

# Universal Powerline Bus



## The UPB System Description

Version 1.1

09/19/03

# Table Of Contents

1. Overview.....	1
1.1. Scope .....	1
1.2. Prerequisites.....	1
2. The UPB Physical Communication Method.....	2
2.1. The UPB Pulse .....	2
2.2. UPB Pulse Generation.....	2
2.3. UPB Pulse Data Encoding.....	2
2.4. UPB Collision Detection.....	4
2.5. UPB Collision Avoidance .....	4
2.6. UPB on a Split-Phase Powerline .....	4
2.7. UPB on a 3-Phase Powerline .....	4
3. UPB Data Communications .....	5
3.1. The UPB Byte .....	5
3.2. The UPB Communication Packet .....	5
3.3. UPB Packet Synchronization.....	6
3.4. The UPB Packet Header (HDR) .....	6
3.5. The UPB Message (MSG).....	7
3.6. The Packet Checksum (CHK).....	8
3.7. The Acknowledgement Frame.....	8
3.8. Destination ID Processing .....	8
3.9. UPB Communication Packet Priorities .....	11
4. The Packet Control Word .....	12
4.1. The Packet Length Field.....	13
4.2. The Link Bit.....	13
4.3. The Repeater Request Field.....	13
4.4. The Acknowledge Request Field.....	13
4.5. The Transmission Count Field.....	13
4.6. The Transmission Sequence Field .....	14
5. UPB Packet Acknowledgements .....	15
5.1. The ACK Pulse .....	15
5.2. The ID Pulse.....	15
5.3. The Acknowledgement Message.....	16
6. Enumeration Methods.....	17
6.1. The ACK-Bit.....	17
6.2. The ACK Pulse .....	17
6.3. The ID-Bit .....	17
6.4. The ID Pulse.....	17
6.5. UPB Network Enumeration.....	17
6.6. UPB Device Enumeration.....	18

7.	UPB Packet Repeating .....	19
7.1.	Repeater Packets .....	20
7.2.	Repeater Devices .....	20
8.	The UPB System Model .....	21
8.1.	UPB Setup Registers.....	22
8.2.	The UPBID .....	22
8.3.	The Configuration Registers.....	27
8.4.	Reading/Writing Setup Registers.....	31
8.5.	Modes Of Operation .....	32
8.6.	Device Command and Status Registers.....	33
9.	The UPB Message.....	34
9.1.	The Message Data ID (MDID) Field .....	34
9.2.	Message Data Arguments (MDA) Field .....	37
10.	The UPB Core Command Set.....	38
10.1.	UPB Core Commands.....	40
11.	The Device Control Command Set.....	50
11.1.	Device Control Commands .....	51
12.	The Reserved Command Set #1 .....	56
13.	The Reserved Command Set #2.....	56
14.	The UPB Core Report Set.....	56
14.1.	UPB Core Reports .....	57
15.	The Reserved Report Set #1.....	60
16.	The Reserved Report Set #2.....	60
17.	The Extended Message Set.....	60
17.1.	Extended Message Set IDs.....	60
18.	Bringing It All Together.....	62
18.1.	Learning A UPB Device's Network Password .....	62
18.2.	Reading A UPB Device's Setup Registers .....	63
18.3.	Commanding A UPB Device into the SETUP Mode .....	63
18.4.	Assigning A UPB Device's Network ID & Unit ID .....	64
18.5.	Controlling A UPB Dimming Wall Switch.....	64
18.6.	Activating A Link .....	65
18.7.	Duplicate IDs.....	65
18.8.	Auto-Addressing .....	66
19.	Glossary Of UPB Abbreviations and Terms .....	67

## 1. Overview

### 1.1. Scope

This document describes the Universal Powerline Bus™ (UPB) communication method that has been developed by Powerline Control Systems (PCS). UPB communication is a method of reliably communicating command, control, and status information across an electrical AC powerline. Because of its low cost and high reliability characteristics, the UPB communication method is ideally suited for command and control applications in both the residential and commercial markets.

This document covers such topics as the UPB Physical Communication Method, UPB Data Communications, UPB Packet Acknowledgement, Enumeration Methods, UPB Packet Repeating, the UPB System Model, and UPB Messages.

The details of generating and detecting UPB Pulses is not included in this document but is left to be described in the [UPB Physical Layer Description](#) document.

### 1.2. Prerequisites

It is assumed that the reader of this document has a basic understanding of digital systems and data communications. Such terms as bits, bytes, logical OR, logical AND, and logical XOR should be thoroughly understood.

A good knowledge of the hexadecimal numbering system is also assumed. Unless otherwise specified, this document uses the convention of placing the "0x" prefix before hexadecimal numbers.

It is also helpful if the reader has a basic understanding of the electrical power system in the United States including single-phase, split-phase, and three-phase power arrangements.

## 2. The UPB Physical Communication Method

### 2.1. The UPB Pulse

The UPB powerline communication method consists of transmitting digitally encoded information over the electrical powerline as a series of precisely timed electrical pulses (called UPB Pulses) that are superimposed on top of the normal AC power waveform (sine wave). For various reasons, not explained here, the UPB Pulse is capable traveling large distances over the powerline and even coupling through the power transformer to the other side of a split phase power arrangement. Receiving UPB devices can easily detect and analyze these UPB Pulses and pull out the encoded digital information from them (see Figure 1).

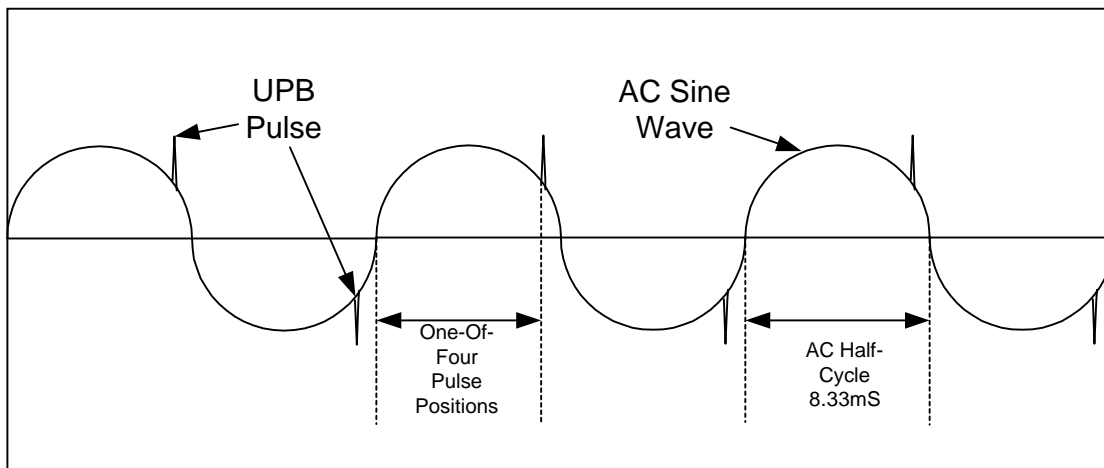


Figure 1: UPB Pulse Communication Method

### 2.2. UPB Pulse Generation

UPB Pulses are generated by charging a capacitor to a high voltage and then discharging that capacitor's voltage into the powerline at a precise time. This quick discharging of the capacitor creates a large "spike" (or pulse) on the powerline that is easily detectable by receiving UPB devices wired large distances away on the same powerline.

### 2.3. UPB Pulse Data Encoding

One UPB Pulse is allowed to be transmitted each half-cycle of the 60Hz AC electrical power cycle. The generation of each UPB Pulse is precisely timed to occur in one of four predefined positions in the half-cycle of the AC powerline. The position of each UPB Pulse determines its value as either 0, 1, 2, or 3. This method of encoding data as a relative position of a pulse is a well-known and used method in digital communications known as Pulse Position Modulation (PPM). Since each UPB Pulse can encode two bits of digital information and

there are 120 AC half-cycles per second (at 60Hz), UPB communication has a raw speed of 240 bits per second (baud). Although this speed isn't fast enough for doing high bandwidth applications it is perfectly adequate for doing command and control communication.

### 2.3.1. The UPB Frame

UPB Pulses are transmitted in a special region toward the end of the AC half-cycle known as the UPB Frame. This region was selected due to its relatively low noise characteristics and for other attributes that make it an optimum position for powerline communications. UPB Frames are synchronized to the low-to-high transition of the AC waveform (known as the AC zero-crossing point) such that one Frame starts  $T_{Frame}$  microseconds after the zero crossing and the other Frame starts 8,333 microseconds (one half-cycle) after the first one (see Figure 2).

### 2.3.2. UPB Pulse Positions

A UPB transmitter is allowed to generate UPB Pulses in one of four nominal positions within the UPB Frame. Each UPB Pulse Position is spaced 160 microseconds apart from each other. UPB transmitters must generate each UPB Pulse at the intended nominal position  $\pm 40$  microseconds in order for receivers to properly detect its value. Figure 2 shows the relative timing of the four UPB Pulse Positions. Each Pulse Position represents a 2-bit encoded data value. Each UPB Pulse can represent a value of 0, 1, 2, or 3.

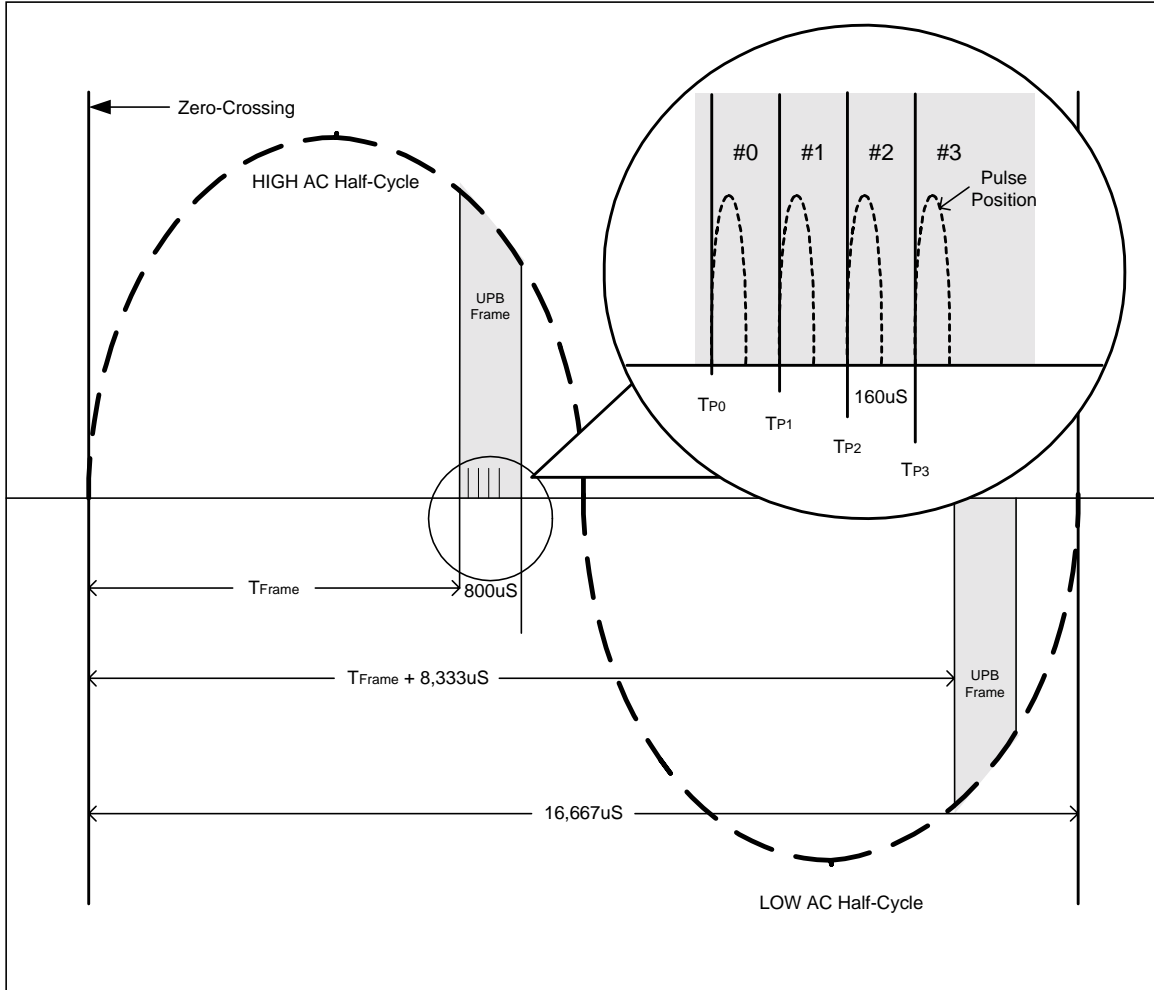


Figure 2: UPB Timing

#### 2.4. UPB Collision Detection

TBD

#### 2.5. UPB Collision Avoidance

TBD

#### 2.6. UPB on a Split-Phase Powerline

TBD

#### 2.7. UPB on a 3-Phase Powerline

TBD

### 3. UPB Data Communications

The UPB communication method would be meaningless unless it were organized in some fashion that allows meaningful data to be communicated between UPB devices. This section describes how UPB Pulses are organized into UPB Communication Packets that convey useful information between UPB devices.

#### 3.1. The UPB Byte

The UPB communication method uses pulses to communicate two bits of information every AC half-cycle. Every four UPB Pulses is logically grouped together and called a UPB Byte. UPB Bytes contain eight bits of information. It takes 4 AC half-cycles or 33.3 milliseconds (at 60Hz) to transmit one UPB Byte.

#### 3.2. The UPB Communication Packet

Each UPB powerline communication is packaged in a variable length structure of UPB Bytes known as a UPB Communication Packet (see Figure 3). All UPB Communication Packets start with a Preamble Byte that helps receiving devices synchronize to the upcoming UPB Communication Packet. The four pulses that make up the Preamble Byte are generated at positions 2-1-1-2.

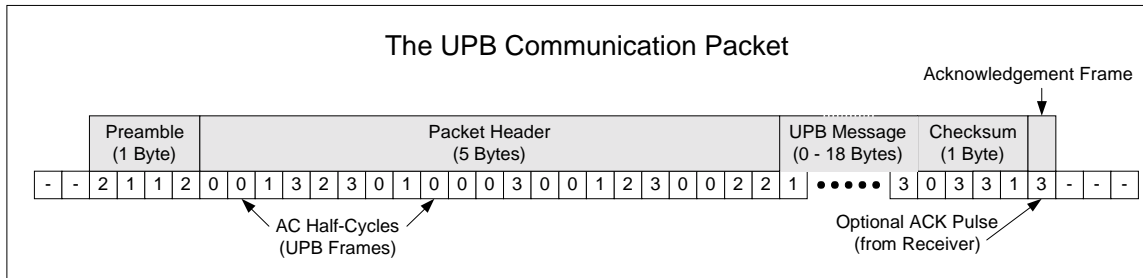


Figure 3: The Standard UPB Communication Packet

After the Preamble Byte, the UPB Communication Packet contains a 5-byte Packet Header. The Packet Header is used to indicate such information as the packet's size, how the packet is to be received, whom the packet is intended for, and who sent the packet. Details on the Packet Header can be found in section 3.4 – "The UPB Packet Header (HDR)".

Following the Packet Header, the UPB Communication Packet contains a variable number of UPB Bytes (0 – 18) that represent the UPB Message. The UPB Message contains all of the information that the transmitting device wishes to convey to the receiving devices. A more detailed description of the UPB Message is presented in section 9 – "The UPB Message".

Following the UPB Message is a 1-byte Checksum field. The Checksum is used to verify the data integrity of the received UPB Communication Packet.



At the end of every UPB Communication Packet is a single frame, known as the Acknowledgement Frame, where the transmitter generates no UPB Pulses. The Acknowledgement Frame is available for a receiving device to generate a special UPB Pulse known as an ACK Pulse that indicates that the UPB Communication Packet was properly received.

### 3.3. UPB Packet Synchronization

The UPB Communication Packet always starts with a 4-pulse Preamble Byte that is used to help the receiving devices synchronize to the upcoming UPB Message. The four Preamble pulses are always transmitted at positions 2-1-1-2. These four pulses are not only intended to mark the start of a new UPB Communication Packet but they are also intended to give the receivers a good idea of the relative size and positions of the UPB Pulses that will make up the upcoming UPB Communication Packet. UPB receivers can use the Preamble Byte to make gain and timing adjustments to allow them to best receive the upcoming UPB Communication Packet pulses.

