| | |
|---|---|
| Project | **IEEE 802.16 Broadband Wireless Access Working Group <http://ieee802.org/16>** |
| Title | **Efficient Systematic CID Allocation and Relay Path Configuration Mechanism for IEEE 802.16j (Multi-hop Relay)** |
| Date Submitted | **2007-1-18** |


Source(s)

Aik Chindapol
Jimmy Chui
Hui Zeng
Siemens Corporate Research
Princeton, NJ, 08540, USA
Voice: +1 609 734 3364
Fax: +1 609 734 6565
Email: aik.chindapol@siemens.com

Teck Hu
Siemens Networks
Boca Raton, FL 33431, USA

Yuan-Ying Hsu
Telcordia Applied Research Center Taiwan Co.,
Taipei, Taiwan
Voice: +886-2-37895177#4558
Fax: +886-2-26552078
yyhsu@tarc-tw.research.telcordia.com

Jen-Shun Yang, Tzu-Ming Lin, Wern-Ho Sheen, Fang-Ching Ren, Chie Ming Chou, I-Kang Fu
Industrial Technology Research Institute (ITRI)/
National Chiao Tung University (NCTU), Taiwan 195,Sec. 4, Chung Hsing Rd.
Chutung, Hsinchu, Taiwan 310, R.O.C.
Voice: +886-3-5914616
Fax: +886-3-5820263
jsyang@itri.org.tw

Byung-Jae Kwak, Sungcheol Chang, Young-il Kim
ETRI
161, Gajeong-Dong, Yuseong-Gu, Daejeon, Korea 305-350
Voice: +82-42-860-6618
Fax: +82-42-861-1966
bjkwak@etri.re.kr

Changkyoon Kim, Kyu Ha Lee, Hyung Kee Kim
Samsung Thales Co., Ltd
San 12-1, Nongseo-Dong, Giheung-Gu, Yongin-City, Gyeonggi-Do, Korea 446-712
Voice: +82-31-280-9919
Fax: +82-31-280-1620
changkyoon.kim@samsung.com

Kenji Saito, Takashi Inoue
KDDI R&D Laboratories Inc.
Hikarino-oka 7-1, Yokosuka,
Kanagawa 239-0847, Japan

Voice: +81 46 847 6347
Fax: +81 46 847 0947
saito@kddilabs.jp

Sungjin Lee,
Samsung Electronics

Voice: +82 31 279 5248
Fax: +82 31 279 5130
steve.lee@samsung.com

| | |
|---|---|
| Re: | IEEE 802.16j-06/034:"Call for Technical Proposals regarding IEEE Project P802.16j" |
| Abstract | This document describes the route update procedures with efficient CID management. |
| Purpose | This contribution is provided as input for the IEEE 802.16j baseline document. |
| Notice | This document has been prepared to assist IEEE 802.16. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein. |
| Release | The contributor grants a free, irrevocable license to the IEEE to incorporate material contained in this contribution, and any modifications thereof, in the creation of an IEEE Standards publication; to copyright in the IEEE's name any IEEE Standards publication even though it may include portions of this contribution; and at the IEEE's sole discretion to permit others to reproduce in whole or in part the resulting IEEE Standards publication. The contributor also acknowledges and accepts that this contribution may be made public by IEEE 802.16. |
| Patent Policy and Procedures | The contributor is familiar with the IEEE 802.16 Patent Policy and Procedures <http://ieee802.org/16/ipr/patents/policy.html>, including the statement "IEEE standards may include the known use of patent(s), including patent applications, provided the IEEE receives assurance from the patent holder or applicant with respect to patents essential for compliance with both mandatory and optional portions of the standard." Early disclosure to the Working Group of patent information that might be relevant to the standard is essential to reduce the possibility for delays in the development process and increase the likelihood that the draft publication will be approved for publication. Please notify the Chair <mailto:chair@wirelessman.org> as early as possible, in written or electronic form, if patented technology (or technology under patent application) might be incorporated into a draft standard being developed within the IEEE 802.16 Working Group. The Chair will disclose this notification via the IEEE 802.16 web site <http://ieee802.org/16/ipr/patents/notices>. |

