입 중으로 변경하고, 그 외의 데이터 블록의 제어 정보를 사용 상태로 변경함과 함께, 최선두의 데이터 블록으로부터 순차적으로 데이터 블록을 찾아갈 수 있도록 각 제어 정보 사이에 링크를 형성하기 위해 상기 제어 정보의 데이터 재기입 동작을 실행하는 단계와,

재기입 요구되어 있는 데이터를 각 제어 정보 사이에 형성된 링크를 찾아가면서 상기 탐색된 데이터 블록군에 기입하고,

상기 탐색된 데이터 블록군의 선두의 제어 정보를 이전의 데이터 블록군의 선두의 제어 정보로 교체함과 아울러, 상기 탐색된 데이터 블록군의 선두의 제어 정보를 기입 중으로부터 사용 상태로 변경하고, 이전의 데이터 블록군의 선두의 제어 정보를 사용 상태에서부터 소거가능으로 변경함으로써 일련의 제어 정보의 재기입을, 1회의 트랜잭션에 의해서 실행하는 단계와,

소거가능을 나타내는 제어 정보 및 거기에서부터 링크된 모든 제어 정보를 미사용 상태로 변경하기 위해 상기 제어 정보의 데이터 재기입 동작을 실행하는 단계

를 포함하는 것을 특징으로 하는 데이터 관리 방법.

### 청구항 33

근접하는 외부 장치로부터의 요구에 따라서 데이터의 재기입을 행하기 위한 처리를 컴퓨터 시스템상에서 실행하도록 기술된 컴퓨터 소프트웨어를 컴퓨터 판독 가능 형식으로 저장한 기억 매체로서,

메모리 영역은, 외부 장치로부터의 요구에 따른 데이터의 재기입을 행하기 위한 1 또는 그 이상의 데이터 블록과 각 데이터 블록마다 설치된 제어 정보를 포함하고,

데이터 블록 재기입에 관련된 제어 정보의 합계가 허용가능한 재기입 데이터 사이즈 이하이면 그 재기입 동작을 통합하여 행하며, 데이터 블록 재기입에 관련된 제어 정보의 합계가 허용가능한 재기입 데이터 사이즈를 초과할 때에는 제어 정보를 링크된 순서에 따라서 각각이 1회의 트랜잭션으로 재기입 가능한 복수의 그룹으로 분할함과 아울러, 재기입 동작이 두절되어도 다음의 트랜잭션을 계속 가능하게 하기 위한 정보를 최후미의 제어 정보에 기입함으로써, 재기입 동작을 1회의 트랜잭션으로 행하는 것을 특징으로 하는 기억 매체.

### 청구항 34

1회의 트랜잭션이 보증되는 시스템상에서 데이터 관리 방법을 컴퓨터 시스템 상에서 실행하도록 기술된 컴퓨터 소프트웨어를 컴퓨터 판독 가능 형식으로 저장한 기억 매체로서, 상기 데이터 관리 방법은,

사용자 데이터를 재기입하기 위한 1 또는 그 이상의 데이터 블록과, 각각의 데이터 블록을 관리하기 위한 제어 정보를 구비한 메모리 영역을 제공하는 단계와,

데이터 블록 재기입에 관련된 제어 정보의 합계가 허용가능한 재기입 데이터 사이즈 이하이면 그 재기입 동작을 통합하여 행하며, 데이터 블록 재기입에 관련된 제어 정보의 합계가 허용가능한 재기입 데이터 사이즈를 초과할 때에는 제어 정보를 링크된 순서에 따라서 각각이 1회의 트랜잭션으로 재기입 가능한 복수의 그룹으로 분할함과 아울러 재기입 동작이 두절되어도 다음의 트랜잭션을 계속 가능하게 하기 위한 정보를 최후미의 제어 정보에 기입함으로써 재기입 동작을 1회의 트랜잭션으로 행함으로써, 재기입 동작을 1회의 트랜잭션으로 행하는 단계를 포함하는 것을 특징으로 하는 기억 매체.

### 청구항 35

근접하는 외부 장치로부터의 요구에 따라서 메모리 영역에 데이터의 재기입을 행하기 위한 처리를 컴퓨터 시스템상에서 실행하기 위한 컴퓨터 프로그램을 갖는 컴퓨터 판독 가능 매체로서,

메모리 영역은, 외부 장치로부터의 요구에 따른 데이터의 재기입을 행하기 위한 1 또는 그 이상의 데이터 블록과 각 데이터 블록마다 설치된 제어 정보를 포함하고,

상기 처리는,

데이터 블록 재기입에 관련된 제어 정보의 합계가 허용가능한 재기입 데이터 사이즈 이하이면 그 재기입 동작을 통합하여 행하는 단계와,

데이터 블록 재기입에 관련된 제어 정보의 합계가 허용가능한 재기입 데이터 사이즈를 초과할 때에는 제어 정보를 링크된 순서에 따라서 각각이 1회의 트랜잭션으로 재기입 가능한 복수의 그룹으로 분할함과 아울러 재기입 동작이 두절되어도 다음의 트랜잭션을 계속 가능하게 하기 위한 정보를 최후미의 제어 정보에 기입함으로써 재기입 동작을 1회의 트랜잭션으로 행하는 단계를 포함하는 것을 특징으로 하는 컴퓨터 판독 가능 매체.

### 청구항 36

메모리 영역에 1회의 트랜잭션으로 데이터를 재기입하는 것이 보증되는 시스템에서 데이터 관리 처리를 컴퓨터 시스템상에서 실행하기 위한 컴퓨터 프로그램을 갖는 컴퓨터 판독 가능 매체로서,

데이터 블록 재기입에 관련된 제어 정보의 합계가 허용가능한 재기입 데이터 사이즈 이하이면 그 재기입 동작을 1회의 트랜잭션으로 통합하여 행하는 단계와,

데이터 블록 재기입에 관련된 제어 정보의 합계가 허용가능한 재기입 데이터 사이즈를 초과할 때에는 제어 정보를 링크된 순서에 따라서 각각이 1회의 트랜잭션으로 재기입 가능한 복수의 그룹으로 분할함과 아울러 재기입 동작이 두절되어도 다음의 트랜잭션을 계속 가능하게 하기 위한 정보를 최후미의 제어 정보에 기입하고, 사용자 데이터의 재기입과 관련된 재기입 동작을 보증하는 트랜잭션 수단에 의해 데이터 블록의 사용자 정보를 재기입하고 제어 정보를 재기입함으로써, 재기입 동작을 1회의 트랜잭션으로 행하는 단계를 포함하는 컴퓨터 판독 가능 매체.

### 청구항 37

삭제

### 청구항 38

삭제

### 청구항 39

삭제

### 청구항 40

삭제

## 명세서

### 기술분야

- <1> 본 발명은, 국소적으로 적용되는 비접촉·근접 통신 시스템에 관한 것으로, 특히, 비접촉 IC 카드 기술로 대표되는, 사용자가 물리적으로 근접 통신을 행하는 디바이스끼리를 가까이 함으로써 적어도 한쪽의 기기가 상대의 디바이스를 발견하여 자동적으로 동작하는 근접 통신 시스템에 관한 것이다.
- <2> 더 자세하게는, 본 발명은, 사용자가 디바이스끼리를 물리적으로 가까이 함으로써 자동으로 동작함과 함께, 사용자가 디바이스끼리를 멀리하여 임의의 타이밍에서 디바이스 사이의 통신이 두절되어도 동작을 보증하는 근접 통신 시스템 및 근접 통신 방법, 데이터 관리 장치 및 데이터 관리 방법, 기억 매체 및 컴퓨터 프로그램에 관한 것으로, 특히, 소정 크기 이상의 데이터를 기입할 때에 임의의 타이밍에서 디바이스간 통신이 두절되어도, 데이터의 정합성을 보증하는 근접 통신 시스템 및 근접 통신 방법, 데이터 관리 장치 및 데이터 관리 방법, 기억 매체 및 컴퓨터 프로그램에 관한 것이다.

### 배경기술

- <3> 국소적으로만 적용 가능한 무선 통신 수단의 일례로서 비접촉 IC 카드를 예를 들 수 있다.
- <4> 이러한 종류의 무선 통신은, 예를 들면 전자 유도 원리에 기초하여 실현된다. 즉, 무선 통신 시스템은, 메모리 기능을 갖는 IC 카드와, IC 카드의 메모리에 대하여 기입 및 판독 액세스를 행하는 카드 리더/라이터로 구성되어, 카드 리더/라이터측의 안테나와 IC 카드측의 루프 코일이 시스템적으로 1개의 트랜스포머를 형성하고 있다.
- <5> 카드 리더/라이터측에서는, 안테나에 흘리는 전류를 변조함으로써, IC 카드 상의 루프 코일의 유기 전압이 변조되는 작용에 의해, 카드 리더/라이터로부터 IC 카드로 데이터 송신을 행할 수 있다. 또한, IC 카드는, 루프 코일의 단자 사이의 부하 변동에 의해, IC 카드 리더/라이터측의 안테나 단자 사이의 임피던스가 변화하여 안테나의 통과 전류나 전압이 변동하는 작용에 의해, 카드 리더/라이터로 회신을 행한다.

- <6> IC 카드로 대표되는 비접촉·근접 통신 시스템은, 조작 상의 손쉬움으로, 광범위하게 보급되어 있다. 예를 들면, 비밀 코드나 그 밖의 개인 인증 정보, 전자 티켓 등의 가치 정보 등을 IC 카드에 저장해 둬으로써, 현금 인출기나 콘서트 회장의 출입구, 역의 개찰구 등에 설치된 카드 리더/라이터는, 이용자가 갖다대는 IC 카드에 비접촉으로 액세스하여 인증 처리를 행할 수 있다.
- <7> 최근에는, 미세화 기술의 향상과 더불어 비교적 대용량의 메모리 공간을 갖는 IC 카드가 출현하고 있다. 대용량 메모리를 갖는 IC 카드에 의해, 복수의 어플리케이션(또는 서비스)을 동시에 저장해 둘 수 있기 때문에, 1장의 IC 카드를 복수의 용도에 이용할 수 있다. 예를 들면, 1장의 IC 카드상에, 전자 결제를 행하기 위한 전자화폐나, 특정한 콘서트 회장에 입장하기 위한 전자 티켓 등, 다수의 서비스를 저장해 둬으로써, 1장의 IC 카드를 다양한 용도에 적용시킬 수 있다.
- <8> 또한, 각 디바이스에 IC 카드 및 카드 리더/라이터 쌍방의 기능을 장착함으로써, IC 카드 기술을 범용성이 있는 쌍방향의 근접 통신 인터페이스로서 이용할 수 있다.
- <9> 예를 들면, 컴퓨터나 정보 가전 기기와 같은 기기끼리 근접 통신 시스템이 구성되는 경우에는, 통신은 일대일로 행해진다. 또한, 어떤 기기가 비접촉 IC 카드와 같은 기기 이외의 상대 디바이스(카드라고 부름)와 통신하는 것도 가능하며, 이 경우에는, 1개의 기기와 복수의 카드에서 일대다의 통신을 행하는 어플리케이션도 생각할 수 있다.
- <10> 이러한 근접 통신 시스템에서는, 사용자가 물리적으로 근접 통신을 행하는 디바이스끼리를 가까이 함으로써, 적어도 한쪽의 기기는 상대의 디바이스를 발견하여 자동으로 동작을 개시할 수 있다. 그러나, 반대로 디바이스 사이의 거리가 멀어져 있는 경우, 임의의 시점에서 통신이 두절된다. 이 때문에, 임의의 타이밍에서 사용자가 디바이스끼리를 멀리하는 상황에서도 시스템이 혼란되지 않도록 동작 보증 기구가 필요하게 된다.
- <11> 기존의 근접 통신 시스템에 있어서는, IC 카드 자체의 기능에 의해, 일정한 크기(예를 들면 128 바이트)의 데이터 기입에 관해서는 동작을 보증하는 경우도 있다. 그러나, 이 경우에도, 그 이상의 크기의 정보를 기입할 때는 타이밍에 따라 데이터의 부정합성이 발생한다.
- <12> 특히, 통신 범위가 국소로 제한되는 IC 카드와 같은 근접 통신에 있어서는, 사용자 조작에 의해 디바이스 사이가 멀어지거나 장애물에 간섭받는 등, 통신이 도중에서 끊길 가능성이 매우 높다. IC 카드의 대 메모리화, 다용도·다서비스화가 진행함에 따라, 비교적 큰 데이터를 IC 카드에 전송할 기회가 점점 더 증가하기 때문에, 갑자기 통신이 두절되었을 때의 데이터 보증의 문제는 중대하다.
- <13> <발명의 개시>
- <14> 본 발명의 목적은, 사용자가 물리적으로 근접 통신을 행하는 디바이스끼리를 가까이 함으로써 적어도 한쪽의 기기가 상대의 디바이스를 발견하여 자동으로 동작할 수 있는, 우수한 근접 통신 시스템을 제공하는 것에 있다.
- <15> 본 발명의 다른 목적은, 사용자가 디바이스끼리를 물리적으로 가까이 함으로써 자동으로 동작함과 함께, 사용자가 디바이스끼리를 멀리하여 임의의 타이밍에서 디바이스 사이의 통신이 두절되어도 동작을 보증할 수 있는, 우수한 근접 통신 시스템 및 근접 통신 방법, 데이터 관리 장치 및 데이터 관리 방법, 기억 매체 및 컴퓨터 프로그램을 제공하는 것에 있다.
- <16> 본 발명의 또 다른 목적은, 소정 이상의 크기의 데이터를 기입할 때에 임의의 타이밍에서 디바이스 사이의 통신이 두절되어도 데이터의 정합성을 적합하게 보증할 수 있는, 우수한 근접 통신 시스템 및 근접 통신 방법, 데이터 관리 장치 및 데이터 관리 방법, 기억 매체 및 컴퓨터 프로그램을 제공하는 것에 있다.
- <17> 본 발명은, 상기 과제를 참작하여 이루어진 것으로, 그 제1 양태는, 근접하는 외부 장치로부터의 요구에 따라서 데이터의 재기입을 행하는 근접 통신 시스템으로서,
- <18> 외부 장치로부터의 요구에 따른 데이터의 재기입을 행하는 1 이상의 데이터 블록과, 각 데이터 블록마다 설치된 제어 정보로 구성되는 메모리 영역과,
- <19> 외부 장치로부터의 데이터 재기입 요구에 따라서 해당하는 데이터 블록 및 그 제어 정보의 재기입을 제어하는 데이터 재기입 제어부를 구비하며,
- <20> 1회의 트랜잭션으로 재기입 가능한 데이터 사이즈 이하에서의 상기 메모리 영역에 대한 데이터 재기입 동작이 보증되고, 또한 제어 정보 K개분이 1회의 트랜잭션에 상당하고,