### 3.4. The UPB Packet Header (HDR)

Every UPB Communication Packet contains a 5-byte Packet Header (see Figure 4). The Packet Header is used to indicate such information as the packet's size, how the packet is to be received, whom the packet is intended for, and who sent the packet. The Packet Header is broken up into four individual fields (CTL, NID, DID, and SID) as described below.

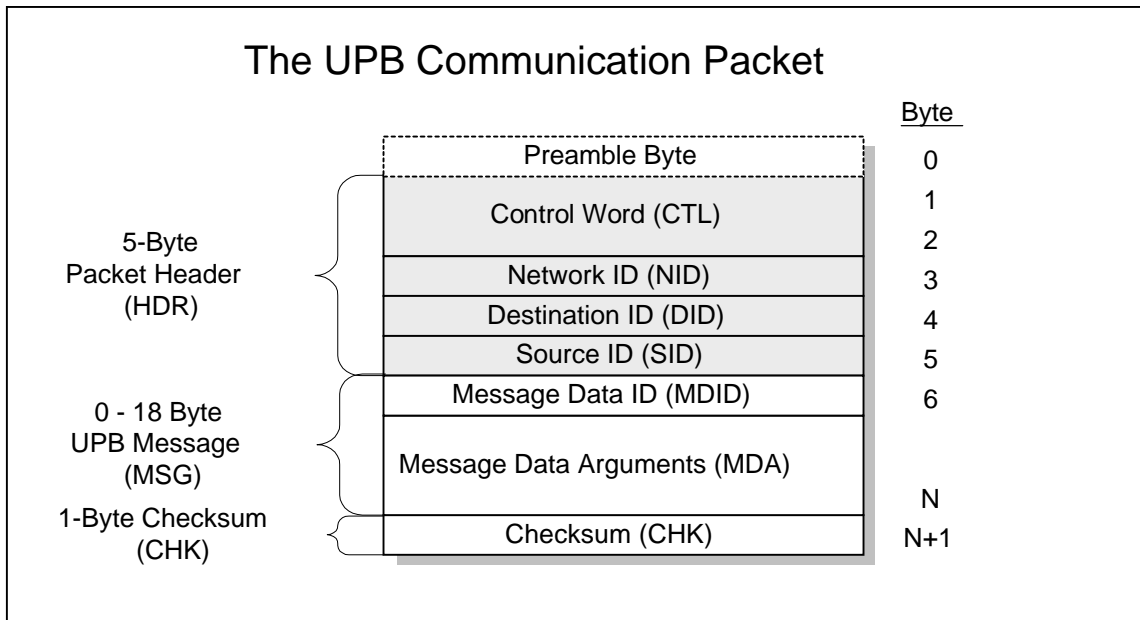


Figure 4: The Packet Header

### **3.4.1. The Control Word (CTL)**

The Control Word of the Packet Header is a two-byte (16-bit) field that has bit fields that are used to indicate such things as the packet's size, how the packet should be received and how it should be responded to, as well as count and sequence information. Since the details of the Control Word are rather complicated, an entire section of this document has been devoted to it. Please refer to Section 4 for more details on the Packet Control Word.

### **3.4.2. The Network ID Field (NID)**

The Network ID field of the Packet Header is used to indicate which UPB network the packet is intended for. Since all UPB devices are capable of receiving any UPB Communication Packet (within range) on the powerline, this field allows separate virtual networks to be formed on the same physical network (powerline). Every UPB device is assigned a single 8-bit Network ID. By assigning the same Network ID to a group of UPB devices, a virtual network can be formed out of those devices. UPB devices will reject any packets that have a different Network ID than their assigned one. Legal values for the Network ID field range from 0 to 255.

### **3.4.3. The Destination ID Field (DID)**

The Destination ID field is used to indicate which individual device, or group of devices (known as links), on a particular UPB network the packet is intended for. Section 3.8 explains in detail how the Destination ID field is decoded and processed.

### **3.4.4. The Source ID Field (SID)**

The Source ID field of the Packet Header is used to indicate whom the packet came from. Every UPB device is assigned a single 8-bit Unit ID in the range from 1 to 250, which it loads in this field when it transmits a UPB Communication Packet. Receivers of a packet can use the SID field to know which device originated the message. This is important information for when a device needs to respond to the packet originator.

## **3.5. The UPB Message (MSG)**

The UPB Message is the actual payload of the UPB Communication Packet. It is a variable length field that can range from 0 to 18 bytes. The content and interpretation of the UPB Message is described in detail in Section 9.

### 3.6. The Packet Checksum (CHK)

Every UPB Communication Packet must end with a 1-byte field called the Packet Checksum. The Packet Checksum is used to verify the integrity of the received packet. The Packet Checksum is computed as follows:

*Sum all of the bytes of the Packet Header and UPB Message fields together. Then take the 2's complement of the sum and truncate the result to 8-bits.*

In this way, a UPB Device that receives a UPB Communication Packet can verify its integrity by summing all of the bytes of the packet (including the Packet Checksum byte) together. The sum of all of the packet bytes should result in a value of 00 for a UPB Communication Packet that was received correctly.

### 3.7. The Acknowledgement Frame

At the end of every UPB Communication Packet is a single frame, known as the Acknowledgement Frame, where the transmitter generates no UPB Pulses. The Acknowledgement Frame is available for a receiving device to generate a special UPB Pulse, known as an ACK Pulse, which is used to inform the transmitting device that the UPB Communication Packet was successfully accepted.

#### 3.7.1. The ACK Pulse

The ACK Pulse is a single UPB Pulse generated by a UPB receiving device that is used to inform the transmitting device that the UPB Communication Packet was accepted (had a correct checksum and ID information). The ACK Pulse is generated in Position #3 of the UPB Frame immediately following the end of a UPB Message. Section 5 describes the ACK Pulse and other acknowledgement methods in more detail.

### 3.8. Destination ID Processing

#### 3.8.1. The Destination ID Field (DID)

The Destination ID field of a UPB Communication Packet is used to indicate which individual device, or group of devices (known as linked devices), on a particular UPB network the packet is intended for. This field has two different interpretations depending on the state of the LNK-bit in the Control Word (CTL). When the LNK bit is set (1) the packet is referred to as a Link Packet. When the LNK bit is clear (0) the packet is referred to as a Direct Packet. This mechanism effectively produces a total address space of 512 addresses within every UPB network. There are 256 addresses dedicated to the link addressing mode and 256 address dedicated to the direct addressing mode.

The interpretation and validation of the DID field for both types of UPB Communication Packets is shown in Figure 5 and explained as follows.

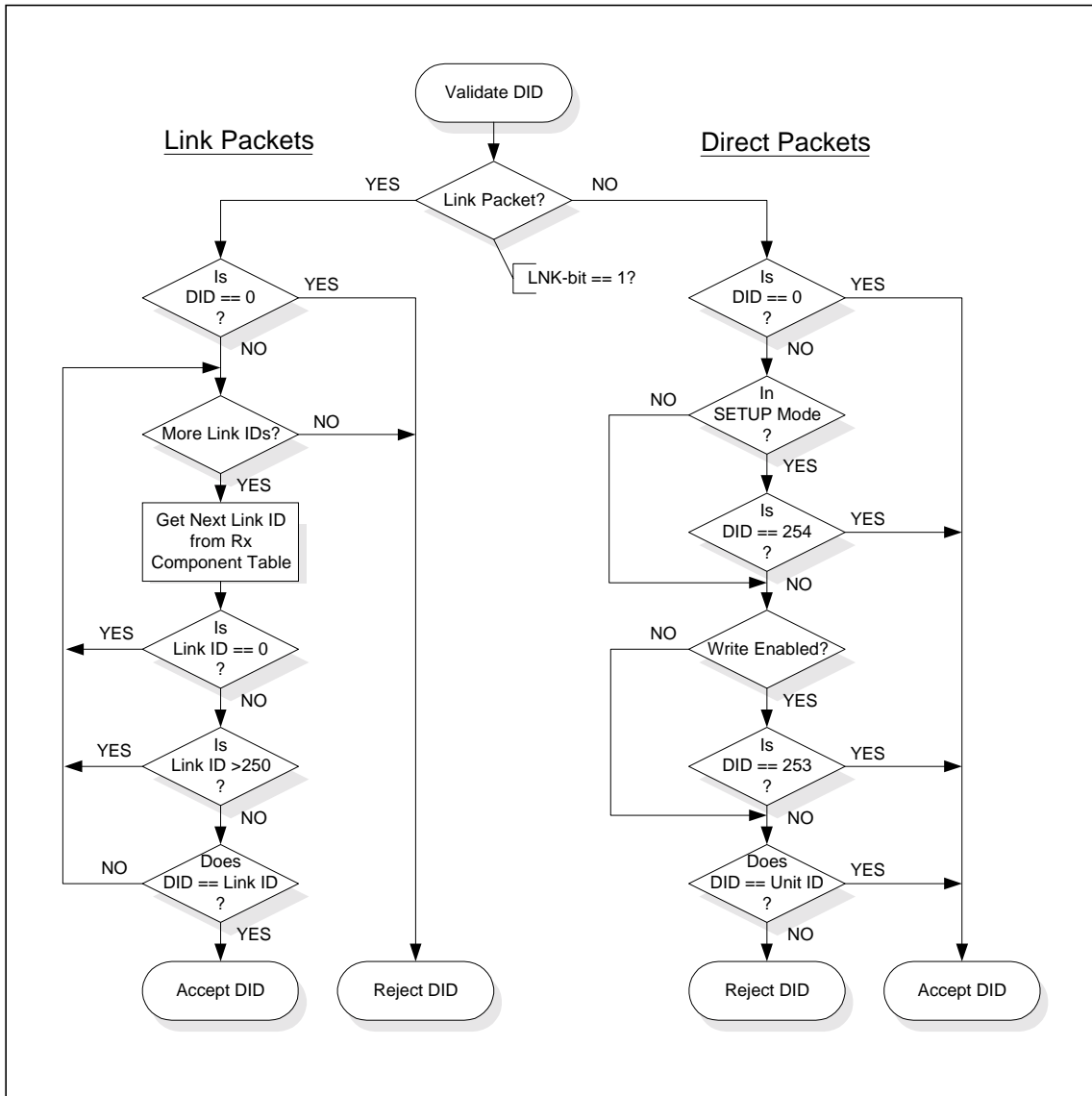


Figure 5: Destination ID Validation Process

### 3.8.1.1. The Link Packet

In a Link Packet, the Destination ID field is interpreted as a Link ID. Every UPB device can be assigned Link IDs, in the range from 1 to 250, which are used to create virtual “links” between UPB device components. UPB devices will accept Link Packets that have a Destination ID that matches one of their Receive Component’s valid Link IDs. Legal values for the Destination ID field in a Link Packet range from 1 to 250. Link Packets with an illegal Destination ID (0 and 251 – 255) will be rejected (see Figure 5). Refer to Section 8.6 for an

explanation of UPB Component Linking to see how Link Packets are used in the UPB System Model.

### 3.8.1.2. The Direct Packet

In a Direct Packet, the Destination ID is interpreted as a Unit ID. Every UPB device is assigned a single 8-bit Unit ID in the range from 1 to 250, which is used to distinguish that device from any other device on the same UPB network. UPB devices will filter out (reject) any Direct Packets that have a Destination ID that is different from their assigned Unit ID (see Figure 5).

Values for the Destination ID field in a Direct Packet range from 1 to 250. In addition, special meaning is assigned to the other Destination ID values (0 and 251 – 255) as described in Table 1 below.

DID	Name	Description
000	Broadcast DID	Accepted by all UPB devices.
251	Unused	Reserved for the future
252	Unused	Reserved for the future
253	Write Enabled DID	Accepted by all UPB devices whose Setup Registers are write enabled.
254	SETUP Mode DID	Accepted by all UPB devices that have been manually put into SETUP Mode.
255	Default DID	Reserved to be used by devices that have never been setup before.

Table 1: Special Direct Destination IDs

#### 3.8.1.2.1. Broadcast Packets

Direct Packets that use 0 as the Destination ID are referred to as Broadcast Packets. All devices on the packet's addressed network accept Broadcast Packets.

#### 3.8.1.2.2. Unprotected Packets

Direct Packets that use 253 as the Destination ID are referred to as Unprotected Packets. All write enabled (unprotected) devices on the packet's NID accept Unprotected Packets.

#### 3.8.1.2.3. SETUP Mode Packets

Direct Packets that use 254 as the Destination ID are referred to as Setup Mode Packets. All devices that are currently in the SETUP Mode on the packet's NID accept Setup Mode Packets. Refer to Section 8.5.2 for an explanation of the SETUP Mode.

### 3.9. UPB Communication Packet Priorities

TBD

#### 4. The Packet Control Word

The first two-bytes of a UPB Packet Header are known as the UPB Packet Control Word. The Control Word (Figure 6) contains bit fields that are used to indicate such information as: how the UPB Communications Packet should be received and how it should be responded to, as well as its length and sequence information.

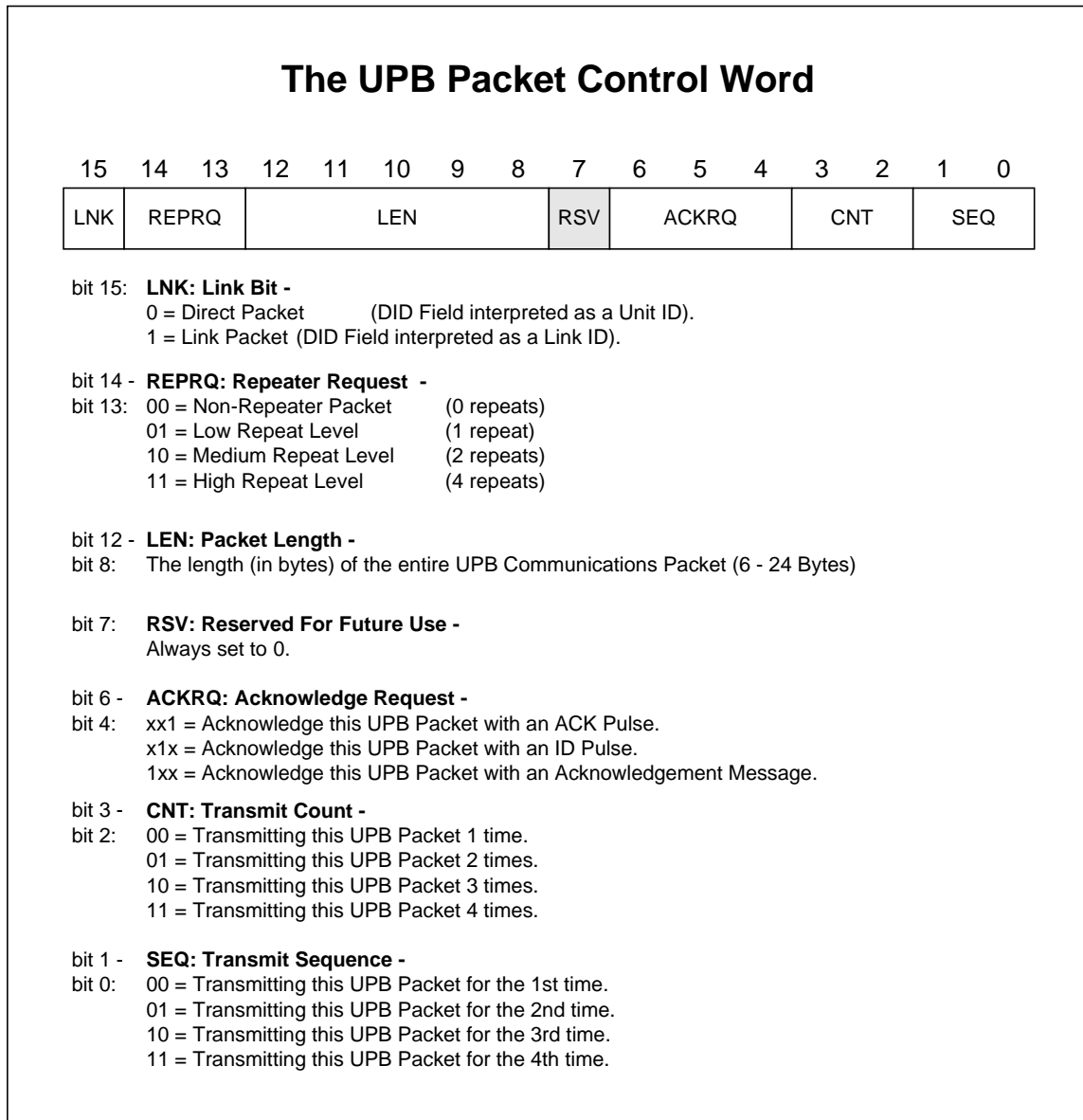


Figure 6: The UPB Packet Control Word

The remainder of this section will describe the various bit fields of the Control Word in detail.