# Efficient Systematic CID Allocation and Relay Path Configuration Mechanism for IEEE 802.16j (Multi-hop Relay)

## 1 Introduction

In 802.16e, each connection (both management and data) is identified by a Connection ID (CID) [2]. There is no routing required; data is transmitted solely between the BS and the MS. In a centralized multi-hop relay system, all relay stations form a tree topology with MR-BS as the root. The routing for each MS is decided by the MR-BS. The routing path could be mainly based on the network topology and other number of factors such as measured channel qualities, QoS of each connection, fairness, etc.

This contribution proposes to use CID assignment in a multi-hop relay system as a means to indicate network topology and further help the MR-BS to perform routing path selection. In this scheme, each relay station is assigned a range of CIDs for which the relay is responsible for decoding and forwarding. The parent node will control a superset of this CID range, and any child nodes (both RS and MS) will be assigned disjoint subsets of the CID range. Because of the structure of this CID assignment, the MR-BS could compute the relay path simply based on CIDs of destination station and each relay station can recognize its packets and forward them to corresponding stations. In this way, the routing can be maintained automatically along with CID assignment and thus significantly decrease complexity, signal overhead, and path setup latency. The systematic CID allocation could also help to reduce the delay when handoff of MRS or MS occurs, which could be achieved by locating anchor RS and updating CID of moving station to maintain the topology quickly.

The proposal for this implementation has the following advantages:

- Simplified operation of the relay
- Reduction of overhead and delay in route updates
- Reuse of existing signaling to reflect topology changes due to the movement of MRSs or MSs
- Quickly locating the anchor RS for fast handoff while MRSs or MSs move

## 2 General Description

A unidirectional connection between BS and MS or between BS and RS is established for service flow traffic, and each connection is identified by a connection identifier (CID) [2][3]. The CID for each connection is inserted within the MAC header of a packet. When it is received, first each station checks whether the CID of the packet is for itself or for its subordinate stations. Each station accepts the packet and does the process if the packet is intended to itself or its subordinate station. Otherwise, it ignores the packet and does nothing.

Each station can distinguish the received packets by examining the CID in the MAC header or in the MAP-IE, and this can be used to maintain the routing structure. By combining CIDs with the routing for each connection, the routing structure can be updated and maintained easily along with CIDs, and the overall overheads for the routing can be reduced.

## 3 Partitioning of Integers

### 3.1 General Idea

To systematically assign CIDs to the MR-BS and RSs, the proposed CID allocation mechanism adopts the partitioning of the positive integers into subsets. In this section, we describe two methods of the partitioning of integers. This can be accomplished by factorization into bit partition, prime numbers [6], or into contiguous blocks [7]. The idea is to map these subsets to nodes in a network (assumed to be a tree topology) which will assist in identifying the placement of the node in the tree.

Each node of the tree represents a subset of **Z**, the set of all positive integers. The leaves of the tree are pairwise disjoint subsets of the integers. Each parent node is a superset of the union of its children. For example, in Figure 1, $B \supset (D \cup E)$ and $D \supset (H \cup J \cup K)$. The tree can grow; at a particular node, its children must satisfy two conditions. 1) the children must be subsets of the parent node; 2) the children are pairwise disjoint.

Due to this structure, any node (root, leaf, or intermediary) can determine whether a particular integer will exist in its subtree (with itself as the root). Intermediary nodes must distinguish between two types of integers; those that terminate at the node (terminal integers), and those that do not terminate at the node (non-terminal integers). We provide two examples of integer partitioning that assume only one terminal integer at each intermediary node, and briefly mention how multiple terminal integers (per intermediary node) can be attained.

