OpenLCB Technical Note	
Common Information	
10/08/10	Preliminary

1 Introduction

The OpenLCB Standards are independently normative. They are what they are, and they say what they say.

This Technical Note provides common background information that many be useful as you read existing Standards and Technical Notes, and may provide useful guidance as you write new ones.

This Technical Note is not normative in any way.

2 Data

2.1 Numerical representation

OpenLCB documents use the prefix “0x” to indicate a hexadecimal value, and “0b” to represent a binary value.

Constants should always include the full field length. A value for a 12-bit field should be written as 0x002 or 0b000000000010, not 0x2 or 0b10.

Commas can be used in numbers after decimal thousands (65,523), 16-bit double-bytes for hex constants (0x1234,4567), and four bits in binary (0b1000,0000). Do not put a space after the comma.

OpenLCB does not define a floating point representation (yet).

2.2 Byte sequences

OpenLCB is, by default, big-endian. When sending multi-byte data, the byte containing the most significant bits is sent first. This is because CAN, particularly the header, and the various Internet protocols are all big-endian.

2.3 Strings

OpenLCB strings are sequences of plain ASCII 7-bit values. In particular, we want to retain the 0x80 bit in the first byte as a way of eventually indicating other codings.

OpenLCB does not prefer length-coded or null-terminated strings. Standards should specify which is used in each case.

OpenLCB uses newline, also known as line-feed (\n 0x0A) as the line-end character. Carriage return (\r 0x0D) should be considered as general white space.

DRAFT

Copyright 2010, by J. Day, D. Harris, B. Jacobsen, and A. Shepherd

There are no specific tab settings. You cannot assume a tab is any particular number of spaces. You can assume it counts as non-null white space.

30

Table of Contents

DRAFT

Copyright 2010, by J. Day, D. Harris, B. Jacobsen, and A. Shepherd