# Low Cost DCC (LC-DCC) 2.2

# Controller Command Reference

22nd April 2020

## Summary

Provided in this document are the details of the LC-DCC controller serial and WIFI commands used for train control and programming.

## Disclaimer

The designers accept no responsibility for any damage to any train or accessory decoder connected to this DCC system through incorrect assembly or use of the hardware design.

Please read s-9.1_electrical_standards_2006.pdf NMRA standard before purchasing and using a power supply. Also note some cheap power supplies can give over voltage output.

## License / Usage Terms

All the software components are protected by license. When buying the software from eBay, you are entitled to one license which will be provided by Email. If you require multiple licenses then contact eBay seller.

## Existing users of the 1.X design

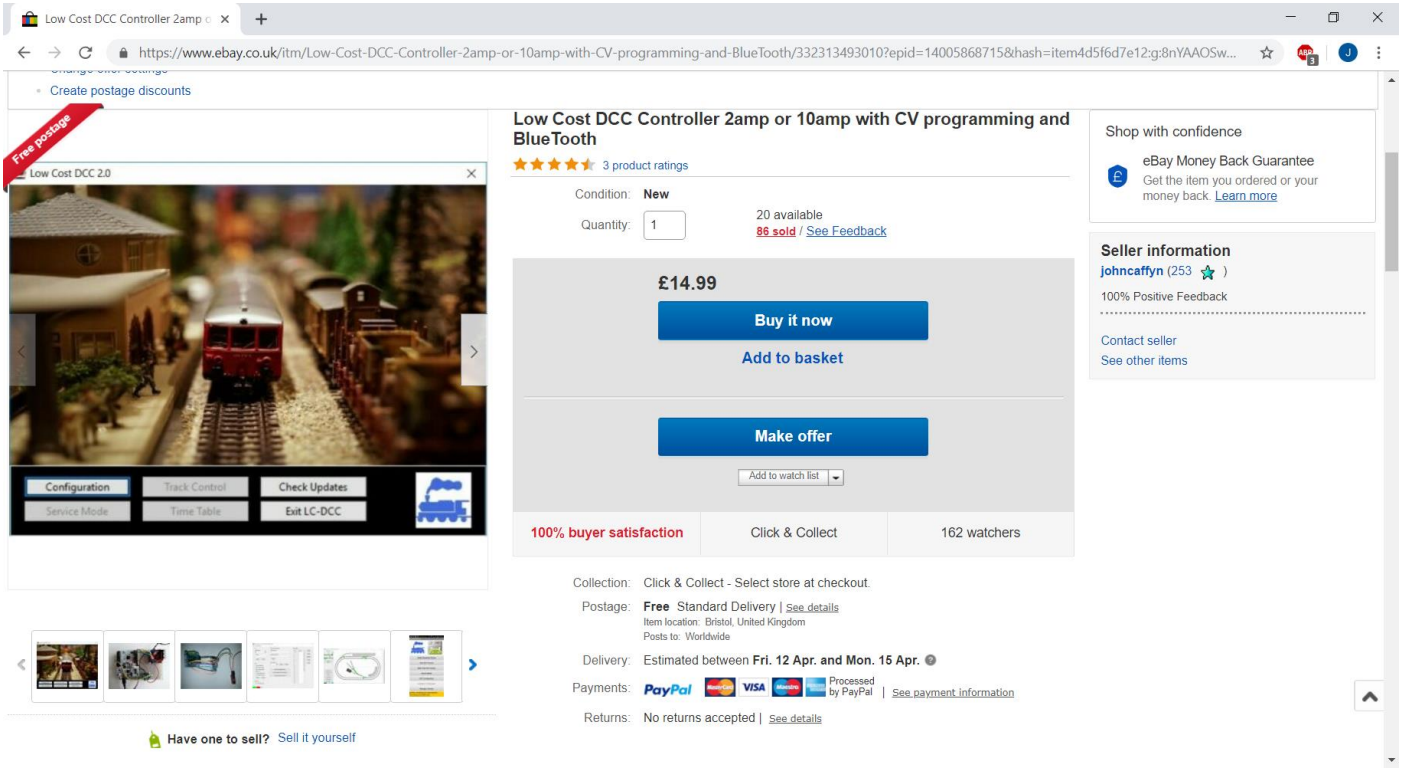
Anyone using version 1.X of this design should be aware that the function of M3/M4 pins A6/A7/PC0/PC1/PB0/PB1 have been swapped. Users should check the latest build documents for version 2.0 onwards. Existing users will need to obtain a license key, this will be provided free if you have purchased via my eBay account. Users creating their own software who want to upgrade will also need to contact us to obtain an “unlock code” for the M3/M4.

## LC-DCC Forum

A low cost DCC forum can be found at http://low-cost-dcc.freeforums.net, this was created in March 2019.

# Buying On EBay

Please note that there are a number of false adverts on eBay that cannot provide support or valid updates for this project and may not even provide anything for your money. The idea of Low Cost DCC is to make available a quality product at a price that everyone can afford. Only buy from the designer and developer of this project. To ensure you are buying from the only official eBay listing check the seller information is as displayed below (seller: johncaffyn, location: Bristol). Any other listing is false.