- <21> 상기 데이터 재기입 제어부는, 외부 기기와의 근접 통신이 두절되는 상황에서도, 데이터 블록의 재기입에 관련된 제어 정보의 재기입 동작을 보증하는 것을 특징으로 하는 근접 통신 시스템이다.
- <22> 단, 여기서 말하는 「시스템」이란, 복수의 장치(또는 특정한 기능을 실현하는 기능 모듈)가 논리적으로 집합한 것을 말하며, 각 장치나 기능 모듈이 단일의 케이스 내에 있는지의 여부는 특별히 상관하지 않는다.
- <23> 또한, 본 발명의 제2 양태는, 근접하는 외부 장치로부터의 요구에 따라서 데이터의 재기입을 행하는 근접 통신 방법으로서,
- <24> 메모리 영역은, 외부 장치로부터의 요구에 따른 데이터의 재기입을 행하는 1 이상의 데이터 블록과, 각 데이터 블록마다 설치된 제어 정보로 구성되며,
- <25> 1회의 트랜잭션으로 재기입 가능한 데이터 사이즈 이하에서의 상기 메모리 영역에 대한 데이터 재기입 동작은 시스템으로 보증되며, 또한 제어 정보 K개분이 1회의 트랜잭션에 상당하고,
- <26> 외부 기기와의 근접 통신이 두절되는 상황에서도 데이터 블록의 재기입에 관련된 제어 정보의 재기입 동작을 보증하는 단계를 포함하는 것을 특징으로 하는 근접 통신 방법이다.
- <27> 여기서, 제어 정보는, 대응하는 데이터 블록에의 링크 정보 및 후속하는 데이터 블록에 대응하는 제어 정보에의 링크 정보를 가지며, 대응하는 데이터 블록이 사용 상태, 미사용 상태, 기입 중, 소거가능 중 어느 상태인지를 나타내도록 구성되어 있다.
- <28> 또한, 메모리 영역은, 선두의 제어 정보에 대한 링크 정보를 포함한 부트스트랩을 더 구비하고 있다.
- <29> 또한, 데이터 블록은, 관련된 데이터 블록에 대한 제어 정보에의 링크 정보를 가질 수 있다.
- <30> 근접 통신 시스템의 트랜스미션 인터페이스로서 비접촉 IC 카드 기술을 적용하는 경우에는, 사용자가 물리적으로 근접 통신을 행하는 디바이스끼리를 가까이 함으로써, 적어도 한쪽의 기기는 상대의 디바이스를 발견하여 자동으로 동작할 수가 있어, 조작성이 우수하다. 그 반면, 디바이스 사이의 거리가 떨어진 경우, 임의의 시점에서 통신이 두절되어, 데이터의 일관성을 잃어버리게 된다. 이 때문에, 데이터 기입 중의 임의의 타이밍에서 사용자가 디바이스끼리를 멀리하여도 메모리의 내용이 파괴되지 않는 동작 보증 기구가 필요하게 된다.
- <31> 본 발명의 제1 양태에 따른 근접 통신 시스템, 또는 본 발명의 제2 양태에 따른 근접 통신 방법에 따르면, 데이터 블록의 재기입에 관련된 제어 정보의 합계가 1회의 트랜잭션으로 재기입 가능한 데이터 사이즈 이하인 경우에는, 그 재기입 동작을 1회의 트랜잭션으로 통합하여 행함으로써, 제어 정보의 정합성을 보증한다. 이 결과, 통신의 두절 등에 의해 만일 데이터 블록의 재기입 동작에 실패한 경우에도, 제어 정보를 기초로 데이터를 회복할 수가 있어, 시스템의 파괴를 방지할 수 있다.
- <32> 또한, 본 발명의 제1 양태에 따른 근접 통신 시스템, 또는 본 발명의 제2 양태에 따른 근접 통신 방법에 따르면, 데이터 블록의 재기입에 관련된 제어정보의 합계가 1회의 트랜잭션으로 재기입 가능한 데이터 사이즈(즉 K개)를 초과할 때에는, 제어 정보를 링크된 순서에 따라서 1회의 트랜잭션으로 재기입 가능한 개수의 그룹으로 분할함과 아울러, 각 트랜잭션에서는 재기입 동작이 두절되어도 다음의 트랜잭션을 계속 가능하게 하기 위한 정보를 최후미의 제어 정보에 기입하도록 한다.
- <33> 이 결과, 통신의 두절 등에 의해 일련의 제어 정보의 재기입이 중단된다고 해도, 중단된 장소에서 재기입 동작을 재개할 수 있으므로, 제어 정보의 일관성이 보증된다. 또한, 통신의 두절 등에 의해 만일 데이터 블록의 재기입 동작에 실패하는 경우에도, 제어 정보를 기초로 데이터를 회복할 수가 있어, 시스템의 파괴를 방지할 수 있다.
- <34> 부트스트랩으로부터 링크되어 있는 데이터 블록군에 대하여 재기입을 행하는 경우에는, 우선, 새롭게 기입을 행할 수 있는 용량을 갖는 미사용 상태의 데이터 블록군을 탐색한다. 그리고, 해당 탐색된 데이터 블록군의 선두에 상당하는 제어 정보를 미사용 상태에서 기입 중으로 변경하고, 그밖의 데이터 블록의 제어 정보를 사용 상태로 변경함과 함께, 선두의 데이터 블록으로부터 순차 데이터 블록을 찾아갈 수 있도록 각 제어 정보 사이에 링크를 형성한다. 여기서의 일련의 제어 정보의 재기입을, 동작 보증한 트랜잭션에 의해서 실행함으로써, 제어 정보의 정합성을 보증한다.
- <35> 계속해서, 재기입 요구되는 데이터를 각 제어 정보 사이에 형성된 링크를 찾아가면서 해당 탐색된 데이터 블록군에 기입한다. 그리고, 그 탐색된 데이터 블록군의 선두의 제어 정보를 기입 중에서 사용 상태로 변경하고, 이전의 데이터 블록군의 선두의 제어 정보를 사용 상태에서 소거가능으로 변경함과 함께, 부트스트랩으로부터의

링크를 유지하기 위해서 상기 부트스트랩으로부터의 링크를 그 탐색된 데이터 블록군의 선두의 제어 정보를 가리키도록 변경한다. 여기서의 일련의 제어 정보의 재기입을 1회의 트랜잭션에 의해서 실행함으로써, 제어 정보의 정합성을 보증한다.

- <36> 마지막으로 소거가능을 나타내는 제어 정보 및 거기에서부터 링크된 모든 제어 정보를 미사용 상태로 변경하여, 이전의 데이터 블록을 재이용 가능하게 한다. 또한, 여기서의 일련의 제어 정보의 재기입을, 동작 보증한 트랜잭션에 의해서 실행함으로써, 제어 정보의 정합성을 보증한다.
- <37> 또한, 다른 데이터 블록으로부터 링크되어 있는 데이터 블록군에 대하여 재기입을 행하는 경우에는, 우선, 새롭게 기입을 행할 수 있는 용량을 갖는 미사용 상태의 데이터 블록군을 탐색한다. 그리고, 해당 탐색된 데이터 블록군의 선두에 상당하는 제어 정보를 미사용 상태에서 기입 중으로 변경하고, 그 이외의 데이터 블록의 제어 정보를 사용 상태로 변경함과 함께, 선두의 데이터 블록으로부터 순차 데이터 블록을 찾아갈 수 있도록 각 제어 정보 사이에 링크를 형성한다. 여기서의 일련의 제어 정보의 재기입을, 동작 보증한 트랜잭션에 의해서 실행함으로써, 제어 정보의 정합성을 보증한다.
- <38> 계속해서, 재기입 요구되어 있는 데이터를 각 제어 정보 사이에 형성된 링크를 찾아가면서 해당 탐색된 데이터 블록군에 기입한다. 그리고, 다른 데이터 블록으로부터 링크를 유지하기 위해서 그 탐색된 데이터 블록군의 선두의 제어 정보와 이전의 데이터 블록군의 선두의 제어 정보를 교체함과 함께, 그 탐색된 데이터 블록군의 선두의 제어 정보를 기입 중에서 사용 상태로 변경하고, 이전의 데이터 블록군의 선두의 제어 정보를 사용 상태에서 소거가능으로 변경한다. 여기서의 일련의 제어 정보의 재기입을, 동작 보증한 트랜잭션에 의해서 실행함으로써, 제어 정보의 정합성을 보증한다.
- <39> 마지막으로 소거가능을 나타내는 제어 정보 및 거기에서부터 링크된 모든 제어 정보를 미사용 상태로 변경하여, 이전의 데이터 블록을 재이용 가능하게 한다. 또한, 여기서의 일련의 제어 정보의 재기입을, 동작 보증한 트랜잭션에 의해서 실행함으로써, 제어 정보의 정합성을 보증한다.
- <40> 여기서, 그 탐색된 데이터 블록군의 선두에 상당하는 제어 정보를 미사용 상태에서 기입 중으로 변경하고, 그 이외의 데이터 블록의 제어 정보를 사용 상태로 변경함과 함께, 선두의 데이터 블록으로부터 순차 데이터 블록을 찾아갈 수 있도록 각 제어 정보 사이에 링크를 형성하는 일련의 제어 정보의 재기입을 행할 때, 재기입하는 제어 정보의 합계가 1회의 트랜잭션으로 재기입 가능한 데이터 사이즈(즉 K개)를 초과할 때에는, 제어 정보를 링크된 순서에 따라서 1회의 트랜잭션으로 재기입 가능한 개수의 그룹으로 분할하고, n회째의 트랜잭션으로는,  $\{(K-1) \times n - (K-2)\}$ 번째부터  $(K-1) \times n$ 번째까지의 제어 정보의 링크가 각각 다음의 제어 정보를 가리키도록 함과 함께  $\{(K-1)n+1\}$ 번째의 제어 정보의 링크를 0(또는 널)으로 하는 재기입 동작을 행하도록 하여도 된다.
- <41> 이 결과, 통신의 두절 등에 의해 일련의 제어 정보의 재기입이 중단된다고 해도, 여분의 링크가 형성되지 않아도 되기 때문에, 제어 정보의 일관성이 보증된다. 또한, 통신의 두절 등에 의해 만일 데이터 블록의 재기입 동작에 실패한 경우에도 제어 정보를 기초로 데이터를 회복할 수가 있어, 시스템의 파괴를 방지할 수 있다.
- <42> 또한, 소거가능을 나타내는 제어 정보 및 거기에서부터 링크된 모든 제어 정보를 미사용 상태로 변경하는 일련의 제어 정보의 재기입을 행할 때, 재기입하는 제어 정보의 합계가 1회의 트랜잭션으로 재기입 가능한 데이터 사이즈(즉 K개)를 초과할 때에는, 제어 정보를 링크된 순서에 따라서 1회의 트랜잭션으로 재기입 가능한 개수의 그룹으로 분할하고, n회째의 기입에서는  $(K-1) \times n$ 번째까지의 제어 정보를 미사용 상태로 함과 함께  $\{(K-1) \times n+1\}$ 번째의 제어 정보를 소거가능으로 하는 재기입 동작을 행하여도 된다.
- <43> 이 결과, 통신의 두절 등에 의해 일련의 제어 정보의 재기입이 중단된다고 해도, 소거가능으로 되어 있는 제어 정보를 찾아 내어, 중단한 재기입 동작을 재개할 수 있으므로 제어 정보의 일관성이 보증된다. 또한, 통신의 두절이 생기더라도, 이미 사용되지 않게 된 데이터 블록의 제어 정보를 확실하게 찾아내어 미사용 상태로 복귀시킬 수 있기 때문에, 메모리 영역을 낭비 없이 효율적으로 사용할 수 있다.
- <44> 또한, 본 발명의 제3 양태는, 1회의 트랜잭션의 동작이 보증되어 있는 시스템에서 데이터를 관리하기 위한 데이터 관리 장치로서,
- <45> 시스템의 메모리 영역에, 사용자 데이터를 기입하기 위한 1 이상의 데이터 블록과, 각 데이터 블록을 관리하기 위한 제어 정보를 설치하고,
- <46> 각 제어 정보는 1회의 트랜잭션으로 K개분을 재기입 가능한 고정 데이터 사이즈이며,
- <47> 데이터 블록의 사용자 데이터의 재기입에 따른, 재기입 동작을 보증한 트랜잭션에 의해 제어 정보의 재기입을



행하는 것을 특징으로 하는 데이터 관리 장치이다.

- <48> 또한, 본 발명의 제4 양태는, 1회의 트랜잭션의 동작이 보증되어 있는 시스템에서 데이터를 관리하기 위한 데이터 관리 방법으로서,
- <49> 시스템의 메모리 영역에, 사용자 데이터를 기입하기 위한 1 이상의 데이터 블록과, 각 데이터 블록을 관리하기 위한 제어 정보를 설치하고,
- <50> 각 제어 정보는 1회의 트랜잭션으로 K개분을 재기입 가능한 고정 데이터 사이즈이며,
- <51> 데이터 블록의 사용자 데이터를 재기입할 수 있음과 함께, 이것에 수반하는 재기입 동작을 보증한 트랜잭션에 의해 제어 정보의 재기입을 행하는 것을 특징으로 하는 데이터 관리 방법이다.
- <52> 여기서, 제어 정보는, 대응하는 데이터 블록에의 링크 정보 및 후속하는 데이터 블록에 대응하는 제어 정보에의 링크 정보를 가짐과 아울러, 대응하는 데이터 블록이 사용 상태, 미사용 상태, 기입 중, 소거가능 중 어느 상태 인지를 나타내도록 구성되어 있다. 또한, 메모리 영역은, 선두의 제어 정보에 대한 링크 정보를 포함한 부트스트랩을 더 구비하고 있다. 또한, 데이터 블록은, 관련된 데이터 블록에 대한 제어 정보에의 링크 정보를 가질 수 있다.
- <53> 메모리 영역에 대한 사용자 데이터의 재기입을 행하기 위해서, 재기입 데이터의 트랜스미션 인터페이스로서 비접촉 IC 카드 기술을 적용하는 경우에는, 디바이스 사이의 거리가 멀어지면, 임의의 시점에서 통신이 두절되어 데이터의 일관성을 잃어버린다. 이 때문에, 데이터 기입 중의 임의의 타이밍에서 사용자가 디바이스끼리를 멀리하여도 메모리의 내용이 파괴되지 않는 동작 보증 기구가 필요하게 된다.
- <54> 본 발명의 제3 양태에 따른 데이터 관리 장치, 또는 본 발명의 제4 양태에 따른 데이터 관리 방법에 따르면, 데이터 블록의 재기입에 관련된 제어 정보의 합계가 1회의 트랜잭션으로 재기입 가능한 데이터 사이즈 이하인 경우에는, 그 재기입 동작을 1회의 트랜잭션으로 통합하여 행함으로써, 제어 정보의 정합성을 보증한다. 따라서, 통신의 두절 등에 의해 만일 데이터 블록의 재기입 동작에 실패해도, 제어 정보를 기초로 데이터를 회복할 수 있다.
- <55> 또한, 본 발명의 제3 양태에 따른 데이터 관리 장치, 또는 본 발명의 제4 양태에 따른 데이터 관리 방법에 따르면, 데이터 블록의 재기입에 관련된 제어 정보의 합계가 1회의 트랜잭션으로 재기입 가능한 데이터 사이즈(즉 K개)를 초과할 때에는, 제어 정보를 링크된 순서에 따라서 1회의 트랜잭션으로 재기입 가능한 개수의 그룹으로 분할함과 함께, 각 트랜잭션에서는 재기입 동작이 두절되어도 다음의 트랜잭션을 계속 가능하게 하기 위한 정보를 최후미의 제어 정보에 기입하도록 한다. 따라서, 통신의 두절 등에 의해 일련의 제어 정보의 재기입이 중단된다고 해도, 중단된 장소에서 재기입 동작을 재개할 수 있으므로, 제어 정보의 일관성이 보증된다.
- <56> 또한, 본 발명의 제5 양태는, 근접하는 외부 장치로부터의 요구에 따라서 데이터의 재기입을 행하기 위한 처리를 컴퓨터 시스템상에서 실행하도록 기술된 컴퓨터 소프트웨어를 컴퓨터 판독 가능 형식으로 물리적으로 저장한 기억 매체로서,
- <57> 메모리 영역은, 외부 장치로부터의 요구에 따른 데이터의 재기입을 행하는 1 이상의 데이터 블록과, 각 데이터 블록마다 설치된 제어 정보로 구성되며,
- <58> 1회의 트랜잭션으로 재기입 가능한 데이터 사이즈 이하에서의 상기 메모리 영역에 대한 데이터 재기입 동작은 시스템으로 보증되며, 또한 제어 정보 K개분이 1회의 트랜잭션에 상당하며,
- <59> 상기 컴퓨터 소프트웨어는, 제어 정보에서의 데이터의 정합성을 유지하면서 상기 메모리 영역의 데이터 재기입 동작을 행하는 단계를 포함하는 것을 특징으로 하는 기억 매체이다.
- <60> 또한, 본 발명의 제6 양태는, 1회의 트랜잭션의 동작이 보증되어 있는 시스템에서의 데이터 관리를 컴퓨터 시스템상에서 실행하도록 기술된 컴퓨터 소프트웨어를 컴퓨터 판독 가능 형식으로 물리적으로 저장한 기억 매체로서, 상기 컴퓨터 소프트웨어는,
- <61> 시스템의 메모리 영역에, 사용자 데이터를 기입하기 위한 1 이상의 데이터 블록과, 각 데이터 블록을 관리하기 위한 제어 정보를 설치하는 단계와,
- <62> 데이터 블록의 사용자 데이터의 재기입과 함께, 이것에 수반하는 재기입 동작을 보증한 트랜잭션에 의해 제어 정보의 재기입을 행하는 단계를 포함하는 것을 특징으로 하는 기억 매체이다.