#### **4.1. The Packet Length Field**

The Packet Length field (LEN) is a 5-bit field that is used to indicate the number of bytes in the entire UPB Communication Packet (not including the Preamble Byte). A UPB Communication Packet can range anywhere from 6 to 24 bytes in length. If the contents of this field are outside the legal range, the receivers will reject the UPB Communication Packet.

#### **4.2. The Link Bit**

The Link bit (LNK) of a UPB Communication Packet distinguishes that packet as either a Link Packet (LNK = 1) or as a Direct Packet (LNK = 0). Link Packets are intended for communicating to a group of devices that all share a common Link ID. Direct Packets on the other hand, are intended for communicating to a single device. The Destination ID field of a Link Packet is addressed to the Link IDs of a device's Receive Component Table whereas the Destination ID field of a Direct Packet is addressed to the single unique Unit ID of a device.

#### **4.3. The Repeater Request Field**

The Repeater Request field (REPRQ) is a 2-bit field that is used to request a special UPB device, known as a UPB Repeater, to repeat this UPB Communication Packet a specified number of times. When this field is non-zero, the packet is considered to be a Repeater Packet and when this field is clear (zero) the message is considered to be a Non-Repeater Packet. Section 7 describes UPB Packet repeating in more detail.

#### **4.4. The Acknowledge Request Field**

The Acknowledge Request field (ACKRQ) is a 3-bit field used to request one or more acknowledgements from the receiving device (or devices) that it received and accepted the UPB Communication Packet. There are three types of acknowledgement methods available: an ACK Pulse, an ID Pulse, and an Acknowledgement Message. Section 5 describes UPB Packet Acknowledgement in more detail.

#### **4.5. The Transmission Count Field**

The Transmission Count field (CNT) is a 2-bit field that indicates the total number of times the packet is to be transmitted. UPB Communication Packets can be transmitted up to four times in a row. Transmitting UPB Communication Packets more than once increases the chances that all receiving devices will properly receive that packet. As seen in Table 2, the binary value in the CNT field is always one below the total number of packet transmissions being performed.



CNT	Description
00	Transmitting this UPB Packet 1 time.
01	Transmitting this UPB Packet 2 times in a row.
10	Transmitting this UPB Packet 3 times in a row.
11	Transmitting this UPB Packet 4 times in a row.

Table 2: The Transmission Count Field (CNT)

#### 4.6. The Transmission Sequence Field

The Transmission Sequence field (SEQ) is a 2-bit field that indicates the relative sequence of the current UPB Communication Packet transmission. As seen in Table 3, the SEQ field will indicate the first time the UPB Packet is transmitted with a value of 00 and it will indicate the second time the UPB Packet is transmitted with a value of 01, and it will indicate the third time the UPB Packet is transmitted with a value of 10, and it will indicate the fourth time the UPB Packet is transmitted with a value of 11.

SEQ	Description
00	Transmitting this UPB Packet for the 1 <sup>st</sup> time.
01	Transmitting this UPB Packet for the 2 <sup>nd</sup> time.
10	Transmitting this UPB Packet for the 3 <sup>rd</sup> time.
11	Transmitting this UPB Packet for the 4 <sup>th</sup> time.

Table 3: The Transmission Sequence Field

##### 4.6.1. Multiple Transmissions

The SEQ field combined with the CNT field will give each UPB Communication Packet in a multiple transmission a unique identity. As an example, suppose a UPB Communication Packet is being transmitted four times in a row. The first transmission will be indicated with the SEQ/CNT fields set to 00/11 telling receivers that this is the first transmission of four. The next transmission will be indicated with the SEQ/CNT fields set to 01/11 telling receivers that this is the second transmission of four. The next transmission will be indicated with the SEQ/CNT fields set to 10/11 telling receivers that this is the third transmission of four. The final transmission will be indicated with the SEQ/CNT fields set to 11/11 telling receivers that this is the fourth transmission of four.

## 5. UPB Packet Acknowledgements

UPB Packet Acknowledgement is a method available to transmitting devices to detect if the intended receiver accepted its transmitted UPB Communication Packet. There are three types of acknowledgement methods available: an ACK Pulse, an ID Pulse, and an Acknowledgement Message. The packet sender can request none, one, two, or all three of these acknowledgement methods. As shown in Table 4, a separate bit exists in the ACKRQ field (of the Control Word) for requesting each type of acknowledgement method. A UPB device will only generate a requested acknowledgement if that device accepts the UPB Communication Packet.

CTL Bit	ACK Request Type	Acknowledgement Description
ACK-bit (4)	ACK Pulse Requested	A UPB Pulse is generated in Position #3 exactly one frame (half-cycle) after the end of this UPB Communication Packet. This is known as an ACK Pulse.
ID-bit (5)	ID Pulse Requested	A UPB Pulse is generated in Position #3 of the frame that corresponds to the Unit ID (UID) of the receiving device after the end of this UPB Communication Packet. This is known as an ID Pulse.
MSG-bit (6)	Acknowledge Message Requested	The receiving device transmits a complete Acknowledgement Message after the end of this UPB Communication Packet.

Table 4: The Acknowledge Request Field

### 5.1. The ACK Pulse

The ACK Pulse is a single UPB Pulse that is generated in Position #3 of the UPB Frame immediately following the end of a UPB Message. Its purpose is to indicate to a transmitting device that the intended receiving device accepted the UPB Communication Packet. A UPB device only generates the ACK Pulse when it receives and accepts a UPB Communication Packet that has the ACK-bit set to 1.

### 5.2. The ID Pulse

The ID Pulse is a single UPB Pulse that is generated in Position #3 of the UPB Frame (immediately following the end of a UPB Message) that corresponds to the receiving device's Unit ID. For instance, if the receiving device's Unit ID is 148 then it will generate its ID Pulse 148 frames (half-cycles) after the end of the received UPB Message. The main purpose of the ID Pulse is to indicate to the

transmitting device which receiving devices properly received a broadcasted message. A UPB device only generates the ID Pulse when it receives and accepts a UPB Communication Packet that has the ID-bit set to 1. The ID Pulse is mainly intended to be used for performing UPB Enumeration as described in Section 6.

NOTE: Devices that broadcast UPB Messages with the ID-bit set to 1 should allow for 256 frames after the packet for all possible ID Pulses to be generated on the powerline.

### **5.3. The Acknowledgement Message**

The Acknowledgement Message is a short UPB Message that is sent back to the transmitting device that indicates that its UPB Communication Packet was received and accepted. The Acknowledgement Message is described in more detail in Section 14.1.10 - The "Acknowledgment" Report. A UPB device only generates the Acknowledgment Message when it receives and accepts a UPB Communication Packet that has the MSG-bit set to 1.

## 6. Enumeration Methods

This section describes methods, built into the UPB system, for quickly discovering the existence of particular UPB devices based on their IDs. Each method discussed is based on the use of ACK Pulses and ID Pulses to quickly learn the Network ID and Unit IDs of all existing devices on a given powerline.

### 6.1. The ACK-Bit

UPB Communication Packets that have the ACK-bit set to 1 request all receiving devices to generate an ACK Pulse to occur in the last Frame of that packet.

### 6.2. The ACK Pulse

The ACK Pulse is a single UPB Pulse that is generated in Position #3 of the UPB Frame immediately following the end of a UPB Message. Its purpose is to indicate to a transmitting device that the intended receiving device accepted the UPB Communication Packet. The receiving device only generates this ACK Pulse if the transmitting device requested it by setting the ACK-bit in the Control Word of the Packet Header.

### 6.3. The ID-Bit

UPB Communication Packets that have the ID-bit set to 1 request all receiving devices to generate an ID Pulse to occur in a Frame that corresponds to the receiving devices' Unit ID.

### 6.4. The ID Pulse

An ID Pulse is used to quickly indicate the Unit ID of a receiving device to the transmitting device. Like the ACK Pulse, the ID Pulse is simply a UPB Pulse generated in Position #3 by the receiving device (or devices). The ID Pulse, however, is placed in the one (of 256 possible) UPB Frame that corresponds to the receiving device's own Unit ID. The receiving device only generates this ID Pulse if the transmitting device requested it by setting the ID-bit in the Control Word of the Packet Header.

### 6.5. UPB Network Enumeration

UPB Network Enumeration is a method available for transmitting devices to quickly determine which Network IDs are occupied (by UPB devices). This method involves a transmitting device broadcasting a UPB Communication Packet that has the ACK-bit set to 1. The ACK-bit requests the receiving device(s) to generate an ACK Pulse at the end of the packet. If the UPB Communication Packet contains an ACK Pulse then its NID is occupied by at least one device. If the UPB Communication Packet does not contain an ACK Pulse then its NID is completely unoccupied. A separate UPB Communication

Packet is broadcasted to each NID in question to obtain an indication of every NID occupied on the powerline.

## 6.6. UPB Device Enumeration

UPB Device Enumeration is a method available for transmitting devices to quickly determine the Unit IDs of all receiving devices that exist on a particular UPB network. This method involves a transmitting device broadcasting a UPB Communication Packet that has the ID-bit set to 1. The ID-bit requests the receiving device(s) to generate an ID Pulse. The transmitting device simply monitors the powerline for ID Pulses in the 256 UPB Frames that follow the transmitted packet. Frames that contain an ID Pulse indicate that their corresponding Unit ID is occupied in the UPB network. Frames that do not contain an ID Pulse indicate that their corresponding Unit ID is not occupied. After 256 Frames (2.13 seconds) the transmitting device will have an indication as to every Unit ID occupied on a particular NID.

## 7. UPB Packet Repeating

UPB Packet Repeating is a method available to increase the communication reliability of a UPB network by having special devices installed that repeat packets to the rest of the UPB network devices. The UPB transmission range (and communication reliability) of a UPB device can be greatly extended by sending a Repeater Packet to a remote Repeater device requesting it to repeat the packet (one or more times) to other UPB devices that are even farther away. This method is employed in commercial applications but is not used in residential applications. UPB Packet Repeating is also an excellent method to allow UPB communication over a 3-phase powerline (see Figure 7).

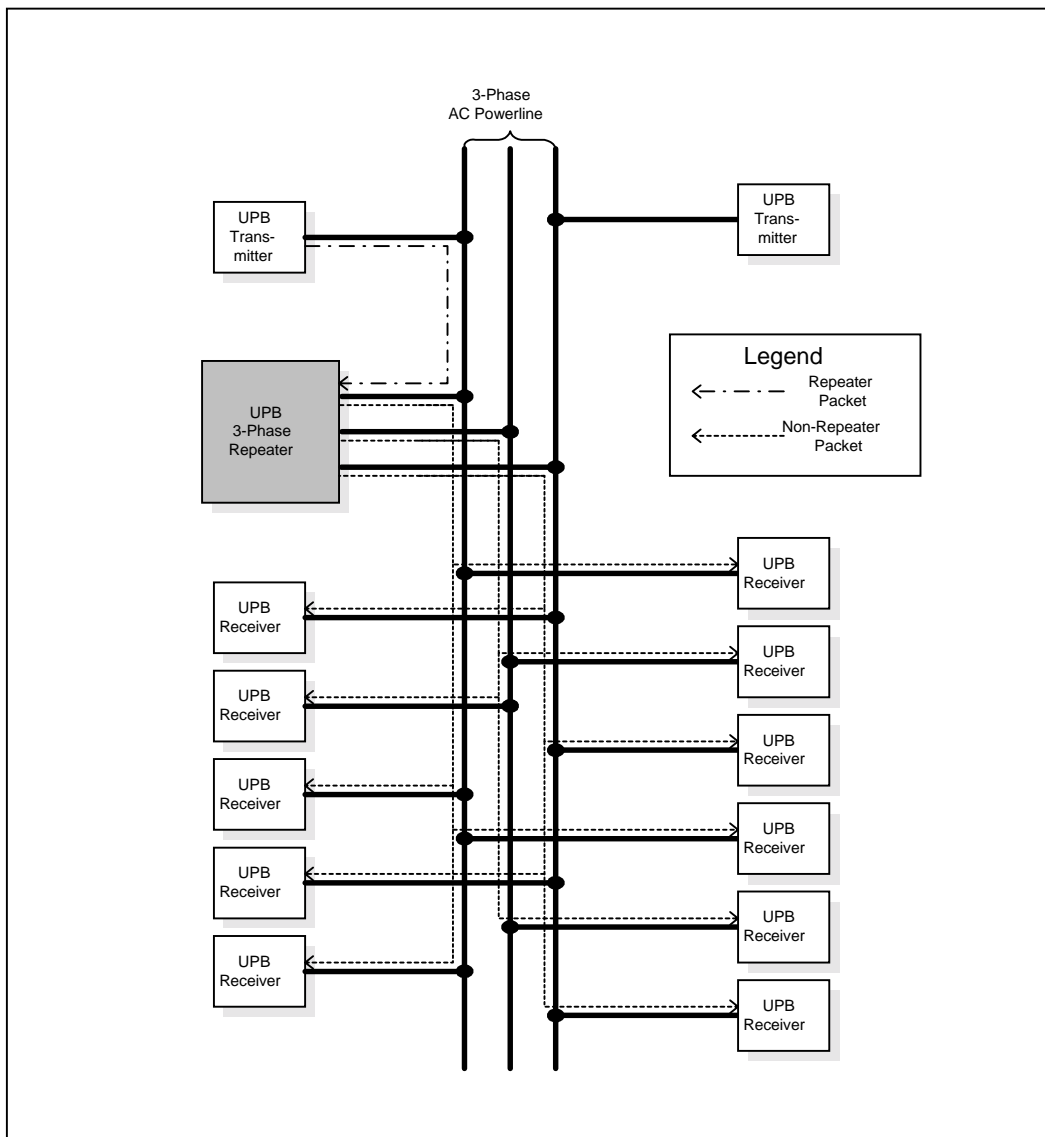


Figure 7: UPB Packet Repeating

## 7.1. Repeater Packets

The Repeater Request field (REPRQ) of the Control Word is used to request a special UPB device, known as a UPB Repeater, to repeat this UPB Communication Packet a specified number of times. When this field is non-zero, the packet is considered to be a Repeater Packet and when this field is clear (zero) the packet is considered to be a Non-Repeater Packet.

### 7.1.1. Repeater Request Levels

By using the two bits in the REPRQ field, UPB Repeater devices can be requested to repeat (i.e. clear the REPRQ field and re-transmit) the UPB Communication Packet either once, twice, or four times as shown in Table 5 below.

REPRQ	Repeater Request Level	Description
01	Low	Repeat the UPB Communication Packet 1 time.
10	Medium	Repeat the UPB Communication Packet 2 times.
11	High	Repeat the UPB Communication Packet 4 times.

Table 5: Repeater Request Levels

## 7.2. Repeater Devices

UPB Repeater devices are a special class of UPB device that are designed to only accept a Repeater Packet and then retransmit it as a Non-Repeater Packet (i.e. clear the REP field and re-transmit it). All other UPB devices are designed to reject Repeater Packets and only accept Non-Repeater Packets (i.e. REPRQ = 00). UPB Repeater devices are used to increase the communication reliability in commercial applications but are not used in residential applications. UPB Repeater devices can also be used in a three-phase wiring arrangement to repeat packets from one phase to the other phases.

## 8. The UPB System Model

UPB Messages are designed to operate on UPB devices that conform to a standard hardware and software model known as the “UPB System Model” (see Figure 8 below).