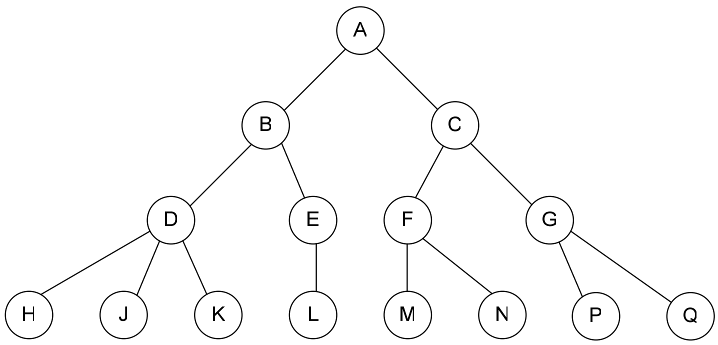


**Figure 1: an example of a network tree (an abstract model)**

### 3.2 Examples of integer partitioning: contiguous integer blocks

This is a simple implementation. The root node represents **Z**. Each of its children (1st tier nodes) are assigned a contiguous range of **Z** (and pairwise disjoint). For a particular 1st tier node (with range [$p_1$, $p_2$]), its children (2nd tier nodes) are each assigned a contiguous subset of [$p_1$, $p_2$] (and pairwise disjoint). This process continues for the entire tree. In Figure 2a, we demonstrate how the tree in Figure 1 can partition the integers using contiguous integer block methods.

In Figure 2a, the terminal integers for nodes B, C, D can be set to 1000, 2000, and 400 respectively. Allowing multiple terminal integers per intermediary node is trivial.

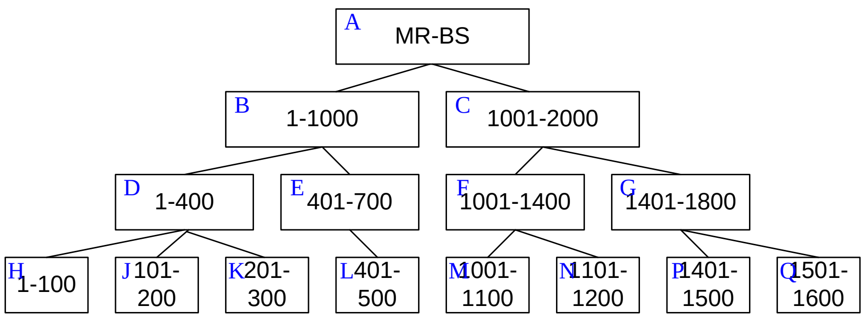


**Figure 2a: partitioning of integers using contiguous blocks. The choice of range length being multiples of 100 is arbitrary.**

### 3.3 Examples of integer partitioning: bit partition

Each decimal number could also be converted into a binary number. Assume there are at most $2^k$ RSs could associate with one RS or BS directly, $k$ bits would be used to identify each RS in the same level. The 1st tier nodes that associate to the MR-BS directly would have CIDs with all possible number in lowest $k$ bits. Their children (2nd tier nodes) is identified by left shifting $k$ bits of parent CID and set lowest $k$ bits. This process continues for the entire tree. In this manner, the CID (without leading 0s) of any RS will be the prefix of CIDs of all its subordinate RSs.

To convert these values into subsets of **Z** (as discussed in Section 3.1) is simple; a nth tier node will have a unique *nk*-bit sequence to identify itself, then the range this node could assign will be all numbers with this nk-bit sequence in the middle and begin with arbitrary number of “0”s as its prefix and with arbitrary combination of 0 and 1 as its suffix. The condition as set out in Section 3.1 is satisfied. We also demonstrate how the tree in Figure 1 can partition the integers using bit partition method in Figure 2b.