# Contents

# Introduction

This book describes the serial and WIFI commands used by the LC-DCC system for programming interface version PI06.

The LC-DCC system has three modes of operation:

- Idle
- Operational
- Service

In IDLE mode the controller generates only NMRA idle packets and no train control is possible.

In SERVICE mode the controller only responds to CV read and write commands.

In OPERATIONAL mode the controller responds to train and accessory control commands.

All commands are ASCII, there are two types of command: single character commands and multi character commands terminated by a line feed (ASCII decimal 10).

All error responses from the controller are preceded by an exclamation mark character (!).

# Controller Commands

The following sections describe all the controller commands supported. The commands apply to both the WIFI and serial USB/Bluetooth designs.

## Mode Control Commands

The following table defines the mode control commands:

| Command | Controller Response | Description |
|---|---|---|
| M0 | MODE-IDLE | Idle mode command. |
| M1 | OP-MODE | Operational mode command. |
| M2 | SERVICE-MODE | Service mode command. |

All commands are terminated by a line feed and all responses are terminated by a line feed.

## Power Control Commands

The following table defines the controller track power commands:

| Command | Controller Response | Description |
|---|---|---|
| PA0 | *None* | Disable track A power. |
| PA1 | *None* | Enable track A power. |
| PB0 | *None* | Disable track B power. |
| PB1 | *None* | Enable track B power. |

All commands are terminated by a line feed.

## Licensing Commands

The following table defines the controller licensing commands:

| Command | Controller Response | Description |
|---|---|---|
| ?ID | XXXXXXXX-XXXXXXXX-XXXXXXXX-XXXXXXXX | Reads controller unique identification registers. |
| UC=xxxxxxxx | xxxxxxxx<br>or<br>!UNLOCK | Unlock controller command to enable controller operation once a valid license has been detected. |

All commands are terminated by a line feed.

The ID command returns four hexadecimal 32 bit numbers from the controller identification register.

The UC command receives an unlock code generated from the ID response as eight hexadecimal lowercase characters.

If you are developing your own software to use with the controller you will need to obtain an unlock code from support@swws.co.uk for each controller you build and use.

## Configuration Commands

The following table defines the controller configuration commands:

| Command | Controller Response | Description |
|---|---|---|
| NSS=28 | *None* | Set engine speed default steps to 28. |
| NSS=128 | *None* | Set engine speed default steps to 128. |
| LAM=0 | *None* | Set long address mode off, 7 bit addressing. |
| LAM=1 | *None* | Set long address mode on, 14 bit addressing. |
| ENLK=0 | *None* | Engine locks off. |
| ENLK=1 | *None* | Engine locks on. |
| ACLK=0 | *None* | Accessory Locks off. |
| ACLK=1 | *None* | Accessory Locks on. |
| FR= | *None* | Set engine function repeat sends value 0 .. 9, 0 is continuous |
| AR= | *None* | Set accessory repeat sends value 1 .. 9 |

All commands are terminated by a line feed.

## Status Commands

The following table defines the controller status commands:

| Command | Controller Response | Description |
|---|---|---|
| ?CST | CST:a:b:c:d:e:f:g;h:i | Controller status query. |
| ?TS | TS:AB:12345 | Track status query. |

The track status response TS is followed by the track power on/off status for track A and B followed by the current being read by the INA219 in milli-amps followed by line feed. When the either track is not powered a 0 (zero) is returned instead of A or B. If a current overload has been detected then the A/B status will be replaced by ** and the measured overload current will be returned.

The controller status response CST has the following fields:

| Field | Description |
|---|---|
| a | Indicates if the long/short address mode is programmed and whether the speed steps have been configured |
| b | The number of speed steps, either 28 or 128 |
| c | The address mode, either L for long address mode (14 bits) or S for short address mode (7 bits) |
| d | Function repeat setting, a value between 1 and 255 |
| e | Accessory repeat setting, a value between 1 and 255 |
| f | Engine locks setting either 1 for ON or 0 for OFF |
| g | Accessory locks setting either 1 for ON or 0 for OFF |
| h | The overload current setting in milli-amps or 0 if no INA219 detected |
| i | The maximum allowed overload current setting in milli-amps |

## Service Mode Commands

The following table defines the controller service mode commands:

| Command | Controller Response | Description |
|---|---|---|
| WV | OK | CV write command to program CV. |
| RV | CV Value Read | CV read command to read CV value. |

The WV command is followed by the CV number and the value to write, such as WV1=3 followed by a line feed.

The RV command is followed by the CV number to read, such as RV3 followed by a line feed. The response is CV followed by the CV number and value read, such as CV3=7 followed by a line feed.