- <63> 본 발명의 제5 및 제6 양태에 따른 기억 매체는, 예를 들면, 여러가지의 프로그램 코드를 실행 가능한 범용 컴퓨터 시스템에 대하여, 컴퓨터 소프트웨어를 컴퓨터 판독 가능한 형식으로 제공하는 매체이다. 이러한 매체는, 예를 들면, DVD(Digital Versatile Disc)나 CD(Compact Disc), FD(Flexible Disk), MO(Magneto-Optical disc) 등의 착탈 가능한 가반성의 기억 매체이다. 또는, 네트워크(네트워크는 무선, 유선을 구별하지 않음) 등의 전송 매체 등을 경유하여 컴퓨터 소프트웨어를 특정한 컴퓨터 시스템에 제공하는 것도 기술적으로 가능하다.
- <64> 또한, 본 발명의 제5 및 제6 양태에 따른 기억 매체는, 컴퓨터 시스템상에서 소정의 컴퓨터 소프트웨어의 기능을 실현하기 위한, 컴퓨터 소프트웨어와 기억 매체와의 구조상 또는 기능상의 협동적 관계를 정의한 것이다. 바꾸어 말하면, 본 발명의 제5 및 제6 양태에 따른 기억 매체를 통하여 소정의 컴퓨터 소프트웨어를 컴퓨터 시스템에 인스톨함으로써, 컴퓨터 시스템상에서는 협동적 작용이 발휘되어, 본 발명의 제2 양태에 따른 근접 통신 방법, 또는 본 발명의 제4 양태에 따른 데이터 관리 방법 방법과 마찬가지로의 작용 효과를 얻을 수 있다.
- <65> 또한, 본 발명의 제7 양태는, 근접하는 외부 장치로부터의 요구에 따라서 데이터의 재기입을 행하기 위한 처리를 컴퓨터 시스템상에서 실행하도록 컴퓨터 판독 가능 형식으로 기술된 컴퓨터 프로그램으로서,
- <66> 메모리 영역은, 외부 장치로부터의 요구에 따른 데이터의 재기입을 행하는 1 이상의 데이터 블록과, 각 데이터 블록마다 설치된 제어 정보로 구성되며,
- <67> 1회의 트랜잭션으로 재기입 가능한 데이터 사이즈 이하에서의 상기 메모리 영역에 대한 데이터 재기입 동작은 시스템으로 보증되고, 또한 제어 정보 K개분이 1회의 트랜잭션에 상당하고,
- <68> 제어 정보에서의 데이터의 정합성을 유지하면서 상기 메모리 영역의 데이터 재기입 동작을 행하는 단계를 포함하는 것을 특징으로 하는 컴퓨터 프로그램이다.
- <69> 또한, 본 발명의 제8 양태는, 1회의 트랜잭션의 동작이 보증되어 있는 시스템에서의 데이터 관리를 컴퓨터 시스템상에서 실행하도록 컴퓨터 판독 가능 형식으로 기술된 컴퓨터 프로그램으로서,
- <70> 시스템의 메모리 영역에, 사용자 데이터를 기입하기 위한 1 이상의 데이터 블록과, 각 데이터 블록을 관리하기 위한 제어 정보를 설치하는 단계와,
- <71> 데이터 블록의 사용자 데이터의 재기입과 아울러, 이것에 수반하는 재기입 동작을 보증한 트랜잭션에 의해 제어 정보의 재기입을 행하는 단계를 포함하는 것을 특징으로 하는 컴퓨터 프로그램이다.
- <72> 본 발명의 제7 및 제8 양태에 따른 컴퓨터 프로그램은, 컴퓨터 시스템상에서 소정의 처리를 실행하도록 컴퓨터 판독 가능 형식으로 기술된 컴퓨터 프로그램을 정의한 것이다. 바꾸어 말하면, 본 발명의 제7 및 제8 양태에 따른 컴퓨터 프로그램을 컴퓨터 시스템에 인스톨함으로써, 컴퓨터 시스템상에서는 협동적 작용이 발휘되어, 본 발명의 제2 양태에 따른 근접 통신 방법, 또는 본 발명의 제4 양태에 따른 데이터 관리 방법과 마찬가지로의 작용 효과를 얻을 수 있다.
- <73> 본 발명의 또 다른 목적, 특징이나 이점은, 후술하는 본 발명의 실시 형태나 첨부하는 도면에 기초하여 보다 상세한 설명에 의해서 분명히 될 것이다.

### 산업상 이용 가능성

- <287> 본 발명에 따르면, 사용자가 디바이스끼리를 물리적으로 가까이 함으로써 자동으로 동작함과 함께, 사용자가 디바이스끼리를 멀리하여 임의의 타이밍에서 디바이스 사이의 통신이 두절되어도 동작을 보증할 수 있는, 우수한 근접 통신 시스템 및 근접 통신 방법, 데이터 관리 장치 및 데이터 관리 방법, 기억 매체 및 컴퓨터 프로그램을 제공할 수 있다.
- <288> 또한, 본 발명에 따르면, 소정 이상의 크기의 데이터를 기입할 때에 임의의 타이밍에서 디바이스 사이의 통신이 두절되어도, 데이터의 정합성을 적절히 보증할 수 있는, 우수한 근접 통신 시스템 및 근접 통신 방법, 데이터 관리 장치 및 데이터 관리 방법, 기억 매체 및 컴퓨터 프로그램을 제공할 수 있다.

### 도면의 간단한 설명

- <74> 도 1은 본 발명의 일 실시 형태에 따른 근접 통신 시스템에서의 인터랙션 모델을 모식적으로 도시하는 도면.
- <75> 도 2는 본 발명의 일 실시 형태에 따른 근접 통신 시스템에서의 프로토콜 스택 구조를 도시하는 도면.



- <76> 도 3은 디스크립션(Description)의 구조를 모식적으로 도시하는 도면.
- <77> 도 4는 이니시에이터를 결정하기 위한 액세스 수순을 도시하는 시퀀스 도면.
- <78> 도 5는 이니시에이터가 컨트롤러의 이니시에이션을 행하기 위한 액세스 수순을 도시하는 시퀀스 도면.
- <79> 도 6은 컨트롤러의 타깃에 대한 액세스 수순을 도시하는 시퀀스 도면.
- <80> 도 7은 본 발명의 일 실시 형태에 따른 근접 통신 시스템에서 동작하는 디바이스의 상태 천이도를 도시하는 도면.
- <81> 도 8은 본 발명의 일 실시 형태에 따른 근접 통신 시스템에 적용되는 비접촉 IC 카드 내의 메모리 공간의 구조를 모식적으로 도시하는 도면.
- <82> 도 9는 부트스트랩 서비스(Boot Strap Service)의 데이터 구조를 모식적으로 도시하는 도면.
- <83> 도 10은 TOC 엔트리의 데이터 구조를 모식적으로 도시하는 도면.
- <84> 도 11은 제너럴 디스크립션(General Description)의 갱신을 행하기 위한 처리 수순을 설명하기 위한 도면.
- <85> 도 12는 제너럴 디스크립션(General Description)의 갱신을 행하기 위한 처리 수순을 설명하기 위한 도면.
- <86> 도 13은 서비스 디스크립션(Servive Description)의 갱신을 행하기 위한 처리 수순을 설명하기 위한 도면.
- <87> 도 14는 서비스 디스크립션(Servive Description)의 갱신을 행하기 위한 처리 수순을 설명하기 위한 도면.
- <88> 도 15는 서비스 디스크립션(Servive Description)의 추가를 행하기 위한 처리 수순을 설명하기 위한 도면.
- <89> 도 16은 서비스 디스크립션(Servive Description)의 삭제를 행하기 위한 처리 수순을 설명하기 위한 도면.
- <90> 도 17은 TOC군을 라이팅(Writing)으로 할 때의 제기입의 수순을 도시한 플로우차트.
- <91> 도 18은 TOC군을 라이팅(Writing)으로 할 때의 각 TOC에 대한 조작을 도시하는 도면.
- <92> 도 19는 TOC군의 가비지 컬렉션을 행하는 처리 수순을 도시하는 흐름도.
- <93> <발명을 실시하기 위한 최량의 형태>
- <94> 이하, 도면을 참조하면서 본 발명의 실시 형태에 대하여 상세히 설명한다.
- <95> A. 근접 통신 시스템
- <96> 본 발명의 일 실시 형태에 따른 근접 통신 시스템은, 각 디바이스에 IC 카드 및 카드 리더/라이터 쌍방의 기능을 장비함으로써, IC 카드 기술을 범용성이 있는 쌍방향의 근접 통신 인터페이스로서 이용하는 것이다.
- <97> 예를 들면, 컴퓨터나 정보 가전 기기와 같은 기기끼리로 근접 통신 시스템이 구성되는 경우에는, 통신은 일대일로 행해진다. 또, 어떤 기기가 비접촉 IC 카드와 같은 기기 이외의 상대 디바이스(카드라고 부름)와 통신하는 것도 가능하고, 이 경우에는, 1개의 기기와 복수의 카드에서의 일대다의 통신을 행하는 어플리케이션도 생각할 수 있다.
- <98> 이러한 근접 통신 시스템에서는, 사용자가 물리적으로 근접 통신을 행하는 디바이스끼리를 가까이 함으로써, 적어도 한쪽의 기기는 상대의 디바이스를 발견하여 자동으로 동작한다. 단, 반대로 디바이스 사이의 거리가 떨어진 경우, 임의의 시점에서 통신이 두절된다. 이 때문에, 임의의 타이밍에서 사용자가 디바이스끼리를 멀리하는 상황에서도 시스템이 정지하지 않는 동작 보증 기구가 필요하게 된다. 이 동작 보증 기구의 상세에 대해서는 후술한다.
- <99> 본 실시 형태에 따른 근접 통신 시스템에서 제공되는 서비스는, IC 카드가 통신 대상인 경우의 친화성을 고려하여, 기본적으로는 근접 통신을 통하여 디스크립션("Description")의 기입 및 판독을 행하는 메모리 액세스 모델로서 운용된다.
- <100> 여기서, 디스크립션(Description)에는 기기나 IC 카드의 고유한 정보나 각 서비스의 정보가 기술된다. 서비스의 실현에 있어서, 프로세스의 개시 및 디스크립션(Description)의 기입 및 판독에서의 역할을 정하고, 역할이 다른 엔티티 사이에서 여러가지 커맨드가 운용된다.
- <101> 본 실시 형태에서는, 디스크립션(Description)은 XML(eXtended Markup Language)에 의해 기술된다. XML은 사

용자가 독자의 태그를 사용하여 데이터의 속성 정보나 논리 정보를 정의할 수 있는 페이지 기술 언어이다.

#### <102> B. 근접 통신 시스템의 인터랙션 모델

<103> 본 실시 형태에 따른 근접 통신 시스템에서는, 서비스를 실현하는데 있어, 이니시에이터(Initiator), 컨트롤러(Controllor), 타깃(Target)이라는 3개의 역할을 규정한다. 근접 통신 시스템상에서 동작하는 각 디바이스는 필요에 따라 이들의 역할중 몇가지를 실현한다.

<104> 도 1에는, 본 실시 형태에 따른 근접 통신 시스템에 있어서, 각 역할을 갖는 엔티티 사이의 인터랙션 모델을 모식적으로 도시하고 있다.

<105> 이니시에이터는, 근접 통신 인터페이스를 이용하여, 국소에서 다른 디바이스를 탐색하여 컨트롤러 및 타깃을 결정, 임명한다.

<106> 컨트롤러로서 임명된 디바이스는 타깃에 대하여 커맨드를 발행할 수 있다.

<107> 또한, 타깃은 컨트롤러로부터의 커맨드에 응답하여 응답을 돌려 준다. 또한, 타깃은 1 이상의 디스크립션(Description)을 구비하고 있다.

<108> 본 실시 형태에 따른 근접 통신 시스템에서는, IC 카드 및 카드 리더/라이터의 양 기능을 구비한 기기는 이니시에이터, 컨트롤러, 및 타깃 중 어느것도 될 수 있다. 이에 비하여, IC 카드 기능만을 구비한 기기는 타깃밖에 될 수 없다.

<109> 또한, 해당 근접 통신 시스템에서는, 기간 통신에 있어서 일대일 통신을 대상으로 하고 있기 때문에, 실제로 서비스를 실행할 때에, 이니시에이터, 컨트롤러, 타깃의 각 엔티티가 어떤 디바이스에 존재할지에 대한 조합은 몇 개로 한정된다.

<110> · 케이스1

<111> 기기(이니시에이터+컨트롤러) 대 기기 또는 IC 카드(타깃)

<112> · 케이스2

<113> 기기(이니시에이터+타깃) 대 기기(컨트롤러)

<114> 또한, 기기대카드인 경우 일대다의 통신도 있을 수 있지만, 그것은 케이스1의 조합이 복수 있는 것으로 생각된다.

<115> 케이스2는, 서비스의 설계 요구 또는 전원이나 통신로 등의 물리적 조건에 의한 특별한 경우이다. 이 경우, 이니시에이터로부터 컨트롤러에의 이니시에이션은 통신 인터페이스를 통하여 전달된다.

#### <116> C. 디스크립션(Description)

<117> 디스크립션(Description)은, 해당 근접 통신 시스템에 대응하고 있는 타깃의 정보를 나타내는 것이다. 본 실시 형태에서 취급되는 디스크립션(Description)은 제너럴 디스크립션("General Description")과 서비스 디스크립션("Service Description")으로 대별된다. 타깃은 제너럴 디스크립션(General Description)을 반드시 1개 갖고, 대응하고 있는 서비스의 수만큼 서비스 디스크립션(Service Description)을 갖는다. 제너럴 디스크립션(General Description)으로부터 찾아가는 것에 의해서, 서비스 디스크립션(Service Description)을 읽을 수 있다(도 3 참조).

#### <118> D. 커맨드 일람

<119> 커맨드 일람을 이하에 나타낸다.

#### 표 1

명칭	설명
Search	타깃의 탐색
Get Title	명칭을 취득한다
Set Title	명칭을 갱신한다
Get Uri	관련 정보 등의 URI를 취득한다

Set Uri	관련 정보 등의 URI를 갱신한다
Get Icon Uri	아이콘의 URI를 취득한다
Modify Icon	아이콘을 갱신한다
Set Icon	아이콘을 추가한다
Unset Icon	아이콘을 삭제한다
Get Information List	정보의 URI의 리스트를 취득한다
Get Information	정보의 취득
Modify Information	정보의 갱신
Set Information	정보의 추가
Unset Information	정보의 삭제
Send	커맨드/응답의 발행
Listen	커맨드/응답의 수신

- <121> 이하, 각 커맨드에 대하여 설명한다.
- <122> (1) Search
- <123> 검색(Search)은 커넥션의 상대 후보를 얻는 것을 목적으로 하는 커맨드이고, 발견된 타깃의 ID가 출력된다.
- <124> (2) Stand
- <125> 스탠드(Stand)는 이니시에이터로서 입후보하는 것을 목적으로 하는 커맨드이고, 이니시에이터가 될 수 있을 때 커맨드는 성공한다. 커맨드가 실패했을 때, 이니시에이터가 될 수 없다.
- <126> (3) Initiation
- <127> 이니시에이션(Initiation)은, 이니시에이터가 발행하는 커맨드이고, 이니시에이터가 컨트롤러로 하고자 하는 기기에 대하여 타깃의 ID를 전달하는 것으로 그것을 지시한다.
- <128> (4) Get General Description
- <129> 제너럴 디스크립션 취득(Get General Description)은 제너럴 디스크립션(General Description)을 취득하는 것을 목적으로 하는 커맨드이고, 대상으로 하는 타깃으로부터 취득된 XML 데이터가 출력된다.
- <130> (5) Get Service Description
- <131> 서비스 디스크립션 취득(Get Service Description)은, 서비스 디스크립션(Service Description)을 취득하는 것을 목적으로 하는 인터페이스이고, 대상으로 하는 타깃의 ID 및 대상으로 하는 서비스 디스크립션(Service Description)의 URI(Uniform Resource Identifier: 자원 식별자)를 입력으로 제공하는 것에 의해, 취득된 XML 데이터가 출력된다.
- <132> (6) Set General Description
- <133> 제너럴 디스크립션 설정(Set General Description)은, 제너럴 디스크립션(General Description)을 갱신하는 것을 목적으로 하는 인터페이스이고, 대상으로 하는 타깃의 ID와 기입하는 XML 데이터를 입력으로 제공하여, 결과가 저장된다.
- <134> (7) Set Service Description
- <135> 서비스 디스크립션 설정(Set Service Description)은, 서비스 디스크립션(Service Description)을 갱신하는 것을 목적으로 하는 인터페이스이고, 대상으로 하는 타깃의 ID와, 대상으로 하는 서비스 디스크립션(Service Description)의 URI와, 기입하는 XML 데이터를 입력으로 제공하여, 결과가 저장된다.
- <136> (8) Add Service Description 1
- <137> 서비스 디스크립션 추가1(Add Service Description 1)은, 서비스 디스크립션(Service Description)을 기입하는 것을 목적으로 하는 인터페이스이고, 대상으로 하는 타깃의 ID와, 기입하는 서비스 디스크립션(Service Description)의 XML 데이터를 입력으로 제공하며, 추가한 서비스 디스크립션(Service Description)의 URI가 출력으로서 얻어진다.