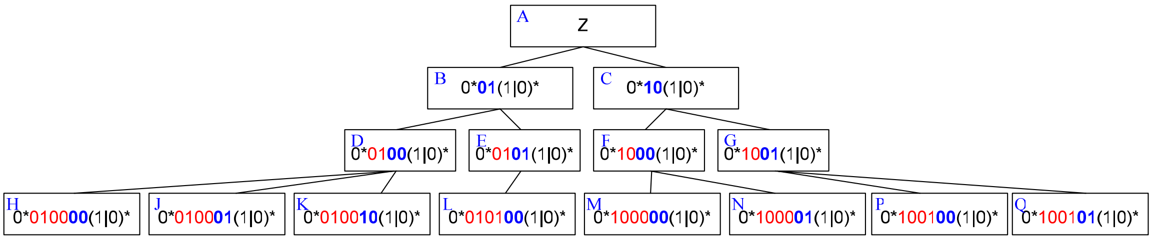


**Figure 2b: partitioning of integers using bit partition.**

Note that a simple way of allowing multiple terminal values at intermediary nodes is to merge nodes. For example, the logical nodes H and J can represent the same physical node.

# 4 CID Assignment

We describe the CID allocation for both contiguous integer partitioning method and bit partition method in the following. This structured CID assignment can be used to assign tunnel CIDs (i.e. CIDs terminating at an access RS), or RS and MS management CIDs. It can also be used to assign MS transport CIDs. Our examples describe the assignment of tunnel CIDs.

4.1 CID allocation using contiguous integer partitioning method

We perform this CID assignment scheme *ignoring the MS in the topology*. This method is compatible with the notion of encapsulating CIDs [9] or tunneling CIDs [8] or embedded MAP IE. [8].

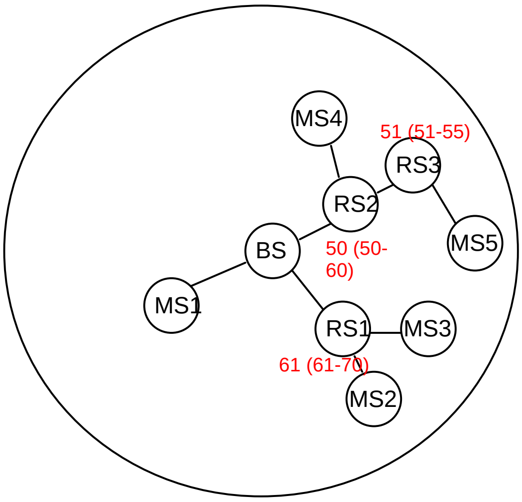


Figure 3: CID Assignment for 4.1

In Figure 3, the MR-BS indicates the range of CIDs that each RS is responsible for, as well as the CID for the connection for each RS node. For each subordinate RS, the assigned CID range has to be a subset of its connecting RS.

4.2 CID allocation using bit partition method

We first define a parameter $2^k$ to identify the maximum number of subordinate RSs that the MR-BS or a RS could have. If k=0, each RS could only have one subordinate RS. For $1^{st}$-tier RSs, which connect to the MR-BS directly, the MR-BS assigns IDs sequentially from 1 to $2^k$ as shown in Figure 4 by setting different values of the lowest *k* bits of the ID. We only show the lowest 8 bits of CIDs in Figure 4. For other n-tier RSs, the MR-BS left shifts *k* bits of its parent ID and sets the lowest *k* bits according to the arriving sequence of the RS. For example, $RS_T$ and $RS_U$ comes one after another to associate with $RS_Q$ (ID: 00 01 00 11) after $RS_R$ and $RS_S$ in Figure 3. To assign an ID to $RS_T$, the MR-BS first perform left shift 2 bits of its parent ID and gets 01 00 11 00, and then it sets the lowest 2 bits as 10 since it is the third RS that attaches to $RS_Q$. Similarly, the MR-BS assigns 01 00 11 11 to $RS_U$ after $RS_T$.