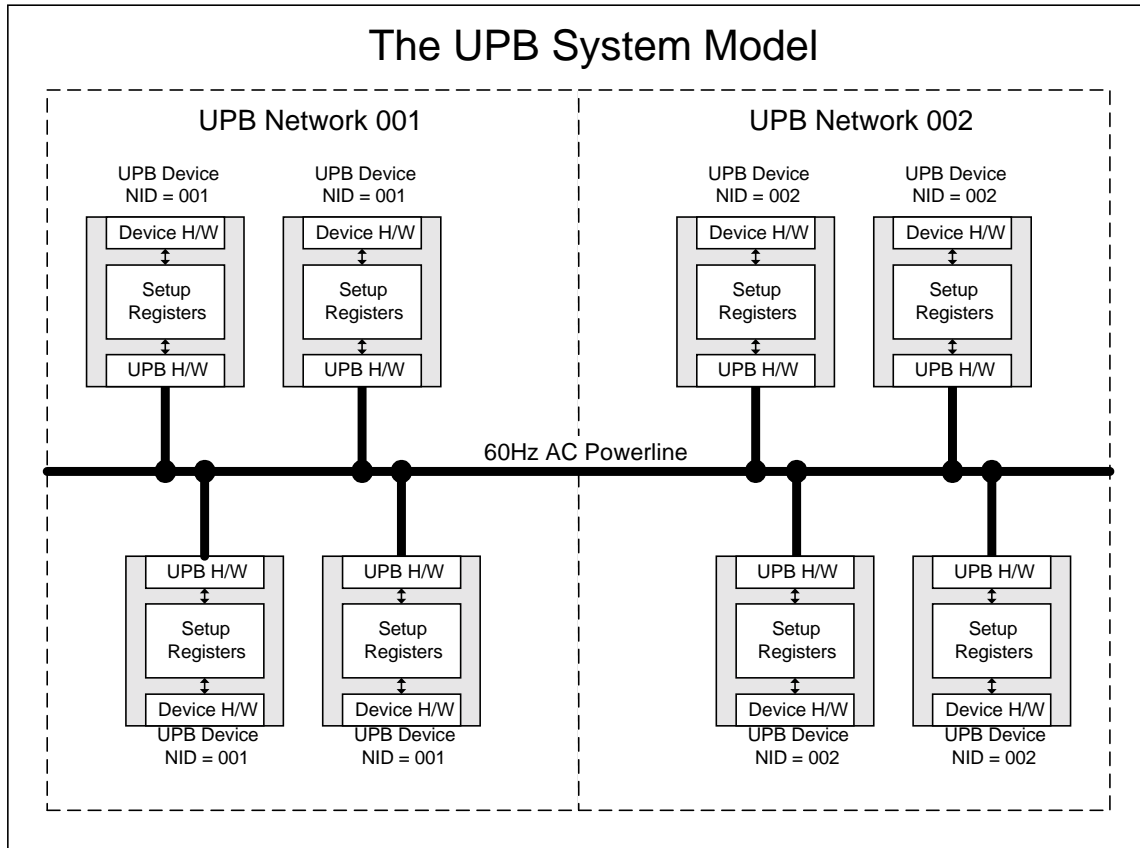


Figure 8: The UPB System Model

In this model, each UPB device is assumed to contain a core set of hardware and software to allow it to receive and transmit UPB Communication Packets on the powerline. The model also assumes each UPB device contains a set of at least 64 non-volatile 8-bit registers known as Setup Registers. The Setup Registers are used as the main interface to configure and customize the UPB device operation. Each UPB device shall respond to a standard set of UPB Messages (known as the Core Command Message Set) that provide the ability to read and write its Setup Registers. Each UPB device shall have the ability to operate in at least two different modes of operation: Normal Mode and SETUP Mode.



### 8.1. UPB Setup Registers

UPB Setup Registers are a block of at least 64 (up to 256) non-volatile 8-bit registers that are used to hold all of the configuration information about that UPB device. The first 64 Setup Registers are known collectively as the UPBID and are defined to have the same meaning in every UPB device. The remaining Setup Registers are known collectively as Configuration Registers and are open for use by the individual application designer. Special UPB commands exist for reading, writing, and modifying the Setup Registers (refer to section 10 – “The UPB Core Command Set” for a list of UPB commands that affect the Setup Registers).

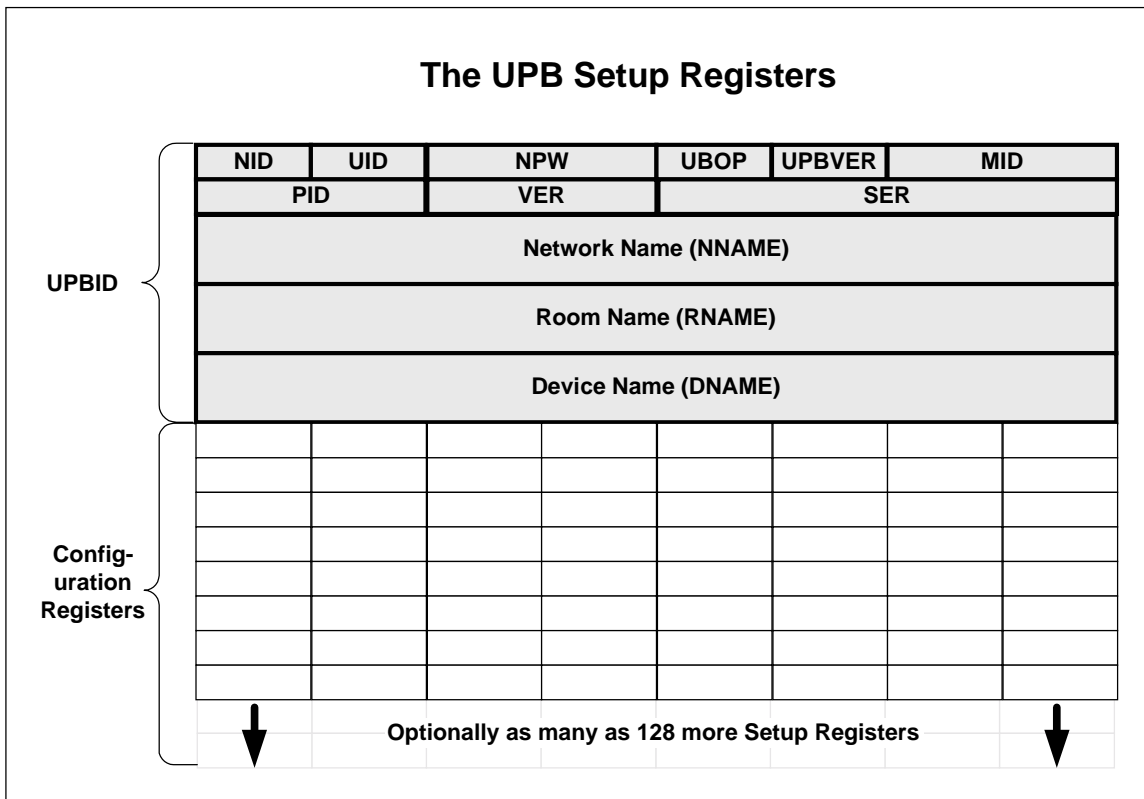


Figure 9: The UPB Setup Registers

### 8.2. The UPBID

In the UPB System Model, the first 64 Setup Registers of each device are known collectively as the UPBID. The UPBID contains information that uniquely identifies the individual UPB device. Table 6 briefly describes the standard meaning of each UPBID register. These registers are described in more detail within this section.

Setup Register Field Name	Reg #	Description
Network ID (NID)	0x00	Unique identifier (1 – 255) for the UPB Network that this device communicates on.
Unit ID (UID)	0x01	Unique identifier (1 – 250) for this UPB device.
Network Password (NPW)	0x02 – 0x03	Two-byte password (MSB first) designed to keep unauthorized users from modifying the Setup Registers of this device.
UPB Options (UPBOP)	0x04	Identifies unique UPB options that are enabled for this device.
UPB Version (UPBVER)	0x05	Identifies the version of the UPB specification this device conforms to.
Manufacturer ID (MID)	0x06 – 0x07	Unique two-byte identifier (MSB first) for the manufacturer of this UPB device.
Product ID (PID)	0x08 – 0x09	The manufacturer's unique two-byte product identifier (MSB first) for this UPB device.
Firmware Version (FWVER)	0x0A – 0x0B	The manufacturer's unique two-byte version number (MSB first) of the firmware used in this UPB device.
Serial Number (SERNUM)	0x0C – 0x0F	The manufacturer's unique four-byte serial number (MSB first) for this UPB device.
Network Name (NNAME)	0x10 – 0x1F	A human readable (ASCII) name for the UPB Network that this device communicates on.
Room Name (RNAME)	0x20 – 0x2F	A human readable (ASCII) name for the Room that this UPB device is installed in.
Device Name (DNAME)	0x30 – 0x3F	A human readable (ASCII) name for this UPB device.

Table 6: The UPBID Registers

### 8.2.1. The Network ID (UPB Networks)

The first Setup Register (Register 0x00) of each UPB device is known as the Network ID (NID). Every UPB device is assigned a single 8-bit Network ID that can range from 000 to 255. By assigning the same Network ID to a group of UPB devices, a virtual network can be formed by those devices. Figure 8 shows an example where two virtual networks are formed on the same physical powerline by assigning one group of devices one Network ID (001) and another group of devices another Network ID (002). UPB devices will reject any UPB Messages that have a different Network ID than their own. Network ID values of 000 and 251 – 255 are reserved for special purposes as described later.

### 8.2.2. The Unit ID

The second Setup Register (Register 0x01) of each UPB device is known as the Unit ID (UID). Every UPB device on a UPB network is assigned a single 8-bit Unit ID that can range from 001 to 255. The Unit ID is used to distinguish each individual device from all other devices on the same UPB Network. UPB devices will reject Direct UPB Communication Packets that are addressed to a Unit ID that is different than their own. Unit ID values of 000 and 251 – 255 are reserved for special purposes as described later.

### 8.2.3. The Network Password

Every UPB device in the UPB System Model has a two-byte field in its UPBID known as the Network Password (NPW). The Network Password is a special 16-bit code (0 – 65,536) that is assigned to a UPB device that must be supplied when trying to do such things as put the device into SETUP Mode and disable its write protection. In the UPB System Model, it is recommended (but not mandatory) that every UPB device in the same UPB Network get assigned the same Network Password. This simply eliminates the need to remember multiple passwords.

Note: The Network Password is read-protected. It can only be read while the device's Setup Register Write-Protection is disabled. While Setup Register Write-Protection is enabled, the Network Password always reads as 0x0000. This offers a level of security to the system in that an unauthorized user cannot read your Network Password without already knowing it (or by physically touching your device in a special sequence as described later).

### 8.2.4. The UPB Options Register

The UPB Options Register (UPBOP) is a one-byte field used to identify various UPB options that are enabled/disabled on this device. Table 7 defines the meaning of each bit in the UPB Options Register.

Bit	Name	Description
0	Repeater	A '1' in this bit indicates that the device is to be installed in a UPB Repeater network.
1	Reserved1	Reserved for the future.
2	Reserved2	Reserved for the future.
3	Reserved3	Reserved for the future.
4	Reserved4	Reserved for the future.
5	Reserved5	Reserved for the future.
6	Reserved6	Reserved for the future.
7	Reserved7	Reserved for the future.

Table 7: UPB Options Register

### 8.2.5. The UPB Version

The UPB Version (UPBVER) is a one-byte field that identifies the version of the UPB specification this device conforms to. **Note: this Register is read-only and cannot be written to except by the device manufacturer.**

### 8.2.6. The Manufacturer ID

The Manufacturer ID (MID) is a two-byte field (MSB first) used to identify the manufacturer of the device. Each UPB device manufacturer is assigned a unique Manufacturer ID number that is placed in this field. Refer to the UPB Device Manufacturers Catalog document for a list of the currently assigned MIDs. **Note: this Register is read-only and cannot be written to except by the device manufacturer.**

### 8.2.7. The Product ID

The Product ID (PID) is a two-byte field (MSB first) used to identify the product code of the device. Each UPB device manufacturer is assigned a set of unique product codes (0 – 65,535) that identify each of their individual UPB devices. Refer to the UPB Device Manufacturers Catalog document for a list of the currently assigned PIDs. **Note: this Register is read-only and cannot be written to except by the device manufacturer.**

### 8.2.8. The Firmware Version

The Firmware Version (FWVER) is a two-byte field (MSB first) used to hold the version number (0 – 65,535) of the firmware used in the product. Each UPB device manufacturer is free to create its own version numbering scheme. **Note: this Register is read-only and cannot be written to except by the device manufacturer.**

### 8.2.9. The Serial Number

The Serial Number (SERNUM) is a four-byte field (MSB first) used to hold the serial number (0 – 4,294,967,295) of the particular product. Each UPB device manufacturer is free to create its own serial numbering scheme. **Note: this Register is read-only and cannot be written to except by the device manufacturer.**

### 8.2.10. The Network Name

The Network Name is a sixteen-byte field used to hold a string of sixteen ASCII characters to represent the name of the UPB Network that the device is installed in (e.g. “Barbara’s House”).

### 8.2.11. The Room Name

The Room Name is a sixteen-byte field used to hold a string of sixteen ASCII characters to represent the name of the room that the device is installed in (e.g. “Master Bedroom”).

### 8.2.12. The Device Name

The Device Name is a sixteen-byte field used to hold a string of sixteen ASCII characters to represent the name of the UPB device (e.g. “Wall Switch #1”).

### 8.3. The Configuration Registers

Any additional Setup Register(s) not used by the UPBID are referred to as Configuration Registers. A UPB Device can have as many as 192 Configuration Registers. These Configuration Registers are available to be used for whatever purposes the application designer desires.

#### 8.3.1. Device Components

One possible use for the Configuration Registers is for an optional, but very powerful, UPB concept known as Device Components. Device Components are logical objects (records) that the UPB device has, usually associated with physical entities on the device (such as pushbuttons, switches, indicators, input channels, output channels, etc.), that are intended to either receive or transmit UPB Link Packets for control purposes. Device Components (Figure 10) are classified into one of two types: either as Receive Components, which receive UPB Link Packets, or as Transmit Components, which transmit UPB Link Packets.

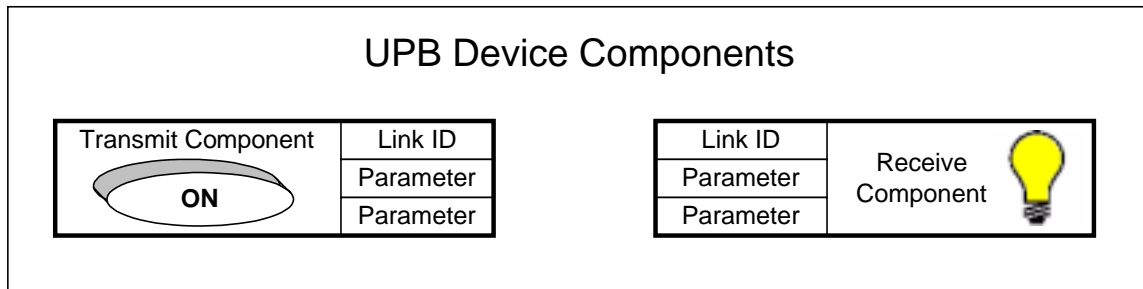


Figure 10: Examples of UPB Device Components

#### 8.3.2. Why Use Device Components

By using Device Components, a single pushbutton on one device can be configured to transmit UPB Link Packets addressed to various lighting channels on other UPB devices, thus creating powerful lighting “Scenes”. Device Components allow the operation of the device to be modified and configured by a setup tool so that the system operation can be customized for each user. Many other uses for Device Components can be thought of.

#### 8.3.3. Device Component Records

Each Device Component record contains (as its 1<sup>st</sup> byte) an associated 8-bit ID known as its Link ID along with optional parameters that are stored in the device’s Configuration Registers (see Figure 10). The size and content (other than the Link ID) of the individual Device Component record is up to the individual device manufacturer to design. There is no limit, except for the amount of Configuration Registers available, as to how many Receive Components and

Transmit Components a UPB device can have. Special UPB Link Packets are available (see section 11 – “The Device Control Command Set” for more details) that allow Transmit Components on one device to control Receive Components on other devices.

### 8.3.4.Link IDs

Besides having a unique Unit ID to identify an individual UPB device, each device can optionally be assigned additional identifiers, called Link IDs, which can be used to group (or link) a set of UPB Device Components together. Each Device Component Record can either be assigned a Valid Link ID (from 1 to 250) or an Invalid Link ID (251 – 255).

### 8.3.5.Linking Device Components Together

Two or more Device Components that are assigned the same Valid Link ID are considered to be “linked” together (see Figure 11). Device Components that have been assigned an invalid Link ID (251 – 255) are considered “unused” and will not transmit or receive UPB Link Packets.

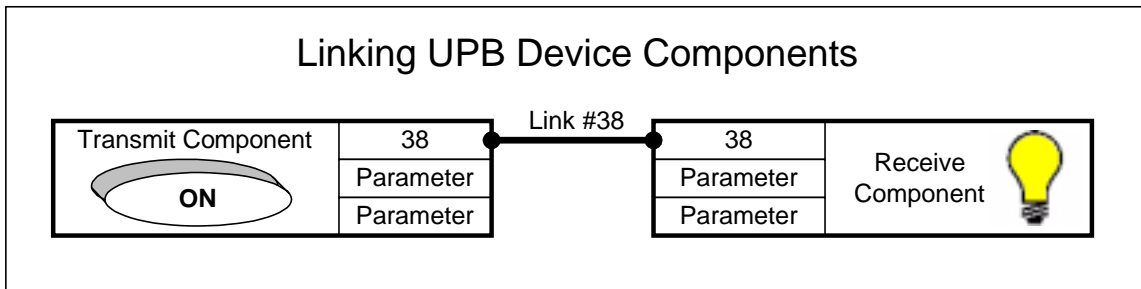


Figure 11: Linking UPB Device Components

### 8.3.6.Link Packets

Once UPB Device Components are “linked” together (by being assigned the same Link ID), they can be controlled with Link Packets. Link Packets are simply UPB Communication Packets that have their LNK-bit set to 1 (refer to section 4.2 for an explanation of the LNK-bit). UPB devices use the Link IDs in their Transmit Components to build and transmit Link Packets with that Link ID inserted in the Destination ID field. Whenever a UPB device receives a Link Packet it will check the Destination ID (DID) field to see if it matches one of its Valid Link IDs in its list of Receive Components. If it does match one of the Valid Link IDs, the packet is accepted. If it doesn’t match, then the packet is rejected. Figure 5 shows how a UPB device processes the Destination ID of a Link Packet.