- <138> (9) Add Service Description 2
- <139> 서비스 디스크립션 추가 2(Add Service Description 2)는, 서비스 디스크립션(Service Description)을 추가(제너럴 디스크립션(General Description)의 갱신 및 서비스 디스크립션(Service Description)의 유효화)하는 것을 목적으로 하는 인터페이스이고, 대상으로 하는 타깃의 ID와 기입하는 제너럴 디스크립션(General Description)의 XML 데이터와, 추가한 서비스 디스크립션(Service Description)의 URI를 입력으로 제공하여, 결과가 저장된다.
- <140> (10) Delete Service Description
- <141> 서비스 디스크립션 소거>Delete Service Description)는, 서비스 디스크립션(Service Description)을 삭제하는 것을 목적으로 하는 인터페이스이고, 대상으로 하는 타깃의 ID와 대상으로 하는 서비스 디스크립션(Service Description)의 URI를 입력으로 제공하여, 결과가 저장된다.
- <142> E. 액세스 수준
- <143> 여기서는, 본 실시 형태에 따른 근접 통신 시스템에서의 디바이스 사이의 액세스 수준에 대하여 설명한다.
- <144> E-1. 이니시에이터의 결정
- <145> 이니시에이터는 대상을 발견하기 위해서 검색(Search) 커맨드를 발행한다.
- <146> 이에 대하여, 검색(Search) 커맨드를 수취한 엔티티는 응답을 돌려 준다.
- <147> 그리고, 다른 엔티티가 컨트롤러의 기능을 갖고 있는 경우에는, 이니시에이터는 Stand를 발행하여, 이니시에이터를 결정한다(도 4 참조).
- <148> E-2. 이니시에이터
- <149> 이니시에이터는 컨트롤러의 기능을 갖게하는 것에 대하여 이니시에이션을 행한다.
- <150> 이니시에이션할 때에는, 컨트롤러에 이용을 원하는 서비스의 URI를 통지한다(도 5 참조).
- <151> E-3. 컨트롤러
- <152> 컨트롤러는, 이니시에이터에 통지된 타깃에 대하여, 제너럴 디스크립션 취득(Get General Description) 커맨드를 발행하여, 그 타깃의 정보를 취득한다.
- <153> 더 필요한 경우에는, 서비스 디스크립션 취득(Get Service Description) 커맨드를 발행하여, 타깃으로부터 서비스 디스크립션(Service Description)을 취득한다(도 6 참조).
- <154> E-4. 상태 천이도
- <155> 도 7에는, 본 실시 형태에 따른 근접 통신 시스템에 있어서 동작하는 디바이스의 상태 천이도를 도시하고 있다.
- <156> 타깃은, 이니시에이션을 수신하면, 이니시에이션 응답을 돌려주고, 컨트롤러로 천이한다.
- <157> 또한, 타깃은, 컨트롤러 기능을 장비하고, 또한 이니시에이터가 되고자 하는 경우에는, 스탠드(Stand) 커맨드를 발행하고, 응답 대기 상태가 된다. 여기서, 거절(Reject)되면, 랜덤의 시간만큼 대기하여 타깃으로 되돌아간다. 또한, 스탠드(Stand) 커맨드가 수리되거나 또는 타임아웃하였을 때에는, 이니시에이터로 천이한다.
- <158> 이니시에이터는, 컨트롤러가 되고자 할 때에는 컨트롤러로 천이할 수 있다. 또, 상대를 컨트롤러로 하고자 할 때는 이니시에이션을 발행하고 타깃으로 천이할 수 있다.
- <159> 컨트롤러는 컨트롤러 기능이 종료되면 타깃으로 되돌아간다.
- <160> F. 메모리 관리
- <161> 앞의 [배경 기술] 부분에서도 이미 진술한 바와 같이, 근접 통신 시스템의 트랜스미션 인터페이스로서 비접촉 IC 카드 기술을 적용하는 경우에는, 사용자가 물리적으로 근접 통신을 행하는 디바이스끼리를 가까이 함으로써, 적어도 한쪽의 기기는 상대의 디바이스를 발견하여 자동으로 동작할 수 있다. 그 반면, 디바이스 사이의 거리가 떨어진 경우, 임의의 시점에서 통신이 두절된다. 이 때문에, 데이터 기입 중의 임의의 타이밍에서 사용자가

디바이스끼리를 멀리하여도 메모리의 내용이 파괴되지 않는 동작 보증 기구가 필요하게 된다.

- <162> 본 실시 형태에서 적용하는 비접촉 IC 카드 시스템에서는, 정보의 기입 및 판독의 단위로서 16 바이트를 1 블록으로 하고, 최대의 동시 기입 사이즈는 8 블록(=128 바이트)이고, 그 이하의 사이즈의 정보의 기입은 1 회의 트랜잭션으로 행해지기 때문에, 동작이 보증되어 있다. 그러나, 그 이상의 크기의 정보를 기입할 때는 정보의 올바른 기입은 보증되지 않는다.
- <163> 따라서, 본 실시 형태에서는, TOC의 구조를 IC 카드 내의 메모리 영역에 구축하여, 최대 동시 기입 사이즈를 초과하는 정보를 기입할때라도, 이것에 수반하는 TOC의 재기입을 1회의 트랜잭션으로 끝내는, 혹은 2회 이상의 트랜잭션에 걸치는 경우에는, 재기입 동작이 도중에서 끊기더라도 이력을 회복할 수 있도록 TOC의 재기입을 행하도록 한다. 이 결과, 기입 중에 임의의 타이밍에서 카드가 떨어지더라도, 카드 내의 정보의 정합성을 유지할 수가 있어, 올바른 기입 동작을 보증할 수 있다.
- <164> F-1. 메모리 공간
- <165> 도 8에는 본 실시 형태에 따른 근접 통신 시스템에 적용되는 비접촉 IC 카드 내의 메모리 공간의 구조를 모식적으로 도시하고 있다.
- <166> 이미 진술한 바와 같이, 타깃으로서 이용되는 비접촉 IC 카드 내에는, 타깃의 정보를 나타내는 디스크립션(Description)의 유지·관리가 행해진다. 그리고, 디스크립션(Description)은 제너럴 디스크립션(General Description)과 서비스 디스크립션(Service Description)으로 대별되고, 타깃은 제너럴 디스크립션(General Description)를 반드시 1개 가짐과 동시에, 대응하고 있는 서비스의 수만큼 서비스 디스크립션(Service Description)를 갖는다.
- <167> 각 디스크립션(Description)은, XML 형식으로 기술된 데이터 본체이고, 각각 가변 길이 또는 고정 길이의 데이터 블록에 기입된다.
- <168> 본 실시 형태에서는, 각 데이터 에리어마다 그 제어 정보인 TOC가 1개씩 설치된다. TOC는 해당하는 데이터 블록의 개시 위치(Start Address of Data Block)를 갖고 있기 때문에, TOC를 기초로 원하는 데이터 블록을 어드레스할 수 있다. 또한, 데이터 블록이 계속인 경우에는, 다음의 TOC의 어드레스 정보(Address of Next TOC)도 TOC에 기입된다.
- <169> 비접촉 IC 카드 시스템이 제공하는 부트스트랩 서비스(Boot Strap Service)는, 제너럴 디스크립션(General Description)의 TOC의 선두 위치에 상당하는 제너럴 디스크립션의 개시 어드레스(Start Address of General Description)를 구비하고 있기 때문에, 부팅 처리 시에 카드의 제너럴 디스크립션(General Description)의 선두 위치를 취득할 수 있다.
- <170> 또한, 제너럴 디스크립션(General Service)은, 대응하는 서비스 디스크립션(Service Description)에의 액세스 방법인 URI를 포함하고 있다.
- <171> 따라서, 도 8에 도시한 바와 같이, 비접촉 IC 카드 내의 메모리 공간에서는, 데이터 블록은 TOC에 의해서 관리되며, 또한 모든 데이터 블록이 어드레스 가능해지도록 구성되어 있다.
- <172> 도 9에는 부트스트랩 서비스(Boot Strap Service)의 데이터 구조를 모식적으로 도시하고 있다. 부트스트랩 서비스(Boot Strap Service)는 D0~Df의 16 바이트로 이루어진다.
- <173> 미디어 타입(Media Type) 필드에는 해당 타깃 디바이스가 컨트롤러가 될 수 있는지의 여부를 나타내는 정보가 기입된다.
- <174> 제너럴 디스크립션의 개시 어드레스(Address of General Description) 필드에는 제너럴 디스크립션(General Description)의 TOC의 스타트 어드레스(Service Code)가 기입되어 있다.
- <175> 엔트리 수(Number of Entries) 필드에는, 본 메모리 공간상에서의 TOC의 엔트리 수가 기입되어 있다.
- <176> 제너럴 디스크립션의 개시 어드레스(Start Address of General Description) 필드에는 제너럴 디스크립션(General Description)의 TOC의 어드레스(Service Code)가 기입되어 있다.
- <177> 부트스트랩 서비스(Boot Strap Service)는 제너럴 디스크립션(General Description)에 관한 정보를 포함하고 있기 때문에, 제너럴 디스크립션(General Description)을 갱신했을 때에는 부트스트랩 서비스(Boot Strap Service)를 재기입할 필요가 있다.



- <178> 도 10에는 TOC 엔트리의 데이터 구조를 모식적으로 도시하고 있다. TOC 엔트리는 D0~Df의 16 바이트로 이루어진다.
- <179> 타입(Type) 필드에는 대응하는 데이터 블록의 상태가 기술된다. 여기서 말하는 상태로서는, 미사용 상태를 나타내는 "Unused", 제너럴 디스크립션(General Description)으로서 사용 중임을 나타내는 "General", 서비스 디스크립션(Service Description)으로서 사용 중임을 나타내는 "Service", 기입 중임을 나타내는 "Writing", 데이터 블록이 소거 가능임을 나타내는 "Erasing"이 정의되어 있다.
- <180> 데이터 블록의 개시 어드레스(Start Address of Data Block) 필드에는, 대응하는 데이터 블록의 스타트 어드레스(Service Code)가 기입되어 있다.
- <181> Number of Blocks 필드에는 대응하는 데이터 블록이 갖는 블록 수가 기입되어 있다.
- <182> 대응하는 디스크립션(Description), 즉 데이터 블록이 계속인 경우에는, 다음 TOC의 어드레스(Address of Next TOC) 필드에는 다음의 TOC의 스타트 어드레스(Service Code)가 기입된다.
- <183> 데이터 사이즈(Data Size) 필드에는 대응하는 데이터 블록 중에 기입되어 있는 데이터의 사이즈가 기입된다.
- <184> F-2. General Description의 취득
- <185> 부트스트랩 서비스(Boot Strap Service)에 포함되는 제너럴 디스크립션의 개시 어드레스(Start Address of General Description)를 읽는 것에 의해, 제너럴 디스크립션(General Description)의 제어 정보인 TOC(Table of Contents)의 선두의 어드레스를 알 수 있다. 제너럴 디스크립션(General Description)의 TOC를 읽는 것에 의해, 타입(Type), 데이터 블록의 개시 어드레스(Start Address of Data Block)나 블록 수(Number of Blocks), 및 다음 TOC의 어드레스(Address of Next TOC)를 알 수 있다. 타입(Type)이 제너럴 디스크립션(General Description)을 나타내고 있으면 정보는 유효하다.
- <186> 데이터는 데이터 블록의 개시 어드레스(Start Address of Data Block)가 가리키는 어드레스부터 블록 수(Number of Blocks) 만큼의 블록에 기록되어 있고, 또한 계속인 경우에는 다음 TOC의 어드레스(Address of Next TOC)에 다음의 TOC의 어드레스가 들어간다. 값이 0이면 이 TOC에서 종료이다.
- <187> F-3. Service Description의 취득
- <188> 지정하는 URI는 서비스 디스크립션(Service Description)의 TOC의 어드레스를 나타내고 있어, 그 영역을 읽는 것에 의해, 타입(Type), 데이터 블록의 개시 어드레스(Start Address of Data Block)나 블록 수(Number of Blocks) 및 다음 TOC의 어드레스(Address of Next TOC)를 알 수 있다. 타입(Type)이 서비스 디스크립션(Service Description)을 나타내고 있으면 정보는 유효하다.
- <189> 데이터는 데이터 블록의 개시 어드레스(Start Address of Data Block)가 가리키는 어드레스부터 블록 수(Number of Blocks) 만큼의 블록에 기록되어 있고, 또한 계속인 경우에는 다음 TOC의 어드레스(Address of Next TOC)에 다음의 TOC의 어드레스가 들어간다. 값이 0이면 이 TOC에서 종료이다.
- <190> F-4. General Description의 갱신
- <191> 기입 중에 통신이 두절되는 상황에서도 데이터의 정합성을 보증하면서 제너럴 디스크립션(General Description)의 갱신을 행하기 위한 처리 수순에 대하여 도 11 및 도 12를 참조하면서 설명한다.
- <192> 단, 도 11은 제너럴 디스크립션(General Description)이 1개의 데이터 블록만을 사용하는 경우를 나타내고, 도 12는 제너럴 디스크립션(General Description)이 2 이상의 데이터 블록에 걸치는 경우를 나타내고 있다.
- <193> (1) 우선, 메모리 공간상에서 미사용의 데이터 블록을 찾아, 새로운 제너럴 디스크립션(General Description)을 기입하는 만큼의 용량이 있음을 확인한다(도 11 참조). 제너럴 디스크립션(General Description)이 2 이상의 데이터 블록에 걸쳐있는 경우에는, 해당하는 개수만큼 미사용의 데이터 블록을 탐색한다(도 12 참조).
- <194> 미사용 데이터 블록의 탐색 처리는, 타입(Type) 필드가 미사용(Unused)으로 설정되어 있는 TOC를 탐색하는 것과, 그 블록 수(Number of Blocks)를 참조함으로써 실현된다.
- <195> 또한, 찾아낸 데이터 블록이 갱신후에 새로운 제너럴 디스크립션(General Description)의 사이즈 이상의 빈 용량이 되는 것을 확인한다. 이것은 제너럴 디스크립션(General Description) 갱신후, 그 이상의 갱신이 불가능하게 되지 않기 위한 확인이다.