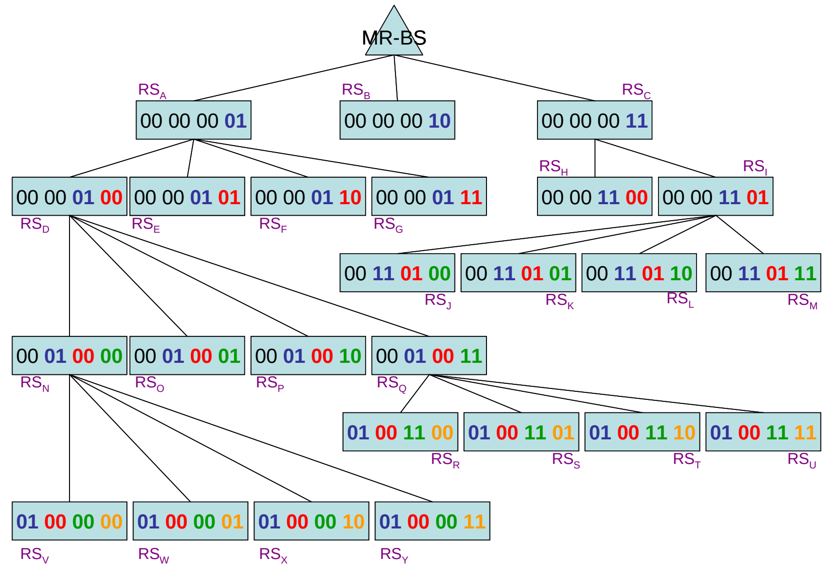


**Figure 4: CID assignment for 4.2.**

# 5 Relay Path Configuration

To utilize the proposed systematic CID allocation in relay path configuration, it is possible to use tunneling [8] [10] or non-tunneling solution [9]. In the following, we take bit partition method to describe tunneling relay path configuration and take integer contiguous method as an example for non-tunneling relay path configuration. *Both methods could be applied two solutions of relay path configuration.*

### 5.1 Examples of relay path configuration using tunneling solution

We take figure 5 for example. There are two MSs, which associate to RS L (CID: 00 01 01 00) and RS G (CID: 00 00 10 01), in the network. The MR-BS has records for these two MSs and knows their serving RSs. The whole relay path could be divided into two segments: from the source RS to the MR-BS and from the MR-BS to the destination RS. For upstream frames, each RS could easily know its parent CID by right shifting *k* bits of its own CID. For example, the CID of access RS L is 00 01 01 00, so its parent CID is 00 00 01 01 by right shifting 2 bits of its CID. For downstream frame received from its parent RS, the RS needs to determine if it should accept, forward, or discard the frame. When the tunneling [8] [10] is applied for relaying, the Tunnel CID could be set as the CID of destination RS. Each intermediate RS could compute if the destination RS belongs to its subordinate RSs by the algorithm in Figure 6. First of all, the RS compares if the destination CID is equal to its own CID and accepts the frame if these two CIDs are the same. If the match fails, it perform *k*-bit right shift of the destination CID and do the comparison with its own CID. If the shifted destination CID is the same as its own CID, it forwards the frame to its subordinate RS. Otherwise, it continues do the right shift and comparison for (maximal level-current level) times and discards the frame if all matches are failed. For example, RS C would know that RS G is its subordinate RS by right shifting the destination CID once.