### 8.3.7.Controlling Linked Device Components

The UPB Device Control Command Set (refer to section 11) contains many commands that can be used to control both individual devices as well as “linked”

Device Components. There are commands to request that the receiving device(s) go to specified states or levels at specified rates. There are also commands that request the receiving device(s) to start and stop fading, start blinking, store their current states, etc.

Whenever these commands are packaged in a Direct Packet they will affect only an individual device. Whenever these commands are packaged in a Link Packet they will affect all of the “linked” Device Components associated with that Link ID. A single Link Packet command can therefore request a set of “linked” lighting channels on separate devices to start fading to 100% at a specified fade rate. Then, with another Link Packet command they can be requested to stop fading.

### **8.3.8. Activating/Deactivating Links**

The UPB Device Control Command Set (refer to section 11) contains two commands that are especially designed for controlling “linked” Device Components: the Activate Link Command and the Deactivate Link Command.

The Activate Link Command is used to “activate” all Receive Components on a specified link (Link ID). The term “activate” is a general term that usually means to turn to the ON state but each individual device manufacturer can define it differently for their device. To some devices the term “activate” might mean to turn ON a lighting channel to a preset level (e.g. 50%). To other devices the term “activate” might mean to blink an indicator every second. The parameters in the Receive Component Record can be used to configure what to do to the associated physical component when it is “activated”.

The Deactivate Link Command is used to “deactivate” all Receive Components on a specified link (Link ID). The term “deactivate” is a general term that usually means to turn to the OFF state but each individual device manufacturer can define it differently for their device. To some devices the term “deactivate” might mean to turn a lighting channel to 0%. To other devices the term “deactivate” might mean to set an indicator to red. The parameters in the Receive Component Record can be used to configure what to do to the associated physical component when it is “deactivated”.

### **8.3.9. Creating Lighting “Scenes”**

By properly configuring the parameters in the Receive Component Records of a network of UPB lighting devices, different lighting “Scenes” can be easily triggered by a single Activate Link command. All devices that have a Receive Component Record with that Link ID will respond to that Activate Link command based on how their parameters were previously configured (by a setup tool) in their associated Receive Component Record. One Receive Component Record may have been configured to turn its lighting channel to 100%, another Receive Component Record may have been configured to turn its lighting channel to



75%, and yet a third Receive Component Record may have been configured to turn its lighting channel to 0% (off). With a single Activate Link command this whole previously configured lighting “Scene” will be triggered.

The Activate/Deactivate mechanism does not have to stop at lighting systems. Just about any automated system can have pre-configured modes or scenes that can be “activated” by a single UPB Link Packet.

### **8.3.10. Linked Device Components Example**

**TBD**

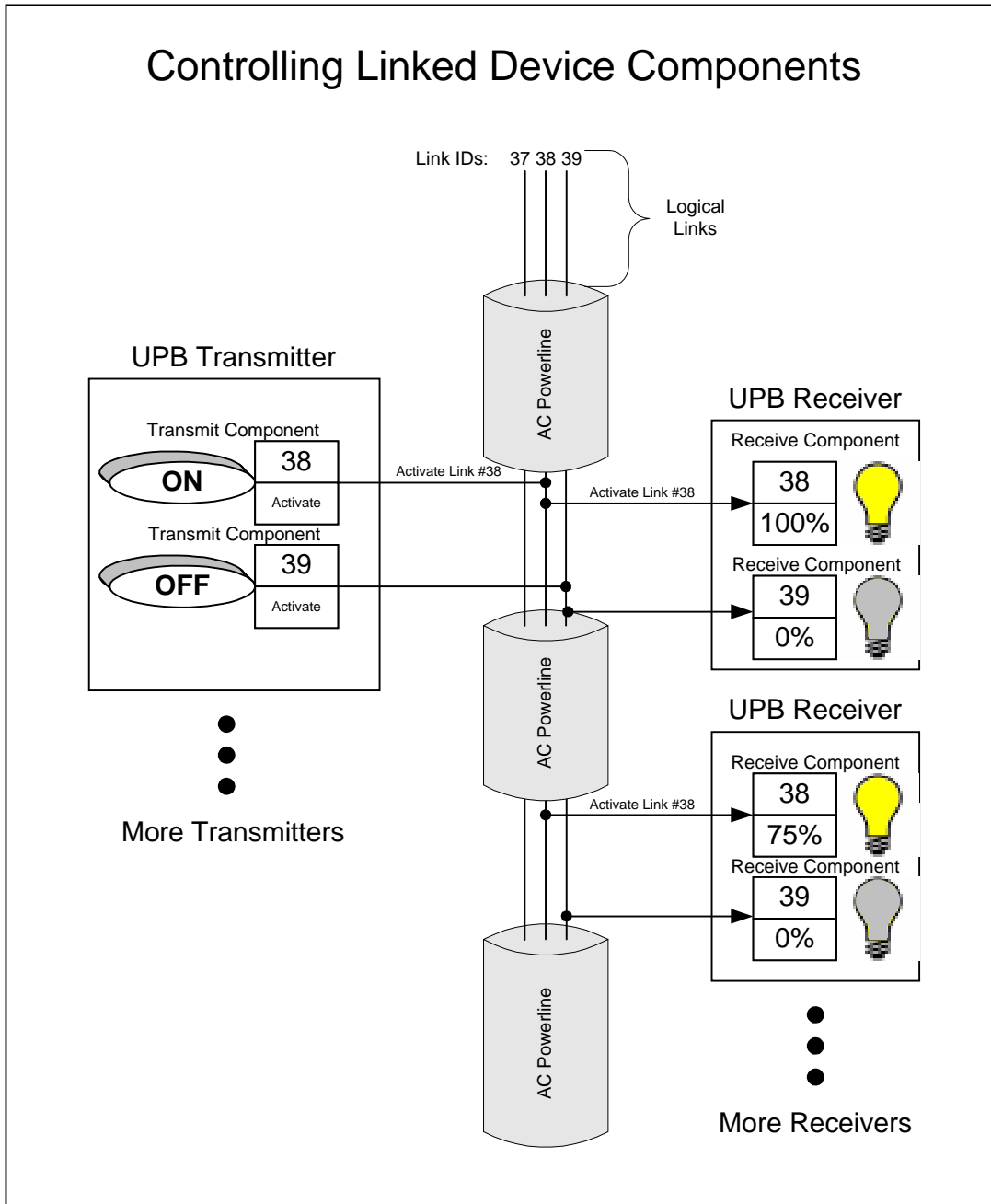


Figure 12: UPB Device Component Linking Example

### 8.4. Reading/Writing Setup Registers

Up to 256 Setup Registers can be read, written, and modified over the powerline by using special UPB Messages contained in the UPB Core Command Message Set (see Section 10 for more details). Every existing Setup Register (except the Network Password) can be read over the powerline at any time. Setup Registers can only be written, however, when Setup Register Write Protection is disabled.

### **8.4.1. Setup Register Write Protection**

In the UPB System Model, all Setup Registers are write-protected by default. While write protection is enabled, the UPB device will reject any attempt to write its Setup Registers by way of the powerline. Each UPB device shall have at least two different methods for disabling write protection: one is by receiving a special password-protected UPB Message and the other is by getting the device in a special mode of operation known as the SETUP Mode. Section 8.5.2 describes how to get a UPB device into and out of the SETUP Mode.

Write protection shall only be disabled on a temporary basis. Once Setup Register Write Protection is disabled, the UPB device shall start an internal timer that will re-enable write protection after five minutes.

### **8.4.2. Reading The Network Password**

Because of its special importance, the Network Password cannot be read unless the device is in the SETUP Mode. Attempts to read the Network Password while not in SETUP Mode shall return the value of 0x0000. This read protection helps to maintain a level of security from someone accidentally (or maliciously) modifying Setup Registers when they are unauthorized to do so. Section 8.5.2 describes how to get a UPB device into and out of the SETUP Mode.

## **8.5. Modes Of Operation**

UPB devices shall implement at least two different modes of operation: Normal Mode and SETUP Mode as described below.

### **8.5.1. Normal Mode**

The Normal Mode is a UPB device's normal operational mode.

### **8.5.2. SETUP Mode**

All UPB devices in the UPB System Model shall implement a special mode of operation, known as SETUP Mode. While in SETUP Mode, Setup Register Write Protection is temporarily disabled allowing Setup Registers to be written using special write/modify UPB Messages contained in the UPB Core Command Message Set (see Section 10 for more details). Also, while in the SETUP Mode, the contents of the Network Password registers can be read over the powerline. This is the only method for accessing the Network Password of a UPB device. In addition, the device automatically gets the benefit of having one additional Unit ID (UID #254), called the Setup ID that it can use to accept Direct UPB Communication Packets. The Setup ID can be very helpful in identifying a single UPB device that's Unit ID is unknown.

### **8.5.2.1. Starting SETUP Mode**

Each UPB device shall have at least two different methods for starting SETUP Mode: one is remotely by receiving a special password protected UPB Message and the other is physical - by “touching” the device (usually by some sort of pushbutton press sequence). The UPB System Model does not specify the exact physical method for each device to enter SETUP Mode but it is recommended that it be made in such a way that it is not easily activated by accident. The UPB System Model also recommends that the device manufacturer design a special indication (such as blinking an LED) that identifies to a user that the device is in SETUP Mode.

### **8.5.2.2. Stopping SETUP Mode**

Each device shall automatically time-out of SETUP Mode (and return to Normal Mode) after five minutes of not receiving a UPB Message addressed to it. In addition, a special UPB Message is defined for stopping the SETUP Mode. The device manufacturer can also develop some physical method for stopping the SETUP Mode on their device. When SETUP Mode expires, Setup Register Write Protection is automatically enabled to avoid accidentally keeping a device write-enabled indefinitely.

## **8.6. Device Command and Status Registers**

Each device in the UPB System Model shall have an 8-bit Device Command Register and an 8-bit Device Status Register. Special UPB Messages are available in the Core Message Set to write the Device Command Register and to read the Device Status Register. It is left up to the individual device manufacturer as to what these two registers are used for on their device.

## 9. The UPB Message

The UPB Message is the actual payload of any UPB Communications Packet (Figure 13). It is a variable length field of the UPB Communications Packet that can range from 0 to 18 bytes. The first byte of the UPB Message is referred to as the Message Data ID (MDID). The optional remaining bytes of the UPB Message are known collectively as the Message Data Arguments (MDA).

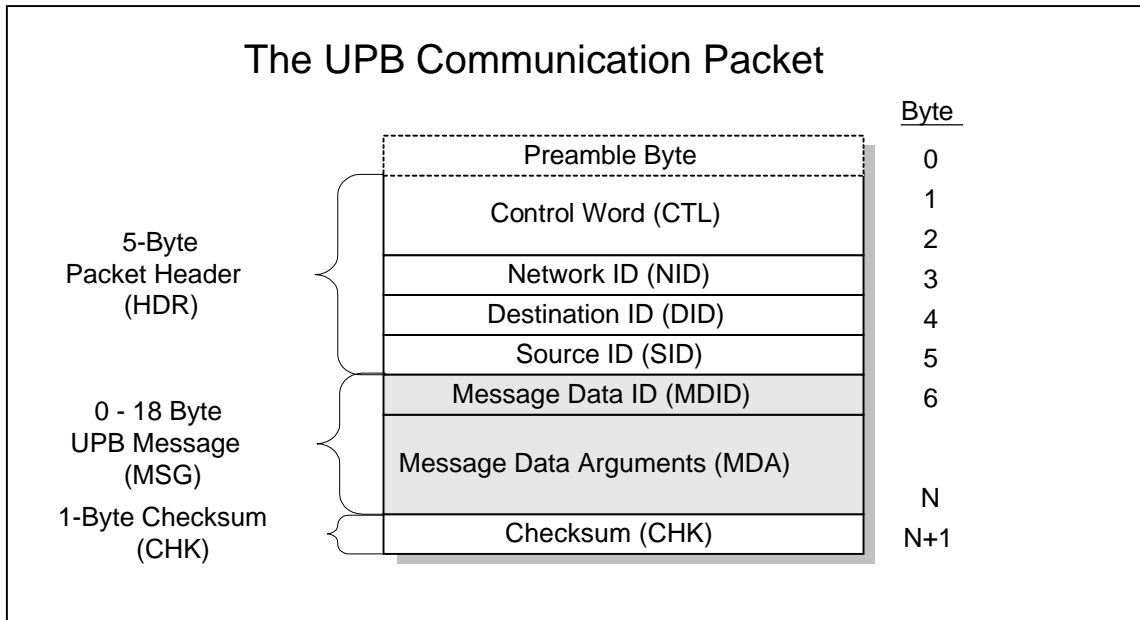


Figure 13: The UPB Message (MSG)

In order to maintain device interoperability, the UPB Message field should only be interpreted/encoded in the format described in this document.

### 9.1. The Message Data ID (MDID) Field

The first byte of any UPB Message is referred to as the Message Data ID (MDID). The MDID identifies the message. It is broken up into two separate fields as shown in Figure 14. Many UPB Messages are made up of only the MDID. Other messages require additional argument bytes that are located in the Message Data Argument field.

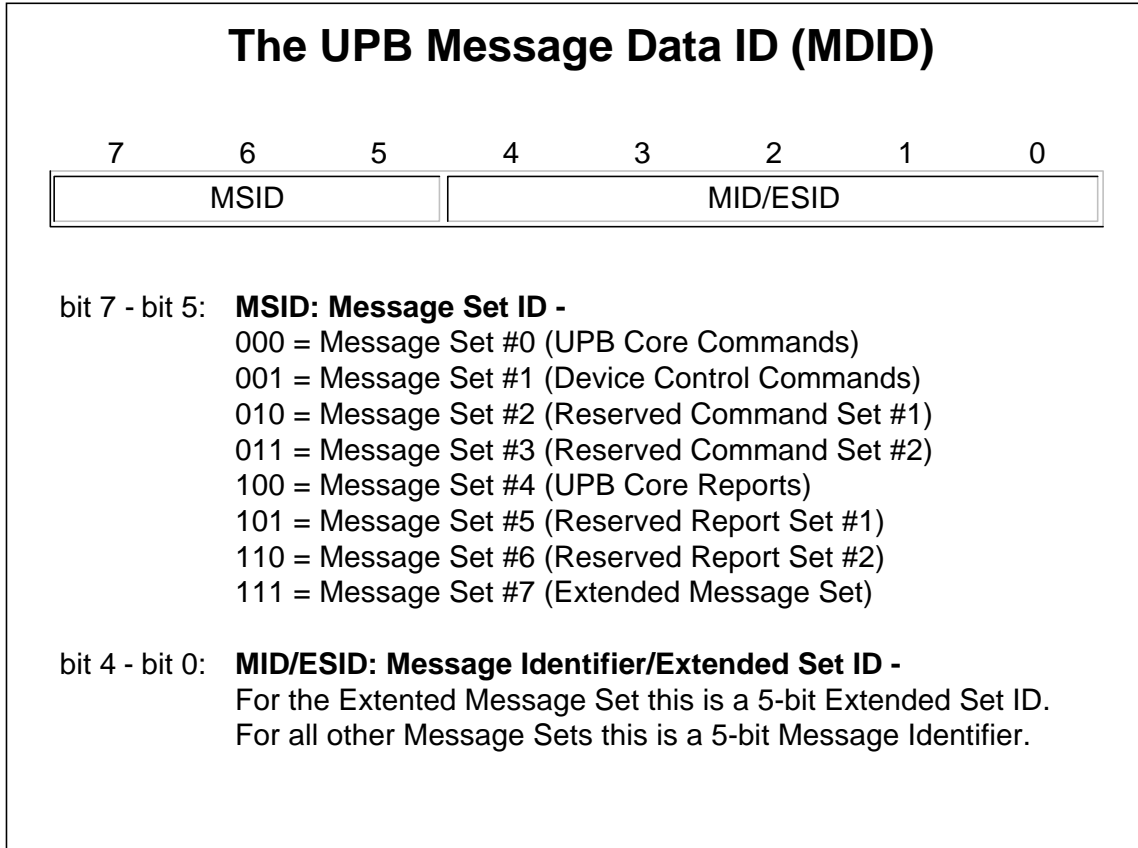


Figure 14: The Message Data ID (MDID)

### 9.1.1. The Message Set ID Field (MSID)

In order to keep the UPB Message format as flexible and expandable as possible, eight separate Message Sets have been defined (see Table 8). Each Message Set is described in detail in a separate section of this document. The Message Set ID (MSID) is a three-bit field of the MDID used to identify which one of eight Message Sets the message belongs to.