- <196> (2) 계속해서, 데이터 블록을 제너럴 디스크립션(General Description)의 재기입에 앞서 할당하기 위해서, 사용하고자하는 데이터 블록의 TOC의 타입(Type)을 Unused(미사용)로부터, Writing(기입 중)으로 변경함과 동시에, 그 밖의 파라미터를 세트한다(도 11 참조).
- <197> 또한, 제너럴 디스크립션(General Description)이 2 이상의 데이터 블록에 걸쳐 있는 경우에는, 선두 TOC의 Type을 Unused(미사용)로부터 Writing(기입 중)으로 함과 동시에, 선두 TOC 이외의 타입(Type)을 제너럴(General)로 변경함과 동시에, 그 밖의 파라미터를 세트한다. 복수의 영역에 걸치는 경우, 다음의 TOC를 순차 찾아갈 수 있도록 파라미터를 세트한다(도 12 참조).
- <198> 여기서의 TOC의 재기입 조작은 1회의 트랜잭션으로 행하도록 한다. 1회의 트랜잭션에서는, 비접촉 IC 카드에서 기입이 보증되어 있기 때문에, TOC의 데이터정합성이 유지된다. 그 후의 데이터 블록의 기입 시에 통신이 두절 되어도, TOC를 기초로 회복할 수 있다.
- <199> 제너럴 디스크립션(General Description)이 복수(9 이상)의 데이터 블록에 걸치기 위해서, 모든 TOC의 재기입 조작이 1회의 트랜잭션으로서는 완료되지 않는 경우에는, 재기입 동작이 도중에서 끊기더라도 이력을 회복할 수 있도록 TOC의 재기입을 행하도록 하며, 이 점은 후에 상세히 설명한다.
- <200> (3) 계속해서, 새로운 제너럴 디스크립션(General Description)의 내용을 (1)에서 할당한 데이터 블록에 기입한다.
- <201> 제너럴 디스크립션(General Description)은 XML 형식으로 기술되어 있지만(전술), 그 데이터 사이즈는, 통상, 비접촉 IC 카드에서 기입이 보증되어 있는 최대의 동시 기입 사이즈를 초과하는 대용량이다. 이 때문에, 제너럴 디스크립션(General Description)의 기입 시에 임의의 타이밍에서 디바이스끼리 사이가 멀어져 통신이 두절 되어, 데이터의 내용이 파괴될 가능성이 있다. 단, TOC를 기초로 통신의 이력을 찾아가는 것에 의해서, 데이터의 회복을 행할 수 있다.
- <202> (4) 계속해서, 새롭게 기입한 데이터 블록의 TOC의 타입(Type)을 Writing(기입 중)으로부터 제너럴(General)로 갱신함과 함께, 이전의 제너럴 디스크립션(General Description)의 TOC의 타입(Type)을 Erasing(소거가능)으로 한다. 이와 동시에, 부트스트랩(Boot Strap)의 제너럴(General)을 가리키는 파라미터를 새로운 제너럴 디스크립션(General Description)의 TOC를 가리키도록 갱신한다(도 11 참조).
- <203> 또한, 제너럴 디스크립션(General Description)이 2 이상의 데이터 블록에 걸쳐있는 경우에는, 선두 TOC의 Writing(기입 중)을 제너럴(General)로 갱신하고, 이전의 제너럴 디스크립션(General Description)의 선두 TOC의 타입(Type)을 Erasing(소거가능)으로 갱신함과 동시에, 부트스트랩(Boot Strap)의 제너럴(General)을 가리키는 파라미터를 새로운 제너럴 디스크립션(General Description)의 TOC를 가리키도록 갱신한다(도 12 참조).
- <204> 여기서의 TOC의 재기입 조작은 1회의 트랜잭션으로 행한다. 1회의 트랜잭션에서는, 비접촉 IC 카드에서 기입이 보증되어 있기 때문에, TOC의 데이터 정합성이 유지된다. 이 경우, 신구의 TOC의 재기입이 동시에 행해지기 때문에, 제너럴 디스크립션(General Description)은 통신의 두절의 영향을 받는 경우가 없다. 제너럴 디스크립션(General Description)이 복수(9 이상)의 데이터 블록에 걸치기 때문에, 모든 TOC의 재기입 조작이 1회의 트랜잭션으로서는 완료되지 않는 경우에는, 재기입 동작이 도중에서 끊기더라도 이력을 회복할 수 있도록 TOC의 재기입을 행하도록 한다(동상).
- <205> (5) 마지막으로, TOC의 타입(Type)이 Erasing(소거가능)으로 되어 있는 TOC, 및 거기에서 찾아갈 수 있는 TOC의 Type을 전부 Unused(미사용)로 한다(도 11 및 도 12 참조). Unused(미사용)로 복귀하는 것에 의해, 이들 TOC에 대응하는 데이터 블록이 가장 이용 가능하게 된다.
- <206> F-5. Service Description의 갱신
- <207> 기입 중에 통신이 두절되는 상황에서도 데이터의 정합성을 보증하면서 서비스 디스크립션(Service Description)의 갱신을 행하기 위한 처리 수순에 대하여 도 13 및 도 14를 참조하면서 설명한다. 단, 도 13은 서비스 디스크립션(Service Description)이 1개의 데이터 블록만을 사용하는 경우를 도시하고, 도 14는 서비스 디스크립션(Service Description)이 2 이상의 데이터 블록에 걸치는 경우를 도시하고 있다.
- <208> (1) 우선, 메모리 공간상에서 미사용의 데이터 블록을 찾아, 새로운 서비스 디스크립션(Service Description)을 기입할 만큼의 용량이 있음을 확인한다(도 13 참조). 서비스 디스크립션(Service Description)이 2 이상의 데이터 블록에 걸쳐 있는 경우에는, 해당하는 개수만큼 미사용의 데이터 블록을 탐색한다(도 14 참조).

- <209> 미사용 데이터 블록의 탐색 처리는, 타입(Type) 필드가 Unused(미사용)로 설정되어 있는 TOC를 탐색하는 것과, 그 블록 수(Number of Blocks)를 참조함으로써 실현된다. 또한, 찾아낸 데이터 블록이 갱신후에 새로운 제너럴 디스크립션(General Description)의 사이즈 이상의 빈 용량이 되는 것을 확인한다. 이것은 제너럴 디스크립션(General Description) 갱신후, 그 이상의 갱신이 불가능하게 되지 않기 위한 확인이다.
- <210> (2) 계속해서, 데이터 블록을 서비스 디스크립션(Service Description)의 재기입에 앞서 할당하기 위해서, 사용하고자 하는 데이터 블록의 TOC의 타입(Type)을 Unused(미사용)로부터 Writing(기입 중)으로 변경함과 동시에, 그 밖의 파라미터를 세트한다(도 13 참조).
- <211> 또한, 서비스 디스크립션(Service Description)이 2 이상의 데이터 블록에 걸쳐 있는 경우에는, 선두 TOC의 타입(Type)을 Unused(미사용)로부터 Writing(기입 중)으로 함과 함께, 선두 TOC 이외의 타입(Type)을 서비스(Service)로 변경함과 동시에 그 밖의 파라미터를 세트한다. 복수의 영역에 걸치는 경우, 다음의 TOC를 순차적으로 하는 파라미터를 세트한다(도 14 참조).
- <212> 여기서의 TOC의 재기입 조작은, 1회의 트랜잭션으로 행하기 때문에, TOC의 데이터 정합성이 유지되어, 그 후의 데이터 블록의 기입 시에 통신이 두절된다고 해도 TOC를 기초로 회복할 수 있다. 서비스 디스크립션(Service Description)이 복수(9 이상)의 데이터 블록에 걸치기 때문에, 모든 TOC의 재기입 조작이 1회의 트랜잭션으로는 완료되지 않는 경우에는, 재기입 동작이 도중에서 끊기더라도 이력을 회복할 수 있도록 TOC의 재기입을 행하도록 한다.
- <213> (3) 계속해서, 새로운 서비스 디스크립션(Service Description)의 내용을 (1)에서 할당한 데이터 블록에 기입한다.
- <214> 서비스 디스크립션(Service Description)은 XML 형식으로 기술되어 있지만(전술), 그 데이터 사이즈는, 통상 비접촉 IC 카드에서 기입이 보증되어 있는 최대의 동시 기입 사이즈를 초과하는 대용량이다. 이 때문에, 제너럴 디스크립션(General Description)의 기입 시에 임의의 타이밍에서 디바이스끼리가 떨어져 통신이 두절되어, 데이터의 내용이 파괴될 가능성이 있다. 단, TOC를 기초로 통신의 이력을 찾아가는 것에 의해서, 데이터의 회복을 행할 수 있다.
- <215> (4) 계속해서, 새롭게 기입한 데이터 블록의 TOC의 정보를 이전의 서비스 디스크립션(Service Description)의 TOC의 정보와 교체하여, 새로운 서비스 디스크립션(Service Description)의 TOC의 타입(Type)을 Writing(기입 중)으로부터 서비스(Service)로 갱신함과 동시에, 이전의 서비스 디스크립션(Service Description)의 TOC의 타입(Type)을 Erasing(소거가능)으로 한다(도 13 참조).
- <216> 또한, 서비스 디스크립션(Service Description)이 2 이상의 데이터 블록에 걸쳐 있는 경우에는, 서비스 디스크립션(Service Description)의 선두 TOC의 정보를 이전의 서비스 디스크립션(Service Description)의 선두 TOC의 정보와 교체하여, 새로운 서비스 디스크립션(Service Description)의 TOC의 타입(Type)을 Writing(기입 중)으로부터 서비스(Service)로 갱신함과 동시에, 이전의 서비스 디스크립션(Service Description)의 TOC의 타입(Type)을 Erasing(소거가능)으로 한다(도 14 참조).
- <217> 서비스 디스크립션(Service Description)의 TOC는, 제너럴 디스크립션(General Description)으로부터 URI로 참조되어 있다. 서비스 디스크립션(Service Description)의 갱신 시에 제너럴 디스크립션(General Description)의 재기입을 행하지 않고 끝나도록, 새로운 서비스 디스크립션(Service Description)의 기입을 행한 후, 친구의 TOC의 교체를 행하여, TOC에의 참조를 변경하지 않도록 하고 있다.
- <218> 여기서의 TOC의 재기입 조작은 1회의 트랜잭션으로 행한다. 1회의 트랜잭션에서는, 비접촉 IC 카드에서 기입이 보증되어 있기 때문에, TOC의 데이터 정합성이 유지된다. 이 경우, 친구의 TOC의 교체가 동시에 행해지기 때문에, 제너럴 디스크립션(General Description)은 통신 두절의 영향을 받지 않는다. 제너럴 디스크립션(General Description)이 복수(9 이상)의 데이터 블록에 걸치기 때문에, 모든 TOC의 재기입 조작이 1회의 트랜잭션으로는 완료되지 않는 경우에는, 재기입 동작이 도중에서 끊기더라도 이력을 회복할 수 있도록 TOC의 재기입을 행하도록 한다(동상).
- <219> (5) 마지막으로, TOC의 타입(Type)이 Erasing(소거가능)으로 되어 있는 TOC, 및 거기에서부터 찾아갈 수 있는 TOC의 타입(Type)을 전부 Unused(미사용)로 한다(도 13 및 도 14 참조). Unused(미사용)로 복귀하는 것에 의해, 이들 TOC에 대응하는 데이터 블록이 가장 이용 가능하게 된다.
- <220> F-6. Service Description의 추가



- <221> 기입 중에 통신이 두절하는 상황에서도 데이터의 정합성을 보증하면서 서비스 디스크립션(Service Description)의 추가를 행하기 위한 처리 수순에 대하여 도 15를 참조하면서 설명한다.
- <222> (1) 우선, 메모리 공간 수로 미사용의 데이터 블록을 찾아, 새로운 서비스 디스크립션(Service Description) 및 제너럴 디스크립션(General Description)을 쓸 만큼의 용량이 있음을 확인한다.
- <223> 또한, 갱신후에 새로운 제너럴 디스크립션(General Description)의 사이즈와 동일한 만큼의 빈 용량이 되는 것을 확인한다. 이것은 서비스 디스크립션(Service Description) 추가후, 그 이상의 갱신이 불가능하게 되지 않기 위한 확인이다.
- <224> (2) 계속해서, 새로운 서비스 디스크립션(Service Description)에 사용하고자 하는 영역을 할당하기 위해서, 사용하고자 하는 영역군의 선두 TOC의 타입(Type)을 Unused(미사용)로부터 Writing(기입 중)으로 함과 아울러, 선두 TOC 이외의 타입(Type)을 서비스(Service)로 변경함과 동시에, 그 밖의 파라미터를 세트한다. 복수의 영역에 걸치는 경우에는, 다음의 TOC를 찾아갈 수 있도록 한다.
- <225> 여기서의 TOC이 재기입 조작은, 1회의 트랜잭션으로 행하기 때문에, TOC의 데이터 정합성이 유지되어, 그 후의 데이터 블록의 기입 시에 통신이 두절된다고 해도 TOC를 기초로 회복할 수 있다. 또한, 서비스 디스크립션(Service Description)이 복수(9 이상)의 데이터 블록에 걸치기 때문에, 모든 TOC이 재기입 조작이 1회의 트랜잭션으로는 완료되지 않는 경우에는, 재기입 동작이 도중에서 끊기더라도 이력을 회복할 수 있도록 TOC의 재기입을 행하도록 한다.
- <226> (3) 계속해서, 새로운 서비스 디스크립션(Service Description)의 내용을 (1)에서 할당한 각 데이터 블록에 기입한다.
- <227> 서비스 디스크립션(Service Description)은 XML 형식으로 기술되어 있지만(전술), 그 데이터 사이즈는, 통상 비접촉 IC 카드에서 기입이 보증되어 있는 최대의 동시 기입 사이즈를 초과하는 대용량이다. 이 때문에, 제너럴 디스크립션(General Description)의 기입 시에 임의의 타이밍에서 디바이스끼리가 떨어져 통신이 두절되어, 데이터의 내용이 파괴될 가능성이 있다. 단, TOC를 기초로 통신의 이력을 찾아가는 것에 의해서, 데이터의 회복을 행할 수 있다.
- <228> (4) 계속해서, 이하에 도시하는 수순에 따라서 제너럴 디스크립션(General Description)에 서비스(Service)를 추가한다.
- <229> (4-1) 서비스(Service)를 추가한 새로운 제너럴 디스크립션(General Description)을 준비하기 위한 미사용의 데이터 블록을 탐색한다.
- <230> (4-2) 계속해서, 데이터 블록을 제너럴 디스크립션(General Description)의 재기입에 앞서 할당하기 위해서, 사용하고자 하는 데이터 블록군의 선두 TOC의 타입(Type)을 Unused(미사용)로부터 Writing(기입 중)으로 함과 동시에, 선두 TOC 이외의 타입(Type)을 제너럴(General)로 변경함과 동시에, 그 밖의 파라미터를 세트한다. 복수의 데이터 블록에 걸치는 경우에는 다음의 TOC를 찾아갈 수 있도록 한다.
- <231> 여기서의 TOC의 재기입 조작은 1회의 트랜잭션으로 행하도록 함으로써 TOC의 데이터 정합성이 유지된다. 그 후의 데이터 블록의 기입 시에 통신이 두절된다고 해도 TOC를 기초로 회복할 수 있다. 또한, 제너럴 디스크립션(General Description)이 복수(9 이상)의 데이터 블록에 걸치기 때문에, 모든 TOC의 재기입 조작이 1회의 트랜잭션으로는 완료되지 않는 경우에는, 재기입 동작이 도중에서 끊기더라도 이력을 회복할 수 있도록 TOC의 재기입을 행하도록 한다.
- <232> (4-3) 할당한 각 데이터 블록에 새로운 제너럴 디스크립션(General Description)의 내용을 기입한다.
- <233> 제너럴 디스크립션(General Description)은 비접촉 IC 카드에서 기입이 보증되어 있는 최대의 동시 기입 사이즈를 초과하는 대용량이기 때문에, 제너럴 디스크립션(General Description)의 기입 시에 임의의 타이밍에서 디바이스끼리가 떨어져 통신이 두절되어, 데이터의 내용이 파괴될 가능성이 있다. 단, TOC를 기초로 통신의 이력을 찾아가는 것에 의해서, 데이터의 회복을 행할 수 있다.
- <234> (4-4) 새롭게 기입한 데이터 블록의 선두 TOC의 타입(Type)을 Writing(기입 중)으로부터 제너럴(General)로 갱신하고, 이전의 제너럴 디스크립션(General Description)의 선두 TOC의 타입(Type)을 Erasing(소거가능)으로 함과 동시에, 부트스트랩(Boot Strap)의 제너럴(General)을 가리키는 파라미터를 새로운 제너럴 디스크립션(General Description)의 TOC를 가리키도록 갱신한다. 그 때 동시에, 새롭게 추가한 Service의 선두 TOC의 타

입(Type)을 Writing(기입 중)으로부터 서비스(Service)로 갱신한다.