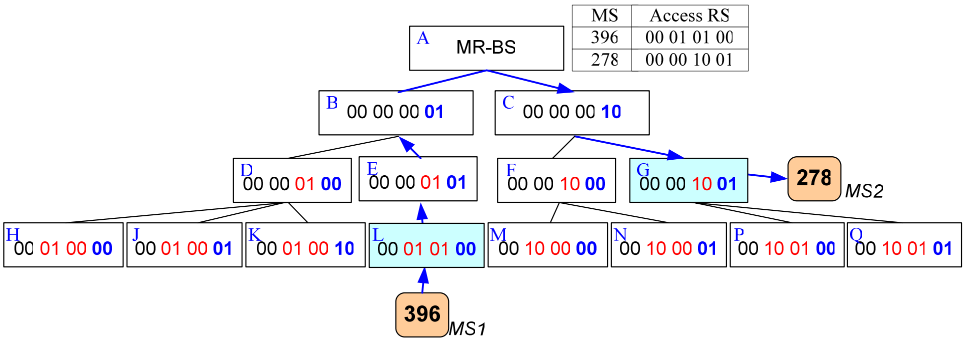


**Figure 5: An example of relay path configuration using bit partition method.**

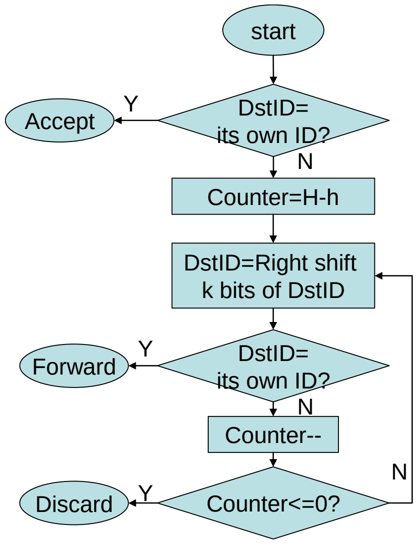


**Figure 6: Subordinate RS differentiation algorithm**

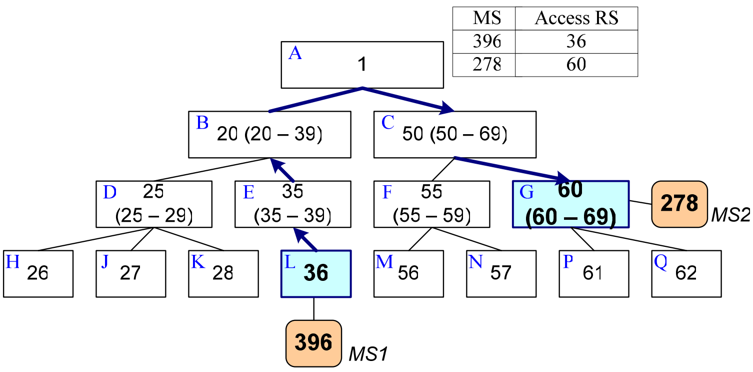


**Figure 7: An example of relay path configuration using contiguous integer partitioning method. The number in parenthesis is the range of CIDs that the MR-BS could allocate to the subordinate RS.**

Similarly with contiguous integer partitioning, the MR-BS keeps records of the access RS for each MS. For data directed towards MS2, the MR-BS sends the data to the access RS with CID 60. Since this CID belongs to the range of CID of the RS C, it forwards the data to the RS G. Meanwhile, the RS B ignores this data as the CID is not within its range. The similar procedure can be done on the uplink.

## 5.2 Examples of relay path configuration using non-tunneling solution

The following example, as shown in Figure 7, describes how the relay operation is performed if the same CID is maintained for every hop involved in the transmission. Relay involving the changing of CIDs, or relay involving encapsulation of CIDs [1] can be done in a similar manner.

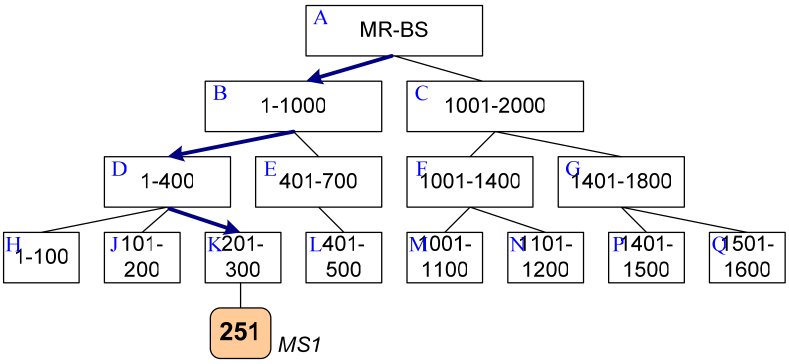


**Figure 7: An example of relay path configuration using contiguous integer method.**

Consider the transmission of a packet with CID 251. RS B decodes and forwards the message to RS D, and RS C ignores this transmission because the CID does not belong to its CID range 1001-2000. Then, RS D forwards the message to RS K in the same manner. RS K is able to decode and forward this packet because CID 251 is within its forwarding range. Finally, the packet with CID 251 is received by MS 1. The uplink transmission from each MS can be delivered to BS through RS in a similar way.