MSID	Message Set	Description
000	UPB Core Command Set	A set of command messages that is common to all UPB devices.
001	Device Control Command Set	A set of command messages that is used to control most UPB devices.
010	Reserved Command Set #1	A set of command messages that is reserved at this time for future use.
011	Reserved Command Set #2	A set of command messages that is reserved at this time for future use.
100	UPB Core Report Set	A set of report messages that is common to all UPB devices.
101	Reserved Report Set #1	A set of report messages that is reserved at this time for future use.
110	Reserved Report Set #2	A set of report messages that is reserved at this time for future use.
111	Extended Message Set	A set of messages that is open for device vendors to register for and define.

Table 8: The UPB Message Sets

### 9.1.2. The Message Identifier (MID)

The Message Identifier (MID) is a 5-bit field of the MDID used to identify the particular command or report. The encoding of this field is unique to each Message Set. Refer to the individual sections on each Message Set for more details on how this field is encoded.

### 9.1.3. Extended Set ID

When the MSID field indicates the Extended Message Set (111), bits 4 – 0 of the MDID are used as a 5-bit Extended Set ID (ESID). The Extended Set ID is used to identify which one of 32 possible Extended Message Sets to use to interpret the remainder of the message. Device manufacturers that have special needs that the other Message Sets don't address can define their own Extended Message Set to handle their needs. If a device manufacturer wants to establish their own Extended Message Set they must first contact Powerline Control Systems to be issued a unique Extended Set ID. This Extended Set ID value should be placed in all messages using this new Extended Message Set.

## 9.2. Message Data Arguments (MDA) Field

The remaining bytes of a UPB Message are known collectively as the Message Data Arguments. A UPB Message can have up to 17 bytes of Message Data Arguments.



## 10. The UPB Core Command Set

The UPB Core Command Set (Table 9 below) is a group of UPB Command Messages that are common to all UPB devices. This Message Set is identified by the MSID value of 000 and it covers the command MDID range of 0x00 to 0x1F.

MDID (Hex)	Command Name	Description
0x00	Null Command	Performs no action.
0x01	Write Enable Command	Commands the receiving device to disable its global write protection of its Setup Registers.
0x02	Write Protect Command	Commands the receiving device to enable its global write protection of its Setup Registers.
0x03	Start SETUP Mode Command	Puts the receiving device into the SETUP Mode.
0x04	Stop SETUP Mode Command	Takes the receiving device out of the SETUP Mode.
0x05	Get SETUP Time Command	Requests the value of the SETUP Mode Timer of the receiving device to be returned in a Setup Time Report.
0x06	Auto-Address Command	Commands the receiving device to select and set itself to a new random Unit ID (between 1 and 250).
0x07	Get Device Status Command	Requests the value of the Device Status Register of the receiving device to be returned in a Device Status Report.
0x08	Set Device Control Command	Sets the Device Control Register to a given value.
0x09	Get Signal Strength Command	Requests the Signal Strength value of the receiving device to be returned in a Signal Strength Report.
0x0A	Get Noise Level Command	Requests the Noise Level value of the receiving device to be returned in a Noise Level Report.
0x0B	Add Link Command	Changes the first Receive Component's "Unused" (0xFF) Link ID to the specified Link ID.

MDID (Hex)	Command Name	Description
0x0C	Delete Link Command	Changes the Receive Component's specified Link ID to an "Unused" (0xFF) Link ID.
0x0D	Transmit This Message Command	Requests that the receiving device transmit the UPB Message contained in the MDA field.
0x10	Get Register Value Command	Requests the values of one or more of the receiving device's Setup Registers to be returned in a Register Values Report.
0x11	Set Register Value Command	Sets one or more Setup Registers in the receiving device to a given list of values.
0x12	Copy Register Value Command	Copies the value of a specified Setup Register into another specified Setup Register.
0x13	Swap Register Nibbles Command	Swaps the MS Nibble with the LS Nibble of a specified Setup Register.
0x14	Increment Register Value Command	Increments the value in a specified Setup Register by 1.
0x15	Decrement Register Value Command	Decrements the value in a specified Setup Register by 1.
0x16	Left Shift Register Command	Shift the value in a specified Setup Register 1-bit position to the left.
0x17	Right Shift Register Command	Shift the value in a specified Setup Register 1-bit position to the right.
0x18	Set Register Bits Command	Set specified bits in a specified Setup Register.
0x19	Clear Register Bits Command	Clear specified bits in a specified Setup Register.
0x1A	Toggle Register Bits Command	Toggle specified bits in a specified Setup Register.
0x0E – 0x0F	Unused	Reserved for future command use.
0x1B – 0x1F	Unused	Reserved for future command use.

Table 9: The UPB Core Command Set

## 10.1. UPB Core Commands

### 10.1.1. The “Null” Command

The “Null” Command does not perform any action on the receiving device. It can be used to simply get Acknowledgement responses back from an addressed UPB device to know that it exists. The “Null” Command has a Message Data ID of **0x00** and has the following syntax:

MDID
0x00

### 10.1.2. The “Write Enable” Command

The “Write Enable” Command is used to disable the global write protection of a device’s Setup Registers so that they may be written into. For extra security, this message must include the assigned Network Password of the addressed device. The “Write Enable” Command has a Message Data ID of **0x01** and has the following syntax:

MDID	Arg1	Arg2
0x01	PM	PL

Where PM & PL are the 16-bit Network Password (0x0000 – 0xFFFF).

When the addressed UPB device receives this command, and validates the Network Password, it will disable its global write protection. If the Network Password in this message fails to match the assigned Network Password of the receiving device then this message is ignored.

### 10.1.3. The “Write Protect” Command

The “Write Protect” Command is used to enable global write protection of a device’s Setup Registers. The “Write Protect” Command has a Message Data ID of **0x02** and has the following syntax:

MDID
0x02

When the addressed UPB device receives this command it will write protect its Setup Registers (the default state).

#### 10.1.4. The “Start SETUP Mode” Command

The “Start SETUP Mode” Command is used to put the addressed device into the SETUP Mode. For extra security, this message must include the Network Password of the addressed device. The “Start SETUP Mode” Command has a Message Data ID of **0x03** and has the following syntax:

MDID	Arg1	Arg2
0x03	PM	PL

Where PM & PL are the assigned 16-bit Network Password (0x0000 – 0xFFFF).

When the addressed UPB device receives this command, and validates the Network Password, it will go into the SETUP Mode for the next 5 minutes. If the Network Password in this message fails to match the assigned Network Password of the receiving device then this message is ignored.

#### 10.1.5. The “Stop SETUP Mode” Command

The “Stop SETUP Mode” Command is used to take the addressed device out of the SETUP Mode and return it to the Normal Mode. The “Stop SETUP Mode” Command has a Message Data ID of **0x04** and has the following syntax:

MDID
0x04

When the addressed UPB device receives this command it will return to Normal Mode and write protect its Setup Registers.

#### 10.1.6. The “Get SETUP Time” Command

The “Get SETUP Time” Command is used to retrieve the amount of time the addressed device has before it times out of the SETUP Mode. When a device enters the SETUP Mode it will automatically time out of it in five minutes. This command is intended primarily for device setup and test operations. The “Get SETUP Time” Command has a Message Data ID of **0x05** and has the following syntax:

MDID
0x05

When the addressed UPB device receives this command, it will respond with a Setup Time Report message (0x85) that contains the current value in the device’s Setup Mode Timer.

### 10.1.7. The “Auto-Address” Command

The “Auto-Address” Command is used to change the Unit ID of the addressed device to a random value. This command is intended to be used for detecting Duplicate IDs as described in Section 18.6. Note that the addressed device must first have write protection disabled in order for this command to take effect. The “Auto-Address” Command has a Message Data ID of **0x06** and has the following syntax:

MDID
0x06

When the addressed UPB device receives this command it will set its Unit ID to a random value between 1 and 250.

**Note: The addressed device must first have global write protection disabled in order for this command to take effect.**

### 10.1.8. The “Get Device Status” Command

The “Get Device Status” Command is used to retrieve the contents of the 8-bit Device Status Register. The “Get Device Status” Command has a Message Data ID of **0x07** and has the following syntax:

MDID
0x07

When the addressed UPB device receives this command, it will respond with a Device Status Report (0x87) that contains the 1-byte contents of the Device Status Register.

### 10.1.9. The “Set Device Control” Command

The “Set Device Control” Command is used to set the 8-bit Device Control Register to a specified value. The “Set Device Control” Command has a Message Data ID of **0x08** and has the following syntax:

MDID	Arg1
0x08	VV

Where VV is the value (0x00 – 0xFF) to be written into the Device Control Register.

When the addressed UPB device receives this command, it will set its Device Control Register to the value specified in the 1<sup>st</sup> argument of the command.

#### 10.1.10. The “Get Signal Strength” Command

The “Get Signal Strength” Command is used to retrieve the contents of the 16-bit Signal Strength Register from the receiving device. The “Get Signal Strength” Command has a Message Data ID of **0x09** and has the following syntax:

MDID
0x09

When the addressed UPB device receives this command, it will respond with a Signal Strength Report (0x89) that contains the 16-bit contents of the Signal Strength Register.

#### 10.1.11. The “Get Noise Level” Command

The “Get Noise Level” Command is used to retrieve the contents of the 16-bit Noise Level Register from the receiving device. The “Get Noise Level” Command has a Message Data ID of **0x0A** and has the following syntax:

MDID
0x0A

When the addressed UPB device receives this command, it will respond with a Noise Level Report (0x8A) that contains the 16-bit contents of the Noise Level Register.

#### 10.1.12. The “Link Component” Command

TBD

#### 10.1.13. The “Unlink Component” Command

TBD

#### 10.1.14. The “Transmit This Message” Command

The “Transmit This Message” Command is used to request that the receiving device transmit the valid UPB Message that is contained in the Message Data Argument field. The “Transmit This Message” Command has a Message Data ID of **0x0D** and has the following syntax:

MDID	Arg1	Arg2	Arg3	Arg4	Arg5	Arg6	Arg7	Arg8
0x0D	CB1	CB2	NID	DID	SID	VV	[VV]	[VV]
Arg9	Arg10	Arg11	Arg12	Arg13	Arg14	Arg15	Arg16	Arg17
[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]

Where CB1, CB2, NID, DID, and SID are the five bytes of the Packet Header and VV is the rest of the UPB Message and Checksum to be transmitted by the receiving device.

### 10.1.15. The “Get Register Values” Command

The “Get Register Values” Command is used to retrieve the contents of one or more Setup Registers from the addressed device. This command is intended primarily for device setup and test operations. The “Get Register Values” Command has a Message Data ID of **0x10** and has the following syntax:

MDID	Arg1	Arg2
0x10	RR	NN

Where RR is the starting register number (0x00 – 0xFF) to be retrieved and NN is the number of registers to be retrieved (0x01 – 0x10).

When the addressed UPB device receives this command, it will respond with a Register Values Report message that includes the contents of the specified Setup Register(s) (see Section 0).

### 10.1.16. The “Set Register Values” Command

The “Set Register Values” Command is used to write a list of (up to 16) values into a block of Setup Registers of the addressed device. This command is intended primarily for device setup and test operations. Note that the addressed device must first have write protection disabled in order for this command to take effect. The “Set Register Values” Command has a Message Data ID of **0x11** and has the following syntax:

MDID	Arg1	Arg2	Arg3	Arg4	Arg5	Arg6	Arg7	Arg8
0x11	RR	VV	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]
Arg9	Arg10	Arg11	Arg12	Arg13	Arg14	Arg15	Arg16	Arg17
[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]

Where RR is the starting register number to be written (0x00 – 0xFF), VV is the value to set that register to, and [VV] is an optional list of up to 15 more values (0x00 – 0xFF) to set the next consecutive register(s) to.

When the addressed UPB device receives this command, it will write the values in the given list into its Setup Registers starting at the register specified in the command.

**Note: The addressed device must first have global write protection disabled in order for this command to take effect.**

### 10.1.17. The “Copy Register Value” Command

The “Copy Register Value” Command is used to copy the contents of one Source Setup Register into one Destination Setup Register in the addressed device. This command is intended primarily for device setup and test operations. Note that the addressed device must first have write protection disabled in order for this command to take effect. The “Copy Register Value” Command has a Message Data ID of **0x12** and has the following syntax:

MDID	Arg1	Arg2
0x12	SR	DR

Where SR is the source register number (0x00 – 0xFF) to be copied from and DR is the destination register number (0x00 – 0xFF) to be written.

When the addressed UPB device receives this command, it will copy the contents of the Source Setup Register into the Destination Register.

**Note: The addressed device must first have global write protection disabled in order for this command to take effect.**

### 10.1.18. The “Swap Register Nibbles” Command

The “Swap Register Nibbles” Command is used to swap the upper and lower nibbles of a Setup Register in the addressed device. This command is intended primarily for device setup and test operations. Note that the addressed device must first have write protection disabled in order for this command to take effect. The “Swap Register Nibbles” Command has a Message Data ID of **0x13** and has the following syntax:

MDID	Arg1
0x13	RR

Where RR is the specified Setup Register number (0x00 – 0xFF) to be modified.

When the addressed UPB device receives this command, it will swap the upper nibble of the specified Setup Register with its lower nibble.



**Note: The addressed device must first have global write protection disabled in order for this command to take effect.**

### 10.1.19. The “Increment Register Value” Command

The “Increment Register Value” Command is used to increment the value of a Setup Register in the addressed device by a specified Step Size. This command is intended primarily for device setup and test operations. Note that the addressed device must first have write protection disabled in order for this command to take effect. The “Increment Register Value” Command has a Message Data ID of **0x14** and has the following syntax:

MDID	Arg1	Arg2
0x14	RR	SS

Where RR is the specified Setup Register number (0x00 – 0xFF) to be incremented and SS is the Step Size (0x00 – 0xFF).

When the addressed UPB device receives this command, it will increment the value of the specified Setup Register by the specified Step Size. The Setup Register will not be allowed to rollover above a value of 0xFF.

**Note: The addressed device must first have global write protection disabled in order for this command to take effect.**

### 10.1.20. The “Decrement Register Value” Command

The “Decrement Register Value” Command is used to decrement the value of a Setup Register in the addressed device by a specified Step Size. This command is intended primarily for device setup and test operations. Note that the addressed device must first have write protection disabled in order for this command to take effect. The “Decrement Register Value” Command has a Message Data ID of **0x15** and has the following syntax:

MDID	Arg1	Arg2
0x15	RR	SS

Where RR is the specified Setup Register number (0x00 – 0xFF) to be decremented and SS is the Step Size (0x00 – 0xFF).

When the addressed UPB device receives this command, it will decrement the value of the specified Setup Register by the specified Step Size. The Setup Register will not be allowed to rollover below a value of 0x00.

**Note: The addressed device must first have global write protection disabled in order for this command to take effect.**

### 10.1.21. The “Left Shift Register” Command

The “Left Shift Register” Command is used to shift the contents of a Setup Register one bit position to the left. This command is intended primarily for device setup and test operations. Note that the addressed device must first have write protection disabled in order for this command to take effect. The “Left Shift Register” Command has a Message Data ID of **0x16** and has the following syntax:

MDID	Arg1
0x16	RR

Where RR is the specified Setup Register number (0x00 – 0xFF) to be shifted.

When the addressed UPB device receives this command, it will shift the contents of the specified Setup Register one bit position to the left. The contents of bit #0 will be filled with a 0.

**Note: The addressed device must first have global write protection disabled in order for this command to take effect.**

### 10.1.22. The “Right Shift Register” Command

The “Right Shift Register” Command is used to shift the contents of a Setup Register one bit position to the right. This command is intended primarily for device setup and test operations. Note that the addressed device must first have write protection disabled in order for this command to take effect. The “Right Shift Register” Command has a Message Data ID of **0x17** and has the following syntax:

MDID	Arg1
0x17	RR

Where RR is the specified Setup Register number (0x00 – 0xFF) to be shifted.