- <235> 여기서의 TOC의 재기입 조작은, 1회의 트랜잭션으로 행하기 때문에, TOC의 데이터 정합성이 유지된다. 이 경우, 신구의 TOC의 재기입이 동시에 행해지기 때문에, 통신의 두절의 영향을 받는 경우는 없다. 또, 재기입 대상이 되는 Description이 복수(9 이상)의 데이터 블록에 걸치기 때문에, 모든 TOC의 재기입 조작이 1회의 트랜잭션으로는 완료되지 않는 경우에는, 재기입 동작이 도중에서 끊기더라도 이력을 회복할 수 있도록 TOC의 재기입을 행하도록 한다(동상).
- <236> (5) 마지막으로, TOC의 타입(Type)이 Erasing(소거가능)으로 되어 있는 TOC, 및 거기에서부터 찾아갈 수 있는 TOC의 타입(Type)을 전부 Unused(미사용)로 한다. Unused(미사용)에 복귀하는 것에 의해, 이들 TOC에 대응하는 데이터 블록이 가장 이용 가능하게 된다.
- <237> 상술한 서비스 디스크립션(Service Description)의 추가 처리는, 서비스 디스크립션(Service Description)의 갱신과 제너럴 디스크립션(General Description)의 재기입의 각 처리 수순의 조합에 따라서 실현된다.
- <238> F-7. Service Description의 삭제
- <239> 기입 중에 통신이 두절되는 상황에도 데이터의 정합성을 보증하면서 서비스 디스크립션(Service Description)의 삭제를 행하기 위한 처리 수순 대하여 도 16을 참조하면서 설명한다.
- <240> (1) 이하에 도시하는 수순에 따라 제너럴 디스크립션(General Description)에서 서비스(Service)를 삭제한다.
- <241> (1-1) 서비스(Service)를 삭제한 새로운 제너럴 디스크립션(General Description)을 준비한다.
- <242> (1-2) 메모리 공간 수로 미사용의 데이터 블록을 찾아, 새로운 제너럴 디스크립션(General Description)을 기입할 만큼의 용량이 있음을 확인한다.
- <243> 또한, 갱신후에 새로운 제너럴 디스크립션(General Description)의 사이즈와 동일한 만큼의 빈 용량이 되는 것을 확인한다. 이것은 제너럴 디스크립션(General Description) 갱신후, 그 이상의 갱신이 불가능하게 되지 않기 위한 확인이다.
- <244> (1-3) 계속해서, 데이터 블록을 할당하기 위해서, 사용하고자 하는 데이터 블록군의 선두 TOC의 타입(Type)을 Unused(미사용)로부터 Writing(기입 중)으로 함과 아울러, 선두 TOC 이외의 타입(Type)을 제너럴(General)로 변경함과 동시에, 그 밖의 파라미터를 세트로 한다. 복수의 데이터 블록에 걸치는 경우에는 다음의 TOC를 찾아갈 수 있도록 한다.
- <245> 여기서의 TOC의 재기입 조작은, 1회의 트랜잭션으로 행하도록 함으로써, TOC의 데이터 정합성이 유지된다. 그 후의 데이터 블록의 기입 시에 통신이 두절된다고 해도 TOC를 기초로 회복할 수 있다. 또한, 제너럴 디스크립션(General Description)이 복수(9 이상)의 데이터 블록에 걸치기 때문에, 모든 TOC의 재기입 조작이 1회의 트랜잭션으로는 완료되지 않는 경우에는, 재기입 동작이 도중에서 끊기더라도 이력을 회복할 수 있도록 TOC의 재기입을 행하도록 한다.
- <246> (1-4) 할당한 각 데이터 블록에 새로운 제너럴 디스크립션(General Description)의 내용을 기입한다.
- <247> 제너럴 디스크립션(General Description)은 비접촉 IC 카드에서 기입이 보증되어 있는 최대의 동시 기입 사이즈를 초과하는 대용량이기 때문에, 제너럴 디스크립션(General Description)의 기입 시에 임의의 타이밍에서 디바이스끼리 떨어져 통신이 두절되어, 데이터의 내용이 파괴될 가능성이 있다. 단, TOC를 기초로 통신의 이력을 찾아가는 것에 의해서, 데이터의 회복을 행할 수 있다.
- <248> (1-5) 새로운 데이터 블록의 선두 TOC의 타입(Type)을 Writing(기입 중)으로부터 제너럴(General)로 갱신하여, 이전의 제너럴 디스크립션(General Description)의 선두 TOC의 타입(Type)을 Erasing(소거가능)으로 함과 아울러, 부스트스트랩 서비스(Boot Strap Service)의 제너럴(General)을 가리키는 파라미터를 새로운 제너럴 디스크립션(General Description)의 TOC를 가리키도록 갱신한다. 이 때 동시에, 삭제하는 데이터 블록의 TOC의 타입(Type)을 서비스(Service)로부터 Erasing(소거가능)으로 갱신한다.
- <249> 여기서의 TOC의 재기입 조작은, 1회의 트랜잭션으로 행하기 때문에, TOC의 데이터 정합성이 유지된다. 이 경우, 신구의 TOC의 재기입이 동시에 행해지기 때문에, 통신의 두절의 영향을 받는 경우가 없다. 또, 재기입 대상이 되는 Description이 복수(9 이상)의 데이터 블록에 걸치기 때문에, 모든 TOC의 재기입 조작이 1회의 트

랜잭션으로는 완료되지 않는 경우에는, 재기입 동작이 도중에서 끊기더라도 이력을 회복할 수 있도록 TOC의 재기입을 행하도록 한다(동상).

- <250> (2) 마지막으로, TOC의 Type이 Erasing(소거가능)으로 되어 있는 TOC, 및 거기에서부터 찾아갈 수 있는 TOC의 타입(Type)을 전부 Unused(미사용)로 한다. Unused(미사용)로 복귀하는 것에 의해, 이들 TOC에 대응하는 데이터 블록이 가장 이용 가능하게 된다.
- <251> 상술한 서비스 디스크립션(Service Description)의 삭제 처리는, 기본적으로는, 해당하는 제너럴 디스크립션(General Description)으로부터 서비스 디스크립션(Service Description)에의 링크를 삭제한다고 하는, 제너럴 디스크립션(General Description)을 갱신하는 처리 수순에 의해서 실현된다.
- <252> F-8. TOC군의 Type을 Writing으로 할때의 재기입 수순
- <253> 본 실시 형태에 따른 근접 통신 시스템의 트랜스미션 인터페이스에 비접촉 IC 카드를 적용하는 경우, 정보의 기입 및 관독의 단위로서 16 바이트를 1 블록으로 하여, 최대의 동시 기입 사이즈는 8 블록(=128 바이트)이고, 그 사이즈의 정보의 기입은 보증되어 있다. 따라서, 데이터 즉 Description을 갱신, 추가, 삭제하는 경우, 이것에 수반하는 모든 TOC의 재기입을 1회의 트랜잭션으로 끝내는 것에 의해, 데이터의 보증을 도모한다.
- <254> 그런데, TOC는 16 바이트 즉 1 블록으로 구성되기 때문에, Description의 갱신, 추가, 삭제가 복수의 데이터 블록에 걸쳐 행해지고, 갱신하는 TOC가 9 이상인 경우에는, 1회의 트랜잭션만으로 전부를 재기입하는 것은 불가능하다.
- <255> 따라서, 본 실시 형태에서는, TOC의 재기입이 2회 이상의 트랜잭션에 걸치는 경우에는, 통신이 도중에서 두절되어도 이력을 회복할 수 있도록 TOC의 재기입을 행하도록 TOC의 재기입 조작을 행한다. 이 결과, 기입 중에 임의의 타이밍에서 카드가 떨어져도, 카드 내의 정보의 정합성을 유지할 수가 있어, 올바른 기입 동작을 보증할 수 있다.
- <256> 상술한 바와 같이, 복수의 데이터 블록에 걸치는 디스크립션(Description)을 갱신, 추가, 삭제하는 경우, 선두 TOC의 타입(Type)은 Writing(기입 중)으로 하고, 선두 이외의 TOC의 타입(Type)은 제너럴(General)(혹은 서비스(Service))로 한다.
- <257> 그리고, 갱신하는 TOC가 9개 이상인 경우에는, 8 블록씩의 기입으로 분할한다. n회째의 기입에서는(단지 n은 1 이상의 정수), (7n-6)번째부터 7n번째까지의 TOC의 다음 TOC의 어드레스(Address of Next TOC)는 각각 다음의 TOC를 가리키도록 하고, (7n+1)번째의 TOC의 다음 TOC의 어드레스(Address of Next TOC)는 0(널)으로 한다. 그리고, 다음회에 (7n+1)번째의 TOC부터 기입할 때에 올바른 어드레스를 기입하도록 한다.
- <258> 이와 같이, 각 TOC 재기입 트랜잭션마다, 최후미의 TOC는 다음의 TOC를 가리키지 않은채로 뒀으로써, 이 트랜잭션이 통신의 두절 등에 의해 실패해도, 불필요한 TOC의 지정을 행하지 않고 끝난다.
- <259> 한편, 최후미의 TOC가 다음의 TOC를 가리키도록 한 경우에는, 트랜잭션에 실패한 후에는, 다음의 TOC는 타입(Type)이 Unused(미사용)이고, 다른 트랜잭션에 의해 전혀 관계가 없는 데이터가 기입되어 버린 후에도 TOC의 링크가 형성되어 있기 때문에, 데이터의 부정합을 초래하게 된다.
- <260> 도 17에는, TOC군을 Writing(기입 중)으로 할 때의 재기입의 수순을 흐름도의 형식으로 나타내고 있다. 또한, 도 18에는, 이 때의 각 TOC에 대한 조작을 도해하고 있다.
- <261> 먼저, 갱신하는 TOC가 9개 이상인지, 즉 1회의 트랜잭션으로 모든 TOC의 갱신을 완료할 수 없는지를 판단한다(단계 S1).
- <262> 여기서, 갱신하는 TOC가 8개 이하, 즉 1회의 트랜잭션으로 모든 TOC의 갱신을 완료할 수 있는 경우에는, 최후의 TOC 이외의 다음 TOC의 어드레스(Address of Next TOC)는 각각 다음의 TOC를 가리키도록 함과 함께, 최후의 TOC는, 어디도 가리키지 않도록 0(널)을 대입한다(단계 S2). 이들 TOC의 재기입 동작은 1회의 트랜잭션만으로 행한다.
- <263> 한편, 갱신하는 TOC가 9개 이상 있는 경우에는, 1번째의 TOC의 타입(Type)을 Writing(기입 중)으로 함과 동시에, 다른 TOC의 타입을 제너럴(General)(또는 서비스(Service))로 한다. 또한, 8번째의 TOC 이외의 다음 TOC의 개시 어드레스(Address of Next TOC)를 다음의 TOC를 가리키도록 함과 함께, 8번째의 TOC에 대해서는 어디도 가리키지 않도록 0을 대입한다(단계 S3). 이들 TOC의 재기입 동작은 1회의 트랜잭션(T1)만으로 행한다.

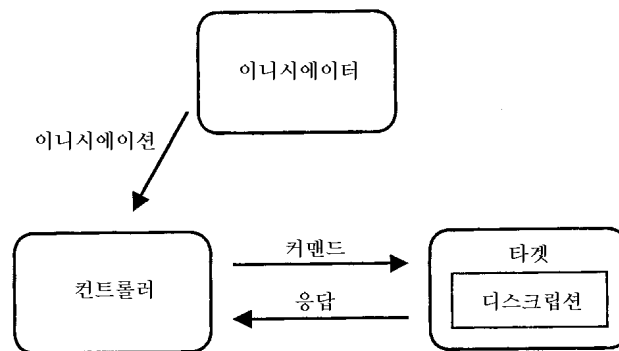
- <264> 이하, 현재 8번째의 TOC 이후에 대하여 처리를 행한다(단계 S4).
- <265> 갱신하는 TOC가 아직 9개 이상 남아 있는지, 즉 1회의 트랜잭션으로 모든 TOC의 갱신을 갱신할 수 없는지의 여부를 판단한다(단계 S5).
- <266> 여기서, 갱신하는 TOC가 8개 이하, 즉 1회의 트랜잭션으로 모든 TOC의 갱신을 갱신할 수 있는 경우에는, 최후의 TOC 이외의 다음 TOC의 어드레스(Address of Next TOC)는 각각 다음의 TOC를 가리키도록 함과 함께, 최후의 TOC는 어디도 가리키지 않도록 0(널)을 대입한다(단계 S2).
- <267> 한편, 갱신하는 TOC가 9개 이상 있는 경우에는, 현재의 TOC의 타입(Type)을 Writing(기입 중)으로 함과 함께, 다른 TOC의 타입을 General(제너럴)(또는 서비스(Service))로 한다. 또한, 8번째의 TOC 이외의 다음 TOC의 어드레스(Address of Next TOC)를 다음의 TOC를 가리키도록 함과 함께, 8번째의 TOC에 대해서는 어디도 가르키지 않도록 0을 대입한다(단계 S6). 이들 TOC의 재기입 동작은 1회의 트랜잭션(T2)만으로 행한다.
- <268> 그 후, 단계 S4로 되돌아가, 모든 TOC에 대하여 처리가 끝날 때까지, 상술한 바와 마찬가지로의 TOC 재기입 동작을 반복하여 실행한다.
- <269> F-9. 가비지 컬렉션
- <270> 본 실시 형태에 따른 근접 통신 시스템의 트랜스미션 인터페이스에 비접촉 IC 카드를 적용하는 경우, 데이터 즉 디스크립션(Description)의 갱신, 추가, 삭제를 행하기 위해서는, 새로운 데이터 블록에 데이터를 기입하고 나서, 오래된 데이터 블록과의 교체를 행한다.
- <271> 그리고, 이러한 데이터 재기입의 사이, 통신의 두절에 수반하는 데이터를 보증하기 위해서, 신구의 데이터 블록의 TOC의 타입(Type)을 데이터 재기입 수순의 진척 상황에 대응하여 순차 갱신하도록 되어 있다. 즉, 새로운 데이터 블록의 TOC의 타입(Type)을, Unused(미사용)으로부터 Writing(기입 중)을 지나서, 제너럴(General) 혹은 서비스(Service)로 천이한다(단, 선두 TOC인 경우). 또한, 원래의 데이터 블록의 TOC의 Type은 제너럴(General) 혹은 서비스(Service)로부터 Erasing(소거가능)으로 천이한다(단, 선두 TOC인 경우).
- <272> 이와 같이 데이터의 재기입 동작이 성공한 뒤에 종료할 때까지는, TOC의 타입(Type)으로서 Writing(기입 중)이나 Erasing(소거가능) 등의 중간적인 상태를 제공함으로써, 트랜잭션 중에 통신이 두절되어도, TOC의 링크를 찾아 기입에 실패한 데이터 블록을 추적하여 특정할 수 있다.
- <273> 또한, TOC의 타입(Type)이 Writing(기입 중) 또는 Erasing(소거가능)으로 되어 있는 데이터 블록은, 통신의 두절 등에 의해 사용할 수 없음을 나타내고 있기 때문에, 개방하여 재이용한다. 즉, TOC의 타입(Type)이 Writing(기입 중) 또는 Erasing(소거가능)의 TOC에서부터 찾아갈 수 있는 TOC군을 Unused(미사용)로 한다. 이와 같이 미사용의 데이터 블록을 확보하기 위한 처리를, 본 명세서에서는 「가비지 컬렉션」(Garbage Collection)이라고 부른다.
- <274> 데이터 블록의 갱신, 추가, 삭제의 과정에서 통신의 두절 등에 의해 데이터 기입 동작이 실패로 끝난 경우, 선두 TOC의 타입(Type)은 Writing(기입 중) 또는 Erasing(소거가능)으로, 선두 이외의 TOC의 타입(Type)은 제너럴(General)(또는 서비스(Service))로 되어 있다(도 12, 도 14, 도 15 및 도 16 참조). 이들은 전부 가비지 컬렉션의 대상으로, Unused(미사용)로 변경하여 데이터 블록의 재이용을 도모한다.
- <275> 본 실시 형태에 적용되는 비접촉 IC 카드 시스템의 사양으로서 8 블록씩밖에 기입 보증이 되지 않기 때문에, 갱신하는 TOC가 9개 이상인 경우, 8 블록씩의 기입으로 분할할 필요가 있다. 따라서, n회째의 기입으로는 7n번째까지의 TOC의 타입(Type)을 Unused(미사용)로, (7n+1)번째의 TOC의 타입(Type)을 Erasing(소거가능)으로 한다. 다음 (n+1)회째의 기입으로는 (7n+1)번째의 TOC에서부터 7개의 TOC의 타입(Type)을 Unused(미사용)로, (7n+8)번째의 TOC의 타입(Type)을 Erasing(소거가능)으로 한다.
- <276> 이와 같이, 가비지 컬렉션에서의 각 TOC 재기입 트랜잭션마다, 최후미의 TOC의 타입(Type)을 Unused(미사용)로 하지 않고, Erasing(소거가능), 즉 가비지 컬렉션을 행하고 있지 않는 상태로 해 둬으로써, 트랜잭션이 통신의 두절 등에 의해 실패해도, Erasing(소거가능) 상태의 최후미의 TOC를 기점으로 하여 가비지 컬렉션을 재개할 수 있다.
- <277> 도 19에는 가비지 컬렉션할 때의 TOC군의 재기입의 수순을 흐름도의 형식으로 도시하고 있다.
- <278> 먼저, 갱신하는 TOC가 9개 이상인지, 즉, 1회의 트랜잭션으로 모든 TOC의 갱신을 완료할 수 없는지의 여부를 판

단한다(단계 S11).