Once the routing decision is made by the BS, CIDs of MS are assigned according to the procedures described above. Each RS only listens to the packets with CIDs which are within its CID range and ignores otherwise. This removes the necessity to maintain and broadcast routing information.

In addition, the handover of moving MS from one RS to another RS within a cell can be handled easily. For example, if MS 1 moves close to RS L and BS decides to serve MS 1 through RS L, then BS changes MS 1's CID from 251 to 405 directly. Then RS L knows automatically that it should detect and decode the packets for MS 1. This handover within a cell is done transparently to MS. In other words, MS only knows that BS changes its CID and does not know that the routing path has been changed. The BS can request the change of CID using BS-initiated DSC procedures as specified in subclause 6.3.14.9.4.2 of [2], which is summarized in Figure 8. A DSC-REQ is sent by BS to dynamically change the service flow (SF) parameters, including CID, of an existing SF according to the specified SFID. Once the SS receives the DSC-REQ and validates the request, the SF parameters will be changed and DSC-RSP will be sent by SS. After receiving the DSC-RSP, the BS will update the service flow profile accordingly, and send DSC-ACK to confirm. The whole procedure is completed once the SS receives DSC-ACK and finalizes the adjustment.

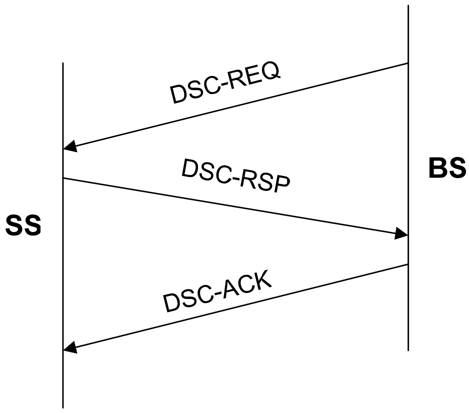

**Figure 8: BS-initiated DSC.**

CID ranges that have to be used by RS are assigned by BS and transmitted to RSs via CIDRNG-REQ and CIDRNG-RSP management messages as specified in Section 6.

# 6 Proposed Text

---------------------------Beginning of Text Changes----------------------------------------------------------------------

*[Add the following text into section 6.3.1.3]*

***6.3.1.3.1 Addressing Scheme for Relaying***

In the procedure of network entry and initialization for a new RS, the MR-BS shall assign a range of CIDs for the new RS. The range could be contiguous integer blocks as in Figure 6.3.1.3.X (a) or bit partition as in Figure 6.3.1.3.X (b). In the bit partition assignment, the MR-BS sets the lowest *k* bits in ascending order to RSs for RSs associated to the MR-BS directly where the maximum number of RSs the MR-BS or a RS could serve is $2^k$. For other level-n RSs, which need *n* hops to reach the MR-BS, the MR-BS left shifts *k* bits of its parent CID and sets the lowest *k* bits according to the arriving sequence of the RS.