When the addressed UPB device receives this command, it will shift the contents of the specified Setup Register one bit position to the right. The contents of bit #7 will be filled with a 0.

**Note: The addressed device must first have global write protection disabled in order for this command to take effect.**

### 10.1.23. The “Set Register Bits” Command

The “Set Register Bits” Command is used to set specified bits of a Setup Register in the addressed device. This command is intended primarily for device setup and test operations. Note that the addressed device must first have write protection disabled in order for this command to take effect. The “Set Register Bits” Command has a Message Data ID of **0x18** and has the following syntax:

MDID	Arg1	Arg2
0x18	RR	BM

Where RR is the specified Setup Register number (0x00 – 0xFF) to be modified and BM is the Bit Mask (0x00 – 0xFF).

When the addressed UPB device receives this command, it will logically OR the Bit Mask value with the contents of the specified Setup Register.

**Note: The addressed device must first have global write protection disabled in order for this command to take effect.**

### 10.1.24. The “Clear Register Bits” Command

The “Clear Register Bits” Command is used to clear specified bits of a Setup Register in the addressed device. This command is intended primarily for device setup and test operations. Note that the addressed device must first have write protection disabled in order for this command to take effect. The “Set Register Bits” Command has a Message Data ID of **0x19** and has the following syntax:

MDID	Arg1	Arg2
0x19	RR	BM

Where RR is the specified Setup Register number (0x00 – 0xFF) to be modified and BM is the Bit Mask (0x00 – 0xFF).

When the addressed UPB device receives this command, it will logically AND the Bit Mask value with the contents of the specified Setup Register.

**Note: The addressed device must first have global write protection disabled in order for this command to take effect.**

### 10.1.25. The “Toggle Register Bits” Command

The “Toggle Register Bits” Command is used to toggle specified bits of a Setup Register in the addressed device. This command is intended primarily for device setup and test operations. Note that the addressed device must first have write

protection disabled in order for this command to take effect. The “Toggle Register Bits” Command has a Message Data ID of **0x1A** and has the following syntax:

MDID	Arg1	Arg2
0x1A	RR	BM

Where RR is the specified Setup Register number (0x00 – 0xFF) to be modified and BM is the Bit Mask (0x00 – 0xFF).

When the addressed UPB device receives this command it will logically XOR the Bit Mask value with the contents of the specified Setup Register.

**Note: The addressed device must first have global write protection disabled in order for this command to take effect.**

## 11. The Device Control Command Set

The Device Control Command Set (Table 10 below) is a group of UPB Command Messages designed to perform the most common device control functions. This Message Set is identified by the MSID value of 001 and it occupies the MDID range of 0x20 to 0x3F.

MDID (Hex)	Command/Report Name	Command Description
0x20	Activate Link Command	Requests the receiving device(s) to activate their linked Receive Components.
0x21	Deactivate Link Command	Requests the receiving device(s) to deactivate their linked Receive Components.
0x22	Goto Command	Requests the receiving device(s) to go to a specified Level at a specified Rate.
0x23	Start Fade Command	Requests the receiving dimming-type device(s) to go to a specified Level at a specified Rate.
0x24	Stop Fade Command	Requests the receiving dimming-type device(s) to stop fading and stay at their current level.
0x25	Blink Command	Requests the receiving device to blink on and off their output at a specified rate.
0x30	Report State Command	Requests the receiving device to report its current state information in a Device State Report.
0x31	Store State Command	Requests the receiving device(s) to store its linked components current state.
0x26 – 0x2F	Unused	Reserved for future use.
0x32 – 0x3F	Unused	Reserved for future use.

Table 10: The Device Control Message Set

## 11.1. Device Control Commands

### 11.1.1. The Activate Link Command

The “Activate Link” Command is used to request the receiving device(s) to activate their “linked” Receive Components. The “Activate Link” Command must be packaged in a Link Packet for it to have any meaning. The “Activate Link” Command has the Message Data ID of **0x20** and has the following syntax:

MDID
0x20

When the addressed UPB device receives this command, it will “activate” the Receive Component(s) that are associated with the Link ID that was made active by this command.

The term “activate” is a general term that usually means to turn to the ON state but each individual device manufacturer can define it differently for their device. To some devices the term “activate” might mean to turn ON a lighting channel to a preset level (e.g. 50%). To other devices the term “activate” might mean to blink an indicator every second. The parameters in the activated Receive Component Record can be used to configure what to do to the associated physical component when it is “activated”.

### 11.1.2. The Deactivate Link Command

The “Deactivate Link” Command is used to request the receiving device(s) to “deactivate” their linked Receive Components. The “Deactivate Link” Command must be packaged in a Link Packet for it to have any meaning. The “Deactivate Link” Command has the Message Data ID of **0x21** and has the following syntax:

MDID
0x21

When the addressed UPB device receives this command, it will “deactivate” the Receive Component(s) that are associated with the Link ID that was made active by this command.

The term “deactivate” is a general term that usually means to turn to the OFF state but each individual device manufacturer can define it differently for their device. To some devices the term “deactivate” might mean to turn a lighting channel to 0%. To other devices the term “deactivate” might mean to set an indicator to red. The parameters in the deactivated Receive Component Record

can be used to configure what to do to the associated physical component when it is “deactivated”.

### 11.1.3. The Goto Command

The “Goto” Command is used to request a receiving device or a group of “linked” Device Components to go to a specified Level at a specified Rate. The “Goto” Command has the Message Data ID of **0x22** and has two different syntaxes based on how it is packaged. The “Goto” Command can either be packaged in a Link Packet to affect a group of “linked” Device Components or it can be packaged in a Direct Packet to affect an individual device.

The message syntax for the “Goto” Command packaged in a Link Packet is as follows:

MDID	Arg1	Arg2
0x22	LL	[RR]

Where LL is the specified Level (0x00 – 0x64) and [RR] is the optional specified Rate (0x00 – 0xFF). When the addressed UPB devices receive this command, they shall set their “linked” Receive Components to the specified Level at the specified Rate. If the Rate is not specified in the command then each device’s Default Rate shall be used instead.

The message syntax for the “Goto” Command packaged in a Direct Packet is designed to account for multiple-channel devices and is as follows:

MDID	Arg1	Arg2	Arg3
0x22	LL	[RR]	[CC]

Where LL is the specified Level (0x00 – 0x64) and [RR] is the optional specified Rate (0x00 – 0xFF) and [CC] is the optional specified Channel (0x00 – 0xFF). When the addressed UPB device receives this command, it shall set the specified Channel to the specified Level at the specified Rate. If the Channel is not specified in the command then all of the device’s channels shall be affected. If the Rate is not specified in the command then each device’s Default Rate shall be used instead.

### 11.1.4. The Fade Start Command

The “Fade Start” Command is used to request the receiving dimming-type device or a group of “linked” Device Components on dimming-type devices to fade to a specified Level at a specified Rate. Non-dimming type devices shall ignore the “Fade Start” Command. The “Fade Start” Command has the Message Data ID of **0x23** and has two different syntaxes based on how it is packaged. The “Fade

Start” Command can either be packaged in a Link Packet to affect a group of “linked” Device Components or it can be packaged in a Direct Packet to affect an individual device.

The message syntax for the “Fade Start” Command packaged in a Link Packet is as follows:

MDID	Arg1	Arg2
0x23	LL	[RR]

Where LL is the specified Level (0x00 – 0x64) and [RR] is the optional specified Rate (0x00 – 0xFF). When the addressed UPB devices receive this command, they shall set their “linked” Receive Components to the specified Level at the specified Rate. If the Rate is not specified in the command then each device’s Default Rate shall be used instead.

The message syntax for the “Fade Start” Command packaged in a Direct Packet is designed to account for multiple-channel devices and is as follows:

MDID	Arg1	Arg2	Arg3
0x23	LL	[RR]	[CC]

Where LL is the specified Level (0x00 – 0x64) and [RR] is the optional specified Rate (0x00 – 0xFF) and [CC] is the optional specified Channel (0x00 – 0xFF). When the addressed UPB device receives this command, it shall set the specified Channel to the specified Level at the specified Rate. If the Channel is not specified in the command then all of the device’s channels shall be affected. If the Rate is not specified in the command then each device’s Default Rate shall be used instead.

Note: The only difference between the Fade Start Command and the Goto Command is that non-dimming type devices shall ignore the Fade Start Command.

### 11.1.5. The Fade Stop Command

The “Fade Stop” Command is used to request the receiving dimming-type device or a group of “linked” Device Components on dimming-type devices to stop fading and stay at their current Level. The “Fade Stop” Command can either be packaged in a Link Packet to affect a group of “linked” Device Components or it can be packaged in a Direct Packet to affect an individual device. The “Fade Stop” Command has the Message Data ID of **0x24** and has the following syntax:



MDID
0x24

When the addressed UPB device receives this command, it shall stop any fading operation that is currently going on.

### 11.1.6. The Blink Command

The “Blink” Command is used to request the receiving device or a group of “linked” Device Components to blink on and off their output at a specified rate. The “Blink” Command has the Message Data ID of **0x25** and has two different syntaxes based on how it is packaged. The “Blink” Command can either be packaged in a Link Packet to affect a group of “linked” Device Components or it can be packaged in a Direct Packet to affect an individual device.

The message syntax for the “Blink” Command packaged in a Link Packet is as follows:

MDID	Arg1
0x25	RR

Where RR is the specified Blink Rate (0x00 – 0xFF). When the addressed UPB devices receive this command, they shall blink their “linked” Receive Components at the specified Rate.

The message syntax for the “Blink” Command packaged in a Direct Packet is designed to account for multiple-channel devices and is as follows:

MDID	Arg1	Arg2
0x25	RR	[CC]

Where RR is the specified Blink Rate (0x00 – 0xFF) and [CC] is the optional specified Channel (0x00 – 0xFF). When the addressed UPB device receives this command, it shall blink the specified Channel at the specified Rate. If the Channel is not specified in the command then all of the device’s channels shall be affected.

### 11.1.7. The Report State Command

The “Report State” Command is used to request the receiving device to report its current state information in a Device State Report. The “Report State” Command must be packaged in a Direct Packet for it to have any meaning. The “Report State” Command has the Message Data ID of **0x30** and has the following syntax:

MDID

0x30

When the addressed UPB device receives this command, it shall report its current state information in a Device State Report.

### 11.1.8. The Store State Command

The “Store State” Command is used to request the receiving device(s) to save the current state of its linked components. The “Store State” Command must be packaged in a Link Packet for it to have any meaning.

The “Store State” Command has the Message Data ID of **0x31** and has the following syntax:

MDID

0x31

When the addressed UPB devices receive this command, they shall store the current state of their “linked” Receive Components. Note: This command was intended for lighting control devices to be able to capture their current light levels into Receive Components and then later have those Receive Components “activated” to produce a lighting “Scene”.

## 12. The Reserved Command Set #1

The Reserved Command Set #1 is just that: reserved (for future use). There is currently no command messages defined for this Message Set, it is simply a placeholder for the future. This Message Set occupies the MDID range of 0x40 to 0x5F.

## 13. The Reserved Command Set #2

The Reserved Command Set #2 is just that: reserved (for future use). There is currently no command messages defined for this Message Set, it is simply a placeholder for the future. This Message Set occupies the MDID range of 0x60 to 0x7F.

## 14. The UPB Core Report Set

The UPB Core Report Set (Table 11) is a group of UPB Report Messages that are common to all UPB devices. This Message Set is identified by the MSID value of 100 and it covers the MDID range of 0x80 to 0x9F.

MDID (Hex)	Report Name	Description
0x80 – 0x84	Unused	Reserved for future use.
0x85	Setup Time Report	Returns the current value that is in the device's Setup Mode timer.
0x86	Device State Report	Returns the current state of the device.
0x87	Device Status Report	Returns the current value that is in the device's Device Status Register.
0x89	Signal Strength Report	Returns the current value that is in the device's Signal Strength Register.
0x8A	Noise Level Report	Returns the current value that is in the device's Noise Level Register.
0x8B – 0x8F	Unused	Reserved for future use.
0x90	Register Values Report	Returns the values of one or more of the device's Setup Registers (EEPROM).
0x91	RAM Values Report	Returns the values of one or more of the device's SRAM Registers
0x92	Raw Data Report	Reports from 1 to 16 raw data values.

MDID (Hex)	Report Name	Description
0x93	Heartbeat Report	Reports no data. It just lets the network know that the device is alive.
0x94 – 0xAE	Unused	Reserved for future report use.
0xAF	Acknowledgement Report	Reports that the previous UPB Communication Packet was accepted.

Table 11: The UPB Core Report Set

## 14.1. UPB Core Reports

### 14.1.1. The Setup Time Report

The “Setup Time” Report is used to return the current value of the device’s Setup Mode Timer that indicates the number of ?? millisecond ticks that are left before timing out of the SETUP Mode. The “Setup Time” Report has a Message Data ID of **0x85** and has the following syntax:

MDID	Arg1	Arg2
0x85	RR	TT

Where RR is the RAM Register Number of the Setup Mode Timer (this will vary from device to device) and TT is the number of ?? millisecond ticks that are left before timing out of the SETUP Mode.

### 14.1.2. The Device State Report

The “Device State” Report is used to return the current state information about the receiving device. The “Device State” Report has a Message Data ID of **0x86** and has the following syntax:

MDID	Arg1	Arg2	Arg3	Arg4	Arg5	Arg6	Arg7	Arg8
0x86	VV	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]
Arg9	Arg10	Arg11	Arg12	Arg13	Arg14	Arg15	Arg16	Arg17
[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]

Where VV is a list of up to 17 retrieved state information values (0x00 – 0xFF).

### 14.1.3. The Device Status Report

The “Device Status” Report is used to return the current value in the Device Status Register. The “Device Status” Report has a Message Data ID of **0x87** and has the following syntax:

MDID	Arg1	Arg2
0x87	RR	VV

Where RR is the RAM Register Number of the Device Status Register (this will vary from device to device) and VV is the 8-bit value of the Device Status Register.

### 14.1.4. The Signal Strength Report

The “Signal Strength” Report is used to return the current value in the device’s 16-bit Signal Strength Register. The “Signal Strength” Report has a Message Data ID of **0x89** and has the following syntax:

MDID	Arg1	Arg2	Arg3
0x89	RR	SM	SL

Where RR is the RAM Register Number of the Signal Strength Register (this will vary from device to device) and SM & SL are the 16-bit value of the Signal Strength Register.

### 14.1.5. The Noise Level Report

The “Noise Level” Report is used to return the current value in the device’s 16-bit Signal Strength Register. The “Noise Level” Report has a Message Data ID of **0x8A** and has the following syntax:

MDID	Arg1	Arg2	Arg3
0x8A	RR	NM	NL

Where RR is the RAM Register Number of the Noise Level Register (this will vary from device to device) and NM & NL are the 16-bit value of the Noise Level Register.

### 14.1.6. The “Register Values” Report

The “Register Values” Report is used to return the value(s) of one or more Setup Registers. This message is usually generated in response to a “Get Register Values” Command. The “Register Values” Report has a Message Data ID of **0x90** and has the following syntax:

MDID	Arg1	Arg2	Arg3	Arg4	Arg5	Arg6	Arg7	Arg8
0x90	RR	VV	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]
Arg9	Arg10	Arg11	Arg12	Arg13	Arg14	Arg15	Arg16	Arg17
[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]

Where RR is the starting Setup Register number (0x00 – 0xFF) that is reported and VV is a list of up to 16 retrieved values (0x00 – 0xFF).

#### 14.1.7. The “RAM Values” Report

The “RAM Values” Report is used to return the value(s) of one or more SRAM Registers. The “RAM Values” Report has a Message Data ID of **0x91** and has the following syntax:

MDID	Arg1	Arg2	Arg3	Arg4	Arg5	Arg6	Arg7	Arg8
0x91	RR	VV	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]
Arg9	Arg10	Arg11	Arg12	Arg13	Arg14	Arg15	Arg16	Arg17
[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]

Where RR is the starting RAM register number (0x00 – 0xFF) that is reported and VV is a list of up to 16 retrieved values (0x00 – 0xFF).

#### 14.1.8. The “Raw Data” Report

The “Raw Data” Report is used to report from 1 to 16 bytes of raw data from one UPB device to another. The “Raw Data” Report has a Message Data ID of **0x92** and has the following syntax:

MDID	Arg1	Arg2	Arg3	Arg4	Arg5	Arg6	Arg7	Arg8
0x92	VV	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]
Arg9	Arg10	Arg11	Arg12	Arg13	Arg14	Arg15	Arg16	
[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	[VV]	

Where VV is a list of from 1 to 16 eight-bit raw data values.