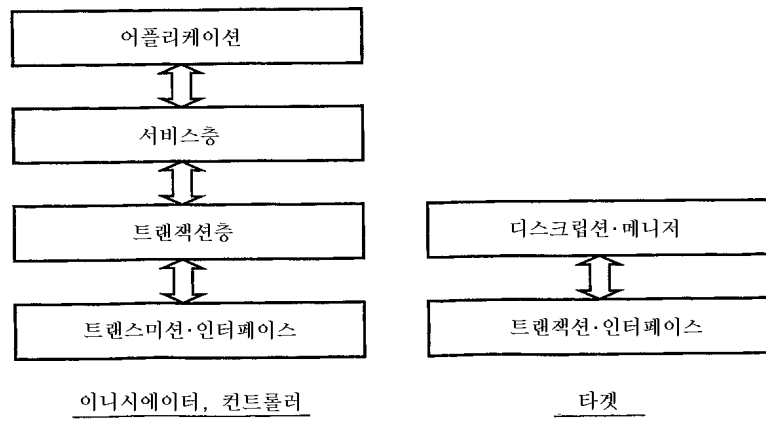
- <279> 여기서, 갱신하는 TOC가 8개 이하, 즉 1회의 트랜잭션으로 가비지 컬렉션을 완료할 수 있는 경우에는, 남은 모든 TOC의 타입(Type)을 Unused(미사용)로 재기입(단계 S12)하고, 본 처리 루틴 전체를 종료한다. 이들 TOC의 재기입 동작은 1회의 트랜잭션만으로 행한다.
- <280> 한편, 갱신하는 TOC가 9개 이상 있는 경우에는, 1~7번째까지의 TOC의 타입(Type)을 Unused(미사용)로 함과 아울러, 8번째의 TOC의 타입(Type)은 Erasing(소거가능)으로 한다(단계 S3). 이들 TOC의 재기입 동작은 1회의 트랜잭션(T11)만으로 행한다.
- <281> 이하, 현재 8번째의 TOC 이후에 대하여 처리를 행한다(단계 S14).
- <282> 즉, 단계 S11로 되돌아가, 갱신하는 TOC가 아직 9개 이상 남아 있는지, 즉 1회의 트랜잭션으로 모든 TOC의 갱신을 갱신할 수 없는지의 여부를 판단한다.
- <283> 그리고, 갱신하는 TOC가 아직 9개 이상 남아 있는 경우에는, 1~7번째까지의 TOC의 타입(Type)을 Unused(미사용)로 함과 아울러, 8번째의 TOC의 타입(Type)을 Erasing(소거가능)으로 한다(단계 S3). 이들 TOC의 재기입 동작은 1회의 트랜잭션(T12)만으로 행한다(동상).
- <284> 이상, 특정 실시 형태를 참조하면서 본 발명에 대하여 설명하였다. 그러나, 본 발명의 요지를 일탈하지 않는 범위에서 당업자가 해당 실시 형태의 수정이나 대응을 할 수 있는 것은 자명하다.
- <285> 본 명세서에서 설명한 실시 형태에서는, IC 카드 기술을 이용하여 구축된 근접 통신 시스템을 예로 하여 설명하였지만, 다른 타입의 근접 통신 시스템이더라도, 마찬가지로 본 발명이 실현 가능한 것은 물론이다. 물론, 국소적인 근접 통신이 아니고, 일반적인 디바이스간 통신, 혹은, 메모리나 그 밖의 기억 장치에의 데이터 기입에 있어서도, 데이터의 정합성을 보증하기 위해서 본 발명을 적용할 수 있다.
- <286> 결국, 본 발명은 예시라는 형태로 개시한 것으로, 본 명세서의 기재 내용을 한정적으로 해석하여서는 안된다. 본 발명의 요지를 판단하기 위해서는, 청구의 범위를 참작하여야 한다.

## 도면

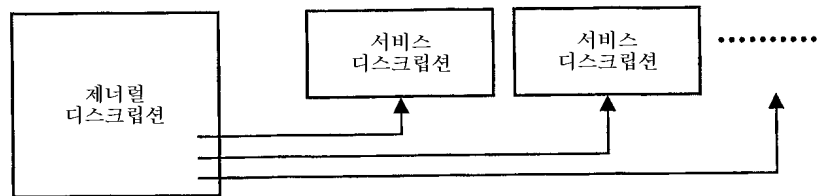
### 도면1



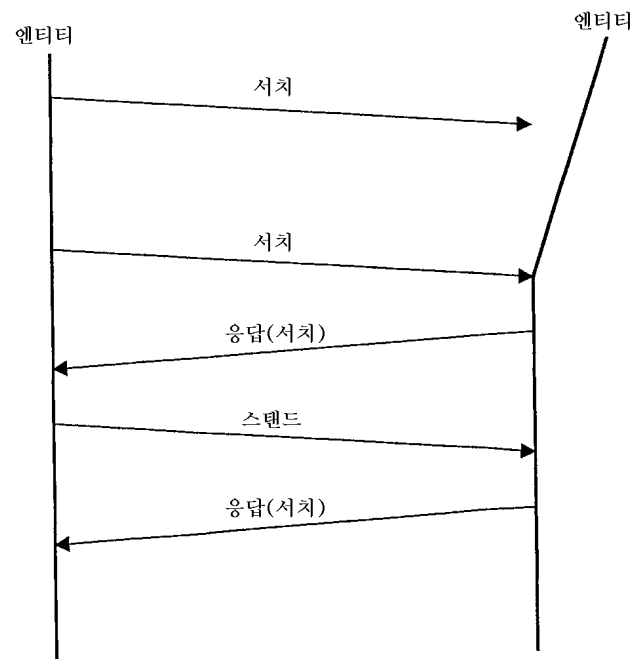
도면2



도면3

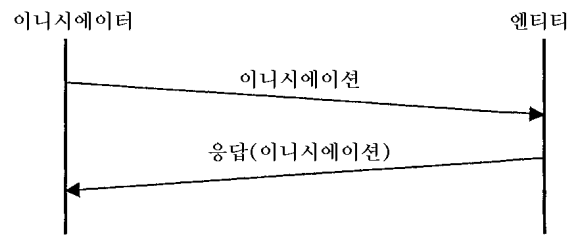


도면4

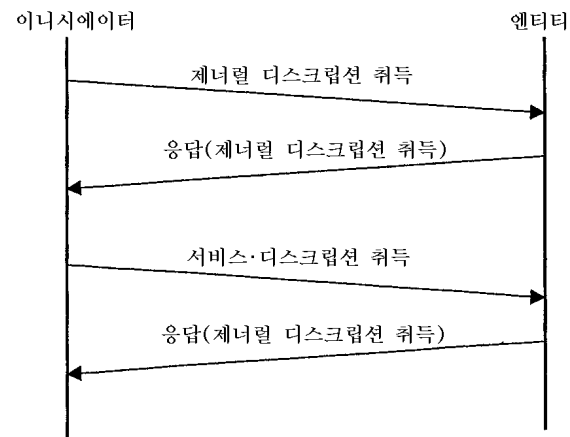




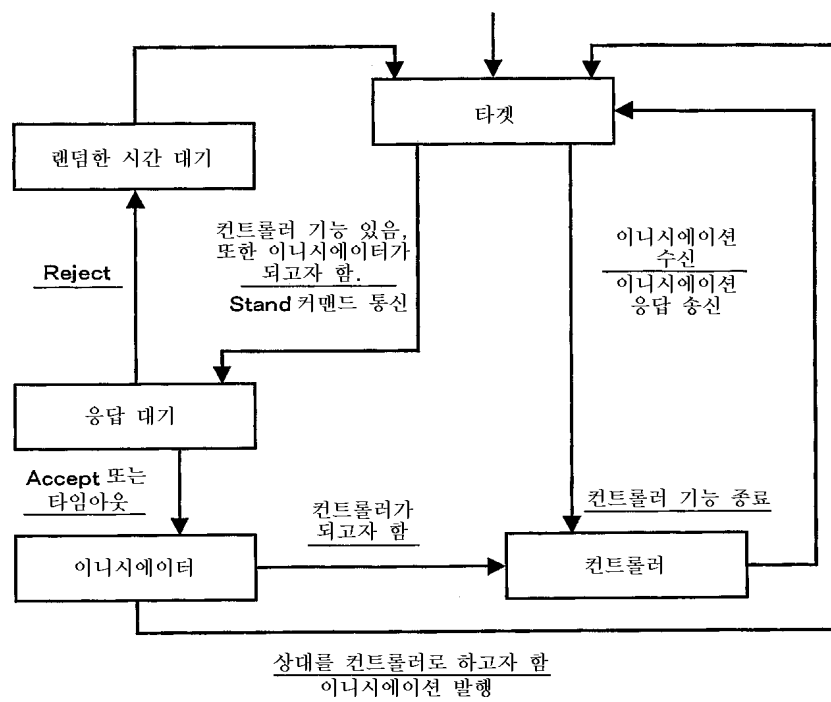
도면5



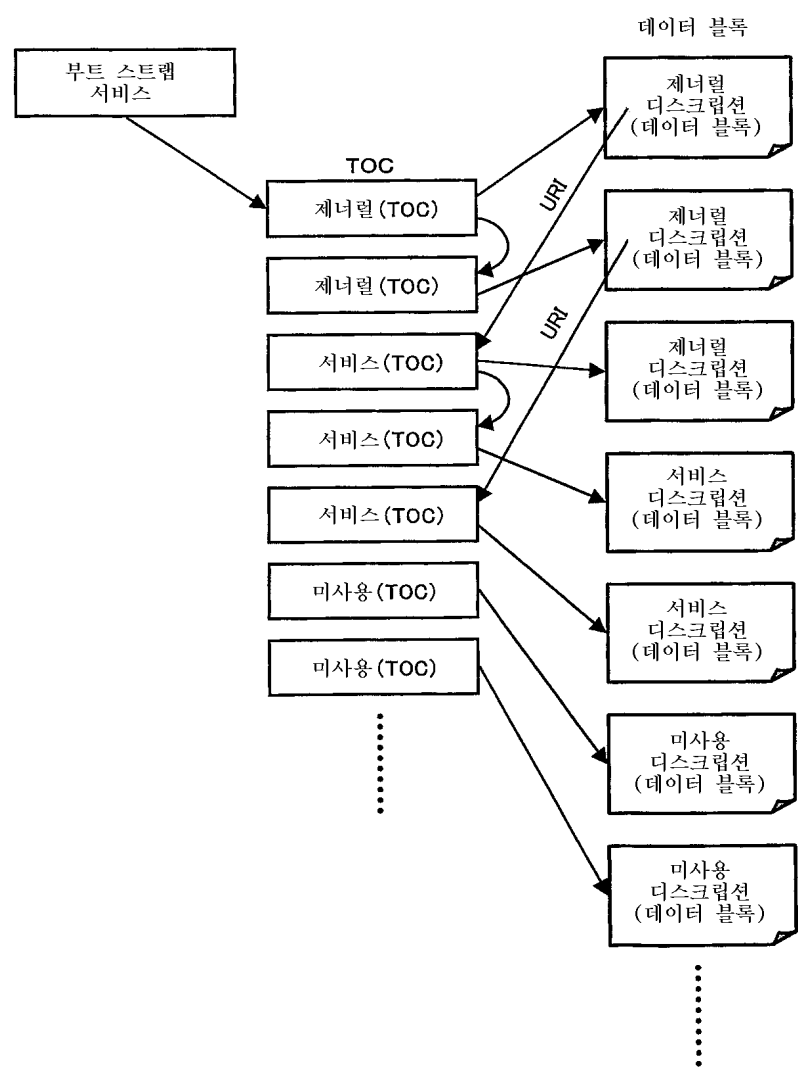
도면6



도면7



도면8



도면9

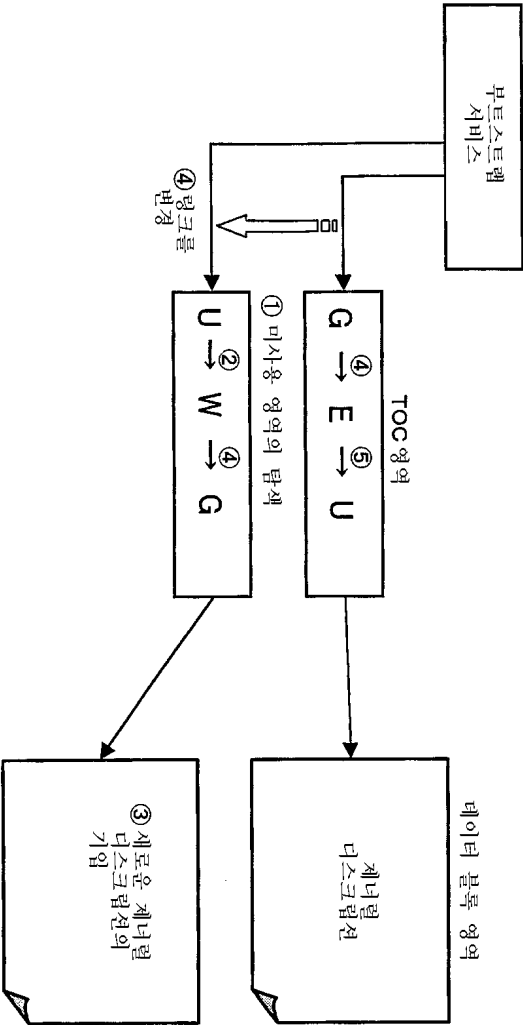
D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	Da	Db	Dc	Dd	De	Df
미디어 타입	속성	TOC 개시 어드레스	엔트리수	제너럴 디스크립션 개시 어드레스	예약										

도면10

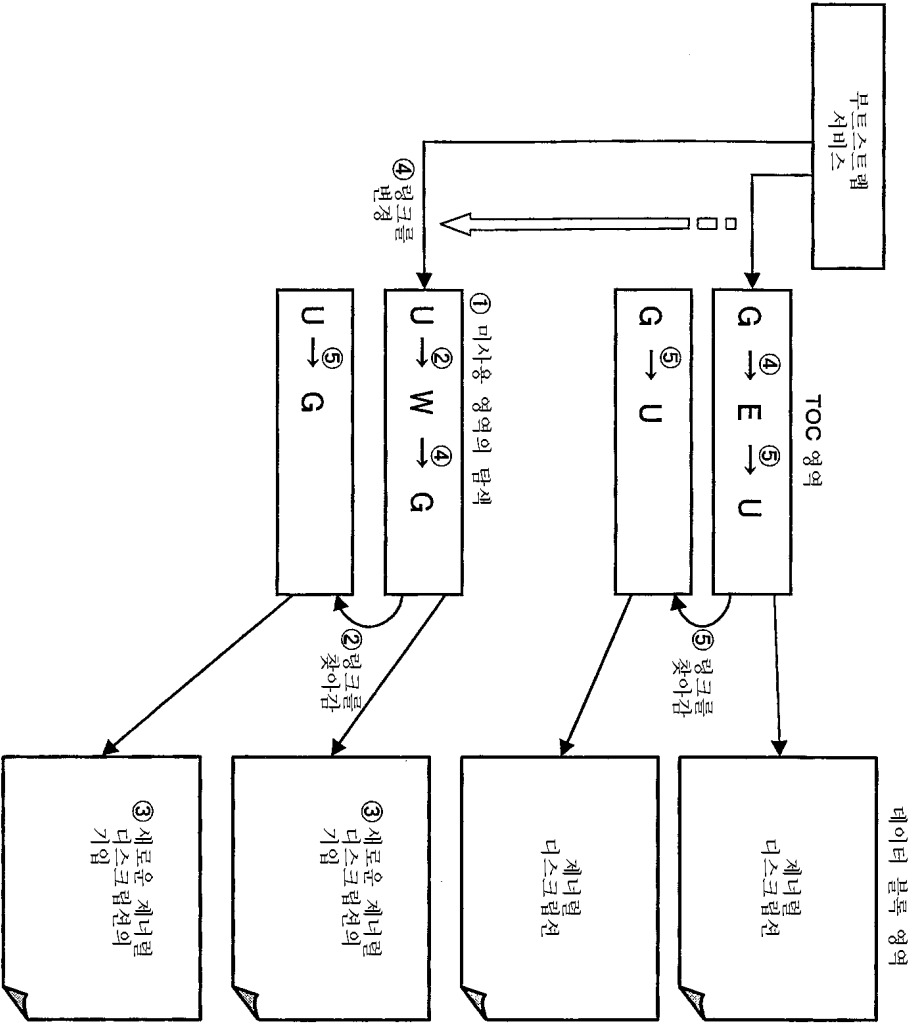
D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	Da	Db	Dc	Dd	De	Df
타입	속성	데이터 블록 개시 어드레스	블록수	다음 TOC 어드레스	데이터 사이즈	예약									