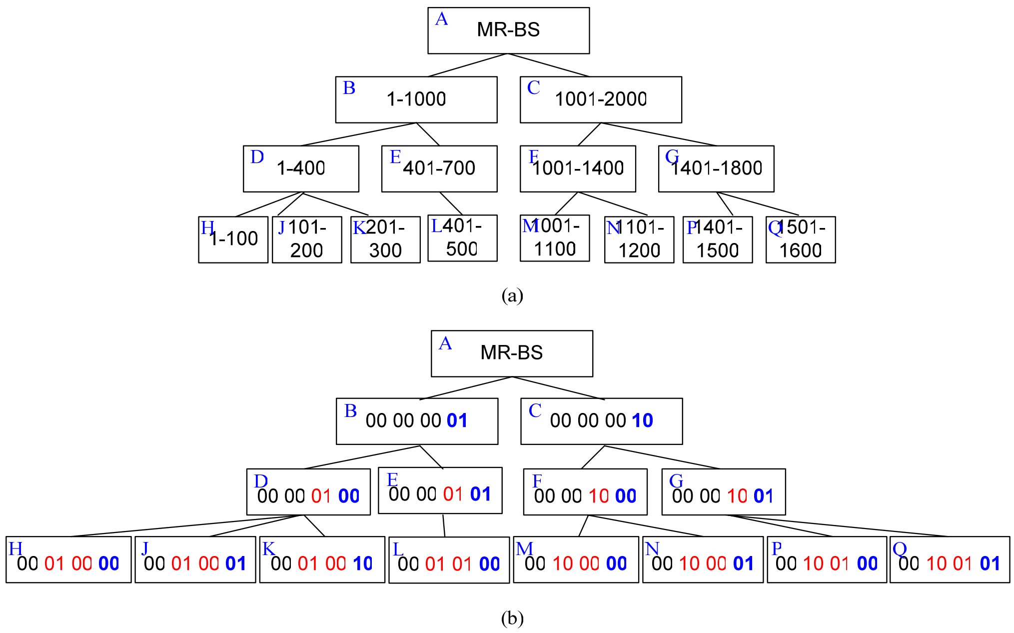


**Figure 6.3.1.3.Y: CID range allocation example, (a) contiguous integer block, (b) bit partition method.**

6.3.2.3.6 Ranging response (RNG-RSP) message

Insert the following text at the end of the 6.3.2.3.6:

The following TLV parameter shall be included in the RNG-RSP message when transmitted during RS initial entry to the network. The MR-BS could assign the range of RSs.

Range of CID for RS

11.6 RNG-RSP management message encodings

Insert the following entries into Table 367:

Table 367 – RNG-RSP message encodings

| Name | Type (1 byte) | Length | Value (variable-length) | PHY Scope |
|---|---|---|---|---|
| If(contiguous integer blocks){ | | | | |
| Start number of management CID for RS | xx | 2 | | OFDMA |
| End number of management CID for RS | xx | 2 | | OFDMA |
| } | | | | |
| If(bit partition){ | | | | |
| CID bit sequence | xx | 2 | | OFDMA |
| } | | | | |

6.3.25 Relay path management and routing

Each relay station is assigned a range of CIDs for which the relay is responsible for decoding and forwarding. The CID range is assigned by the MR-BS, and are transmitted to RSs via CIDRNG-REQ and CIDRNG-RSP management messages. During operation, the RS is only responsible for listening to CIDs transmitted within this range.

The BS is responsible for managing the entire CID range. Each RS connected to a parent node (BS or RS) is assigned a subset of the CIDs assigned to the parent node. These subsets are non-overlapping.

By assigning a systematic CID to each RS, the MR-BS already specifies the relay routing path of the connection.

# 7 References

[1] IEEE C802.16j-06/004r1, “Recommendations on IEEE 802.16j”.

[2] IEEE 802.16-2004, “Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems”.

[3] IEEE 802.16e-2005, “Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems, Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands *and* Corrigendum 1”.

[4] IEEE C802.16j-06/014r1, “Harmonized definitions and terminology for 802.16j Mobile Multihop Relay"

[5] IEEE C802.16j-06/015, “Harmonized Contribution on 802.16j (Mobile Multihop Relay) Usage Models"

[6] IEEE C802.16j-06/171, “Systematic relay station identification allocation and relay path configuration mechanism for IEEE 802.16j (Multi-hop Relay)”.

[7] IEEE C802.16j-06/253, “Route Update with Efficient CID Management”.

[8] IEEE C802.16j-06/274, “Proposal on addresses, identifiers and types of connections for 802.16j”.

[9] IEEE C802.16j-06/254, “Fast Connection Establishment and Maintenance with Relays”.

[10] IEEE C802.16j-06/170, “Connection Identification and Transmission for Relay Support”