#### 14.1.9. The “Heartbeat” Report

The “Heartbeat” Report is used to report that the transmitting device is still alive (plugged-in). The “Heartbeat” Report has a Message Data ID of **0x93** and has the following syntax:

MDID

0x93

#### 14.1.10. The “Acknowledgment” Report

The “Acknowledgment” Report is used to report that the transmitting device received and accepted a UPB Communication Packet with the MSG-bit set. The “Acknowledgment” Report has a Message Data ID of **0x94** and has the following syntax:

MDID

0x94

### 15. The Reserved Report Set #1

The Reserved Report Set is just that: reserved (for future use). There is currently no command messages defined for this Message Set, it is simply a placeholder for the future. This Message Set occupies the MDID range of 0xA0 to 0xBF.

### 16. The Reserved Report Set #2

The Reserved Report Set is just that: reserved (for future use). There is currently no command messages defined for this Message Set, it is simply a placeholder for the future. This Message Set occupies the MDID range of 0xC0 to 0xDF.

### 17. The Extended Message Set

The Extended Message Set is a (currently undefined) group of UPB Command and Report Messages that is available for individual UPB device manufacturers to define for themselves. This Message Set is represented by the Message Set ID (MSID) value of 111 along with a unique 5-bit Extended Set ID that identifies which of the 32 possible Extended Message Sets it is. It occupies the MDID range of 0xE0 to 0xFF.

If a device manufacturer wants to establish his own Extended Message Set he must first contact Powerline Control Systems to be issued a unique Extended Set ID. This Extended Set ID value should be placed in all messages using this new extended message set so that it won't interfere with other Message Sets.

#### 17.1. Extended Message Set IDs

Table 12 lists the currently assigned Extended Set IDs. Note that, when using the Extended Message Set, the actual message information is placed in the Message Data Argument field.

ESID	Assigned To	Message Set Purpose
0	Powerline Control Systems	Reserved for PCS use.
1	Powerline Control Systems	Reserved for PCS use.
2	Available to be assigned	For future use.
3	Available to be assigned	For future use.
4	Available to be assigned	For future use.
5	Available to be assigned	For future use.
6	Available to be assigned	For future use.
7	Available to be assigned	For future use.
8	Available to be assigned	For future use.
9	Available to be assigned	For future use.
10	Available to be assigned	For future use.
11	Available to be assigned	For future use.
12	Available to be assigned	For future use.
13	Available to be assigned	For future use.
14	Available to be assigned	For future use.
15	Available to be assigned	For future use.
16	Available to be assigned	For future use.
17	Available to be assigned	For future use.
18	Available to be assigned	For future use.
19	Available to be assigned	For future use.
20	Available to be assigned	For future use.
21	Available to be assigned	For future use.
22	Available to be assigned	For future use.
23	Available to be assigned	For future use.
24	Available to be assigned	For future use.
25	Available to be assigned	For future use.
26	Available to be assigned	For future use.
27	Available to be assigned	For future use.
28	Available to be assigned	For future use.
29	Available to be assigned	For future use.
30	Available to be assigned	For future use.
31	Available to be assigned	For future use.

Table 12: Currently Assigned Extended Set ID (ESID) Values



## 18. Bringing It All Together

Now you have read this whole document and you still don't quite understand how to use UPB communications to do something as simple as turn a UPB Dimming Wall Switch ON and OFF. Well this is the place to look because here we will bring it all together with explanations and examples.

### 18.1. Learning A UPB Device's Network Password

Before you can ever configure the Setup Registers of a new UPB Device you must first know its assigned Network Password so you can disable Setup Register Write Protection. Issuing the device either a "Write Enable" command or a "Start SETUP Mode" command can do this. However, both of these commands require (for security purposes) that you already know and supply the device's assigned Network Password. Furthermore, the UPB System Model prevents you (for security purposes) from reading a device's assigned Network Password unless it is in the SETUP Mode.

In summary, you can't read the assigned Network Password unless the device is in SETUP Mode and you can't issue a "Start SETUP Mode" command without knowing (supplying) the assigned Network Password. This whole "catch-22" situation was purposely designed into the UPB System Model to prevent unauthorized users from easily modifying a device's Setup Registers.

To get around this security protocol the UPB System Model specifies that each UPB device have at least one method to physically get the device into the SETUP Mode. This physical method is usually done with some sort of pushbutton press sequence but varies from device type to device type. It is up to device manufacturers to design a method to physically get their UPB device into SETUP Mode.

Once the new UPB device has been physically put in the SETUP Mode, the assigned 2-byte Network Password can be read by issuing the "Get Register Values" command to the Global NID (0x00) and the SETUP Mode DID (0xFE) as follows:

CTL		NID	DID	SID	MDID	Arg1	Arg2	CHK
0x09	0x00	0x00	0xFE	0xFF	0x10	0x02	0x02	0xE6

When the new UPB device that is in SETUP Mode receives this command it will respond with a "Register Values" Report message that includes the contents of the 2-byte Network Password (e.g. 0x1234) as follows:

CTL	NID	DID	SID	MDID	Arg1	Arg2	Arg3	CHK	
0x0A	0x00	0xFF	0xFF	0x01	0x90	0x02	0x12	0x34	0x75

Note that from this “Register Values” Report message we can see that the device’s Network Password is 0x1234 and also that its NID is 0xFF and its Unit ID is 0x01.

## 18.2. Reading A UPB Device’s Setup Registers

Assuming you now know the assigned Network ID (e.g. 0xFF) and Unit ID (e.g. 0x01) of your UPB device, you can read any of its Setup Registers (except the Network Password registers) by issuing a “Get Register Values” command to the device’s Network ID and Unit ID as follows:

CTL	NID	DID	SID	MDID	Arg1	Arg2	CHK
0x09	0x00	0xFF	0x01	0xFF	0x10	0x00	0xE8

When the addressed UPB device receives this command it will respond with a “Register Values” Report message that includes the contents of the specified Setup Registers (e.g. 0x00 – 0x0F) as follows:

CTL	NID	DID	SID	MDID	Arg1	Arg2	
0x0A	0x00	0xFF	0xFF	0x01	0x90	0xFF	
Arg3	Arg4	Arg5	Arg6	Arg7	Arg8	Arg9	Arg10
0x01	0x00	0x00	0x00	0x00	0x00	0x01	0x00
Arg11	Arg12	Arg13	Arg14	Arg15	Arg16	Arg17	CHK
0x22	0x00	0x01	0x00	0x00	0x00	0x01	0xFF

Note that the Network Password (Arg4 & Arg5) is protected and always reads as the value 0x0000 unless the device is in the SETUP Mode.

## 18.3. Commanding A UPB Device into the SETUP Mode

Assuming you know the assigned Network ID (e.g. 0xFF), Unit ID (e.g. 0x01) and Network Password (e.g. 0x1234) of your UPB device, you can command that device into the SETUP Mode by issuing it a “Start Setup Mode” command as follows:

CTL	NID	DID	SID	MDID	Arg1	Arg2	CHK
0x09	0x00	0xFF	0x01	0xFF	0x03	0x12	0xAF

When the addressed UPB device receives this command it will enter the SETUP Mode for the next five minutes. Re-issuing this command will start the five-minute time-out all over again. You can always check on how much time is left in SETUP Mode by issuing the device a “Get SETUP Time” command as follows:

CTL	NID	DID	SID	MDID	CHK
0x07	0x00	0xFF	0x01	0xFF	0xAF

When the addressed UPB device receives this command, it will respond with a “RAM Values” Report message that includes the number of ?? millisecond ticks that are left before timing out of SETUP Mode:

CTL	NID	DID	SID	MDID	Arg1	Arg2	CHK
0x09	0x00	0xFF	0xFF	0x01	0x91	0x5A	0xB9

Note that this device currently has 84 (0x54) ticks before it will time out of SETUP Mode. The Setup Mode Timer for this device (Arg1) happens to be at RAM address 0x5A.

#### 18.4. Assigning A UPB Device’s Network ID & Unit ID

Assuming your UPB device is in the SETUP Mode, you can assign it a new Network ID (e.g. 0x44) and Unit ID (e.g. 0x66) by issuing a “Set Register Values” command as follows:

CTL	NID	DID	SID	MDID	Arg1	Arg2	Arg3	CHK	
0x0A	0x00	0xFF	0x01	0xFF	0x11	0x00	0x44	0x66	0xAF

When the addressed UPB device (that is in SETUP Mode) receives this command it will write the values 0x44 (Arg2) and 0x66 (Arg3) into its Setup Registers 0x00 (Arg1) and 0x01 respectively.

#### 18.5. Controlling A UPB Dimming Wall Switch

Assuming you know the assigned Network ID (NID) and Unit ID (UID) of your UPB Dimming Wall Switch, you can easily control its Light Level using UPB Messages from the Device Control Command Set. In the examples to follow we will assume the UPB Dimming Wall Switch is assigned NID = 0x44 and UID = 0x66.

##### 18.5.1. Example: Turning A UPB Dimming Wall Switch ON

To turn a UPB Dimming Wall Switch ON to 100% simply send it a “Start Fade” command with a Light Level of 100 (0x64) and a Fade Rate of 0 (Snap) as follows:

CTL		NID	DID	SID	MDID	Arg1	Arg2	CHK
0x09	0x00	0x44	0x66	0xFF	0x24	0x64	0x00	0xC6

When the UPB Dimming Wall Switch receives this UPB Message it will immediately “snap” its Light Level to 100%.

### 18.5.2. Example: Turning A UPB Dimming Wall Switch OFF

To turn a UPB Dimming Wall Switch OFF simply send it a “Start Fade” command with a Light Level of 0 (OFF) and a Fade Rate of 0 (snap) as follows:

CTL		NID	DID	SID	MDID	Arg1	Arg2	CHK
0x09	0x00	0x44	0x66	0xFF	0x24	0x00	0x00	0x2A

When the UPB Dimming Wall Switch receives this UPB Message it will immediately “snap” its Light Level to 0% (OFF).

### 18.5.3. Example: Fading A UPB Dimming Wall Switch To 50%

To slowly fade a UPB Dimming Wall Switch ON to 50% simply send it a “Start Fade” command with a Light Level of 50 (0x32) and a Fade Rate of 4 (or any other Fade Rate desired) as follows:

CTL		NID	DID	SID	MDID	Arg1	Arg2	CHK
0x09	0x00	0x44	0x66	0xFF	0x24	0x32	0x04	0xF4

When the UPB Dimming Wall Switch receives this UPB Message it will slowly “fade” its Light Level to 50%. Note: the Fade Rate value can range from 0x00 to 0xFF and it is left up to the application designer to assign the exact meaning of each Fade Rate value for their device.

## 18.6. Activating A Link

TBD

## 18.7. Duplicate IDs

There is always the possibility that two or more UPB devices share the same Network ID and the same Unit ID. This condition can obviously cause a great deal of confusion if not detected and remedied. Fortunately, the UPB system has facilities built in to allow for the detection and elimination of Duplicate IDs.

### 18.7.1. Example: Duplicate ID Detection

Duplicate ID Detection can be performed as follows:

- 1) Perform Network Enumeration to find an unoccupied Network ID.
- 2) Put the NID/UID under test into SETUP Mode.
  - a. One or more devices will go into SETUP Mode and disable write protection.
- 3) Move that Unit ID to the unoccupied Network ID.
  - a. This is done by writing the unoccupied Network ID value to the Network ID Setup Register of the Unit ID under test.
  - b. One or more devices will now be assigned to the unoccupied Network ID.
- 4) Broadcast the "Auto Address" command to the unoccupied Network ID.
  - a. This will command all devices in that network to pick a new random Unit ID.
- 5) Perform Device Enumeration on the unoccupied NID to determine all of the occupied Unit IDs on that UPB network.
  - a. If only one Unit ID is occupied then there were no duplicates.
    - i. Move the single device back to the original NID giving it an unoccupied Unit ID.
  - b. If more than one Unit ID is occupied then there were duplicates.
    - i. Move each device back to the original NID giving them unoccupied Unit IDs.

## 18.8. Auto-Addressing

TBD

## 19. Glossary Of UPB Abbreviations and Terms

ACK-bit	Bit #4 of the Control Word (CTL) that is used to request an ACK Pulse to be generated.
ACK Pulse	A single UPB Pulse generated by a UPB receiving device that is used to inform the transmitting device that the UPB Communication Packet was accepted.
Broadcast DID	A Destination ID (DID) value of 0x00 that all UPB devices on a network will accept.
Configuration Registers	Any additional Setup Register(s) not used by the UPBID.
CTL	Packet Control Word. The first two-bytes of a UPB Packet Header that contains bit fields that are used to indicate such information as: how the UPB Communications Packet should be received and how it should be responded to, as well as its length and sequence information.
Device Components	Logical objects (records) that the UPB device has, usually associated with physical entities on the device (such as pushbuttons, switches, indicators, input channels, output channels, etc.), that are intended to either receive or transmit UPB Link Packets for control purposes.
Device Enumeration	A method available for transmitting devices to quickly determine the Unit IDs of all receiving devices that exist on a particular UPB network.
DID	Destination ID. The 3 <sup>rd</sup> byte of a UPB Communication Packet that indicates who it is intended for.
Direct Packet	A UPB Communication Packet that uses the DID field for a Unit ID. Direct Packets are usually used for communicating to a single device.
ESID	Extended Set ID.
Global NID	A Network ID (NID) value of 0x00 that all UPB devices accept.
ID-bit	Bit #5 of the Control Word (CTL) that is used to request an ID Pulse generation.
ID Pulse	A single UPB Pulse that is generated in Position #3 of the UPB Frame (immediately following the end of a UPB Message) that corresponds to the receiving device's Unit ID

Link Packet	A UPB Communication Packet that uses the DID field for a Link ID. Link Packets are usually used for communicating to a group of device components.
LNK-bit	Bit #15 of the Control Word (CTL) that is used to indicate that the UPB Communication Packet is a Link Packet.
MDA	Message Data Arguments. The bytes in a UPB Message that follow the MDID.
MDID	Message Data ID. A unique identifier byte for a UPB Message.
MID	Message ID or Manufacturer ID. The Message ID is a unique 5-bit identifier for the particular command or report in a specified Message Set. The Manufacturer ID is a unique 16-bit code assigned to each UPB device manufacturer.
MSG-bit	Bit #6 of the Control Word (CTL) that is used to request an Acknowledge Message generation.
MSID	Message Set ID. A 3-bit field of the MDID that breaks the UPB Messages into eight Message Sets.
Network Enumeration	A method available for transmitting devices to quickly determine which Network IDs are occupied (by UPB devices).
Network Password	A special 16-bit code (0 – 65,536) that is assigned to a UPB device that must be supplied when trying to do such things as put the device into SETUP Mode and disable its write protection.
Normal Mode	A UPB device's normal (or standard) mode of operation.
NID	Network ID. A unique 8-bit identifier assigned to each UPB device that allow it to communicate with other UPB devices with the same NID but not with devices with another NID.
PCS	Powerline Control Systems. The company, in Northridge California that developed the Universal Powerline Bus powerline communication method and protocol.
SETUP DID	A Destination ID (DID) value of 0xFE that all UPB devices on a network that are in the SETUP Mode will accept.
SETUP Mode	A special mode of operation in which Setup Register Write Protection is temporarily disabled allowing Setup

	Registers to be written.
Setup Registers	A block of from 64 to 256 non-volatile 8-bit registers that every UPB device shall have available for identification and configuration purposes.
SID	Source ID. The Unit ID of the UPB device that generated the UPB Communication Packet.
TBD	To Be Determined
UID	Unit ID. A unique 8-bit identifier assigned to each UPB device in a UPB network.
UPB	Universal Powerline Bus.
UPB Byte	A group of four UPB Pulses that represent an 8-bit value.
UPB Communication Packet	A variable length structure of UPB Bytes used to communicate useful information from one device to others.
UPB Frame	A period of time just prior to every AC zero-crossing point where UPB communication is performed.
UPBID	A set of 64 Setup Registers that are used to identify the UPB device to the UPB network.
UPB Pulse	A precisely timed electrical pulse that is superimposed on top of the normal AC power waveform (sine wave) that is used to communicate useful information based on its position.
UPB Pulse Position	The relative position value of a UPB Pulse. There are four possible Pulse Positions (0, 1, 2, or 3) in each UPB Frame.
UPB Repeater	A special class of UPB device that are designed to only accept a Repeater Packet and then retransmit it as a Non-Repeater Packet.
Zero-Crossing	The point in time where the AC powerline voltage is at zero volts. This occurs roughly every 8.333mS on a 60Hz powerline.