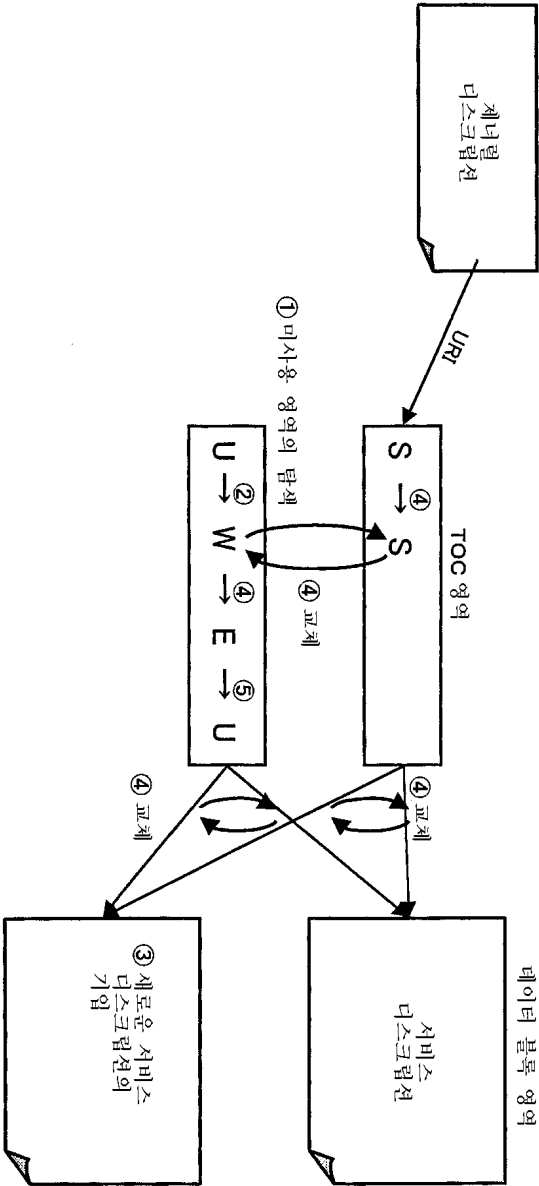
도면11



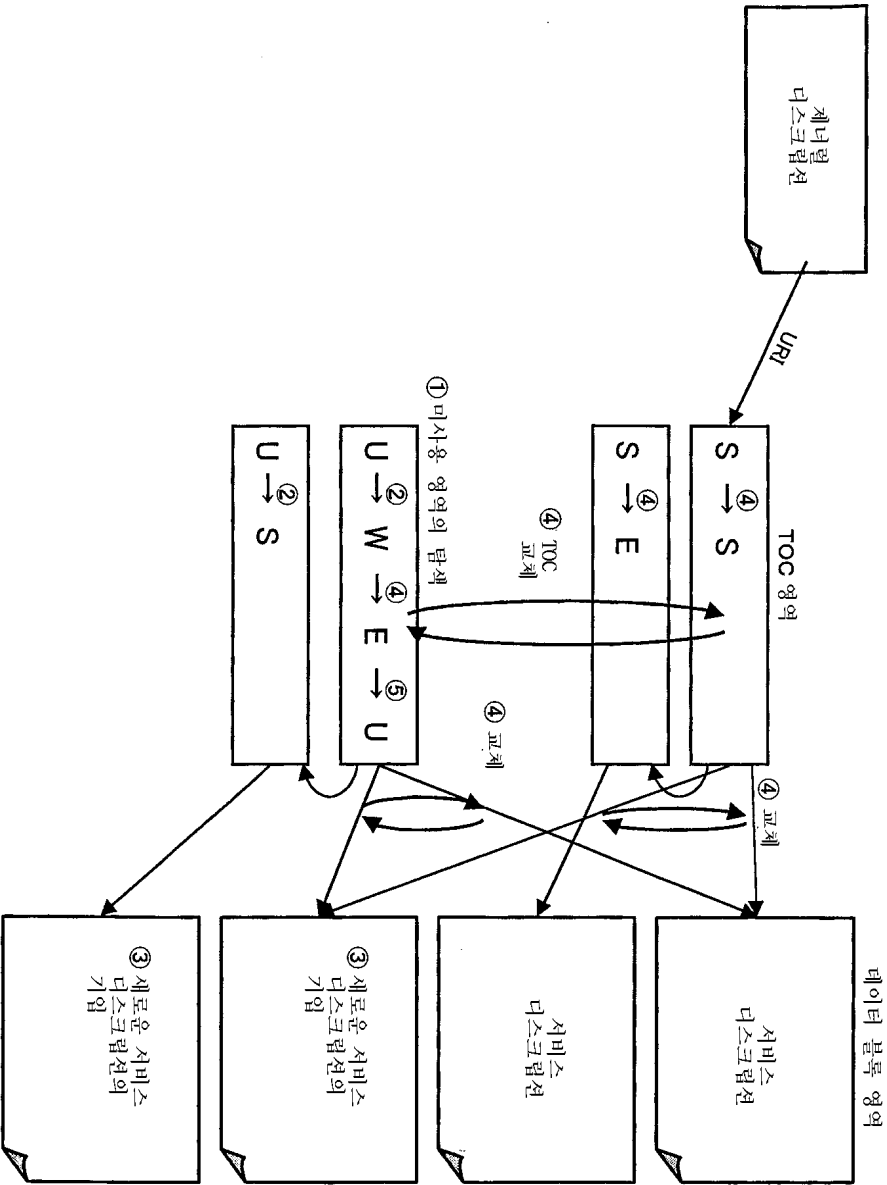
도면12



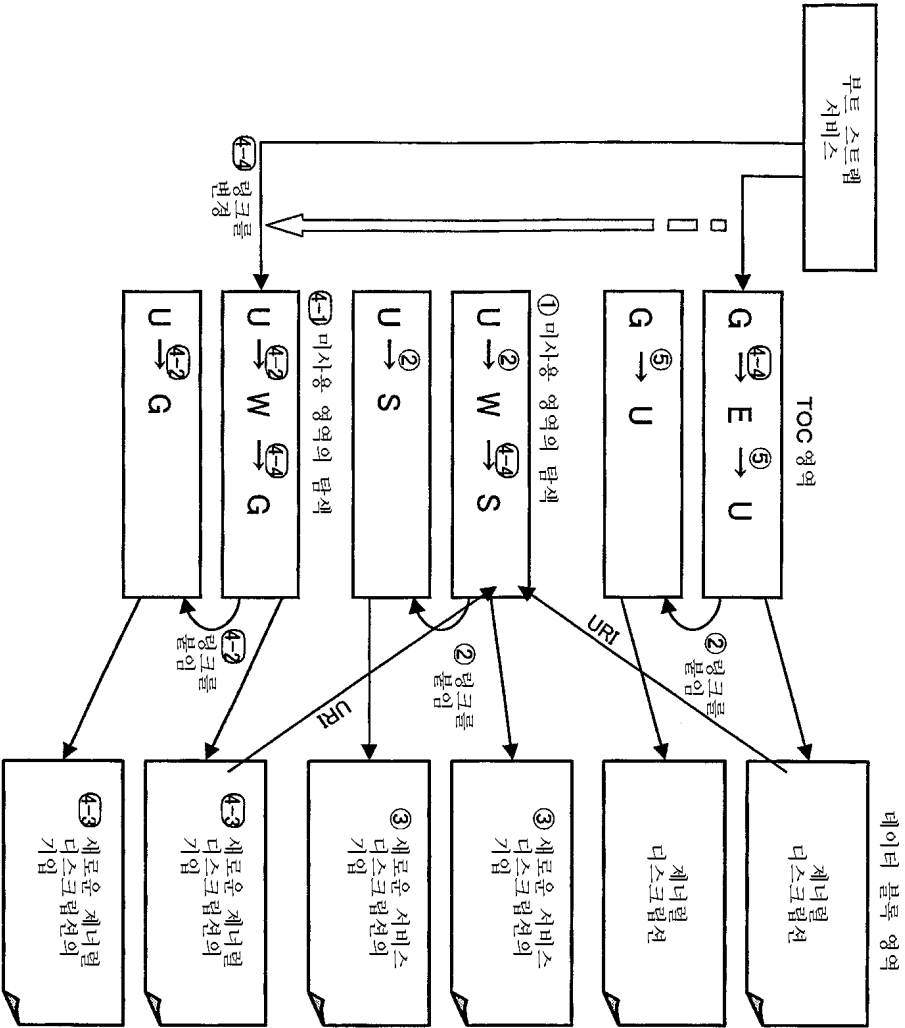
도면13



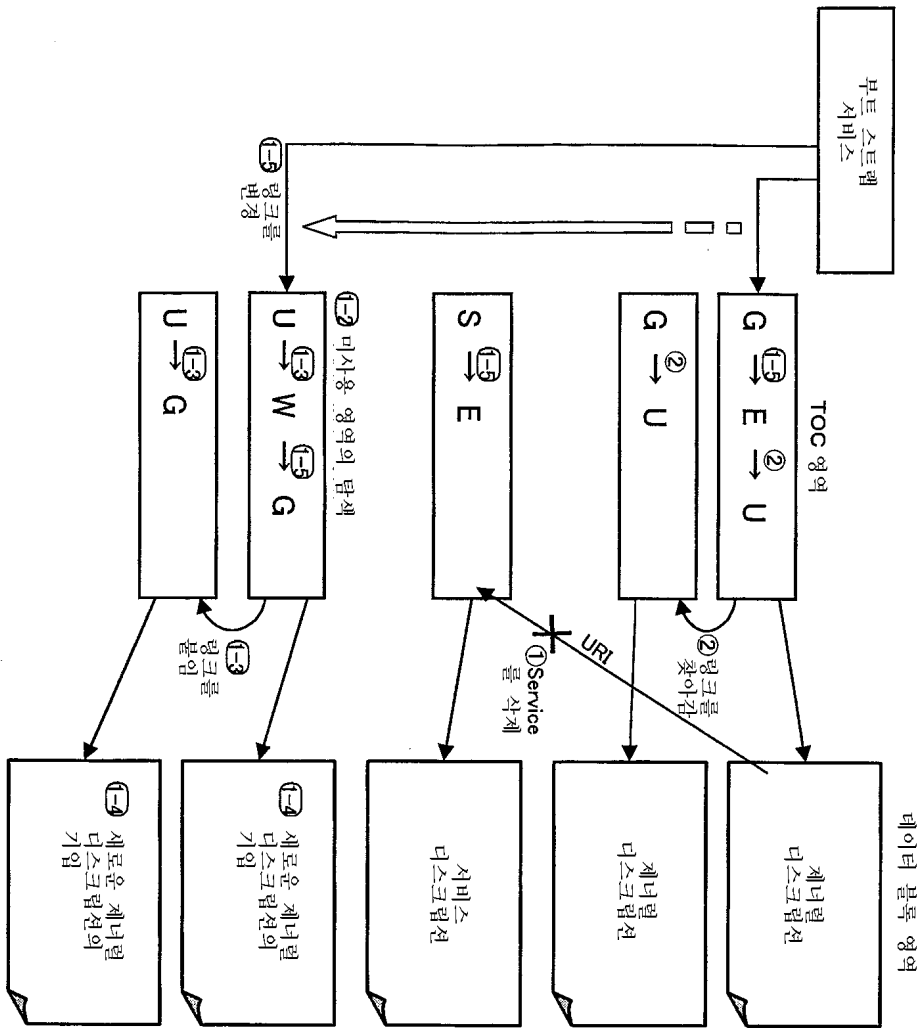
도면14



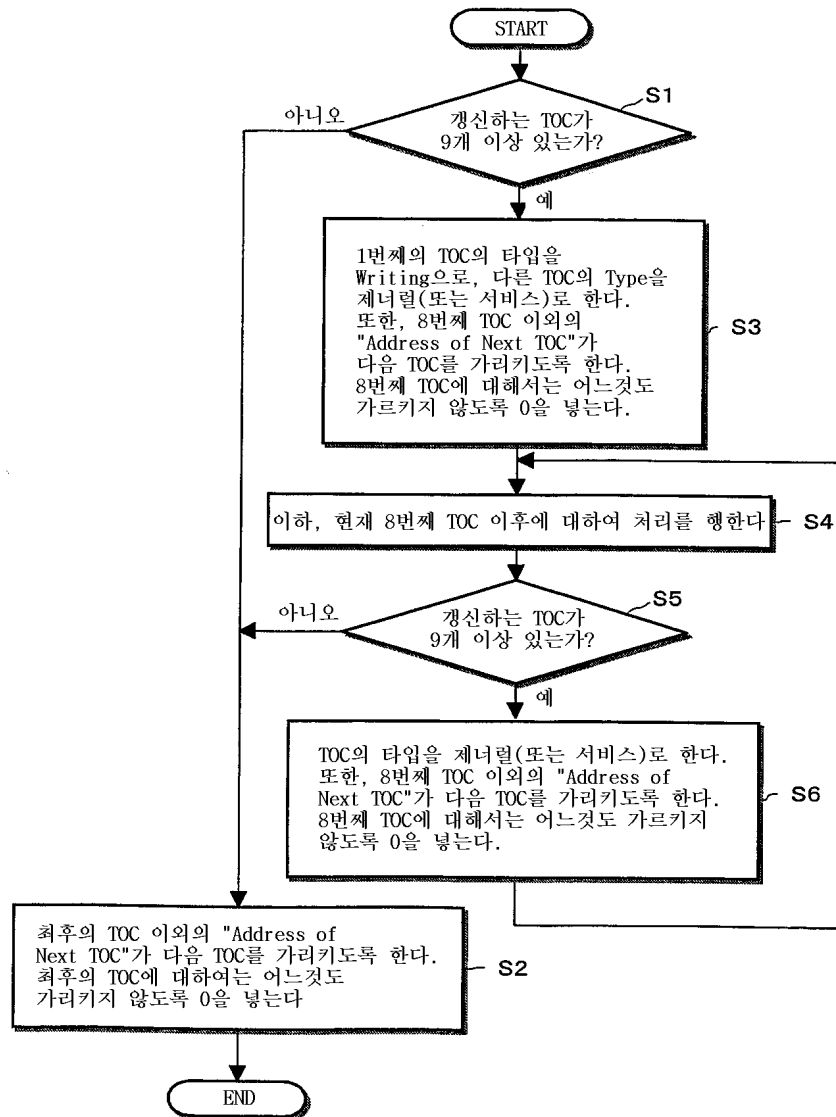
도면15



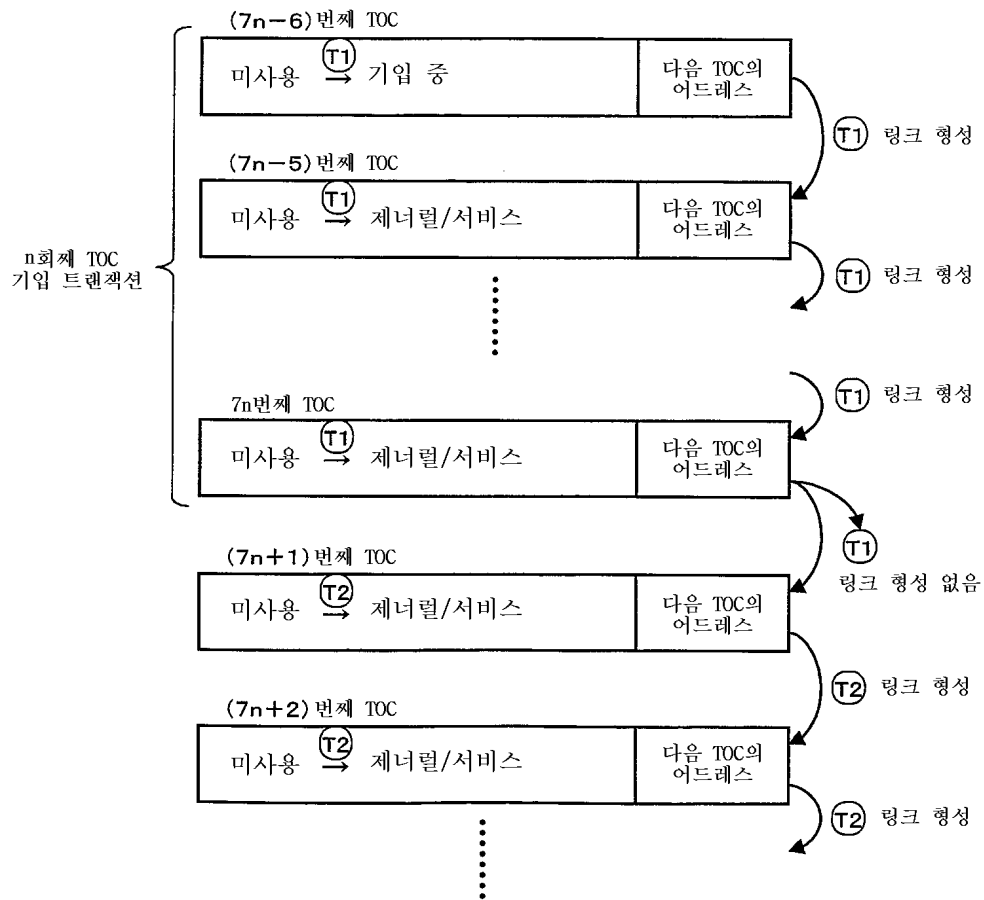
도면16



도면17



도면18



도면19