## Operational Mode Commands

The following table defines the controller service mode commands:

| Command | Controller Response | Description |
|---|---|---|
| X | *None* | Generate NMRA DCC reset packets. |
| E | *None* | Set engine address for engine commands |
| Y | *None* | Set engine control to 28 speed steps |
| J | *None* | Set engine control to 128 speed steps |
| Q | ES: | Query engine settings. |
| + | *None* | Increase engine speed. |
| - | *None* | Decrease engine speed. |
| 0 | *None* | Set engine speed to stop. |
| ! | *None* | Emergency stop all engines. |
| H | *None* | Broadcast stop all engines. |
| > | *None* | Set engine direction forward. |
| < | *None* | Set engine direction reverse. |
| S | *None* | Set engine speed to value following S, for example S0, S28, S126 |
| F0= | *None* | Set function F0 ON (1) or OFF (0) |
| F1= | *None* | Set function F1 ON (1) or OFF (0) |
| Fn= | *None* | Set function n ON (1) or OFF (0) |
| F28= | *None* | Set function F28 ON (1) or OFF (0) |
| FA= | *None* | Function values F0 and F1 to F4 |
| FB= | *None* | Function values F5 to F12 |
| FC= | *None* | Function values F13 to F20 |
| FD= | *None* | Function values F21 to F28 |
| FV= | *None* | Set engine sound volume value |
| PQ | *None* | Add DCC packets to the output queue |

Commands X,J,Y,Q,+,-,0,!,<,> are all single character commands without a line feed following them.

The function commands F0, F1 up to F28 use the same command format which is F0=1 for ON and F0=0 for function OFF etc.

The speed command S if followed by the step value; for example, S28, S0 can be used for engine stop.

All commands are terminated by a line feed.

The engine status response ES: has the following format: ES:a:b:c:d:e:f:g:h:i

**a** Engine address **b** Engine Speed **c** Function FA bits

**d** Function FB bits **e** Function FC bits **f** Function FD bits

**g** Engine Volume **h** Engine Speed Steps **i** Potentiometer controlling engine

The PQ command is followed by the packet bytes values as hexadecimal character pairs. For example, a DCC reset packet can be added to the queue using:

PQ000000

Multiple packets can be added by adding * and the number of repeats from 1 to 9, for example:

PQ000000*4

generates four DCC reset packets

Some other examples are:

DCC Idle packet command PQFF00FF

## Diagnostics Commands

The commands are available for diagnostics use only, they report various controller status information.

| Command | Controller Response | Description |
|---|---|---|
| ?LPB | *See Below* | Display processor DCC packet performance. |
| ?ADC | *See Below* | ADS1115 ADC status. |
| ?POT1 | *See Below* | Display potentiometer 1 status. |
| ?POT2 | *See Below* | Display potentiometer 2 status. |
| ?POT3 | *See Below* | Display potentiometer 3 status. |
| ?POT4 | *See Below* | Display potentiometer 4 status. |
| ?POT5 | *See Below* | Display potentiometer 5 status. |
| ?POT6 | *See Below* | Display potentiometer 6 status. |
| ?POT7 | *See Below* | Display potentiometer 7 status. |
| ?POT8 | *See Below* | Display potentiometer 8 status. |
| z | *None* | Enables command character echo so characters entered while a terminal is connected to the controller are displayed |
| Z | *None* | Disables command character echo |

**?LPB Response**

LPB=0,MW1=213,MW2=179,MPB=80

The late packet byte response shows how well the processor is generating DCC packet bits. The LPB indicates the number of times the processor has failed to meet a packet bit output deadline. The MW1 and MW2 show how much time is spare between the packet bit deadline and the processing of other interfaces such as UARTs and I2C buses. The MPB shows the maximum number of packet bits transferred in the longest packet byte sequence.

**?ADC Response**

ADC1:EN=1,ADC2:EN=1,CH=1,SV=4768

This response indicates the state of the two ADC modules (enabled or disabled) the last channel read (1..8) and the last sample voltage that has been digitized.

**?POTx Response**

P2,ST=EN,VL=13083,SP=-256,EA=0,AC=13211

P1,ST=DA,VL=4722,SP=-256,EA=0,AC=2371

The potentiometer number 1 to 8 follows a “P”, followed by EN (enabled), VL (ADC value last read), SP (engine speed), EA (engine address) and AC (analogue centre reading for zero speed).

## Analogue Control Commands

These commands are only supported if the ADC (ADS1115) is connected and the correct software is programmed into the M3 or M4 processor.

| Command | Controller Response | Description |
|---|---|---|
| P1EA | *None* | Set the engine address to be controlled by POT1 ADS1115 A0 input. |
| P2EA | *None* | Set the engine address to be controlled by POT2 ADS1115 A1 input. |
| P3EA | *None* | Set the engine address to be controlled by POT3 ADS1115 A2 input. |
| P4EA | *None* | Set the engine address to be controlled by POT4 ADS1115 A3 input. |
| P5EA | *None* | Set the engine address to be controlled by POT5 ADS1115 A0 input. |
| P6EA | *None* | Set the engine address to be controlled by POT6 ADS1115 A1 input. |
| P7EA | *None* | Set the engine address to be controlled by POT7 ADS1115 A2 input. |
| P8EA | *None* | Set the engine address to be controlled by POT8 ADS1115 A3 input. |

These commands program the engine address to be controlled when a potentiometer is connected to the ADS1115 input. Up to eight engines can be controlled using two ADS1115 modules or four engines with one module.

An example command is P1EA=3 followed by a line feed. To disable the engine control potentiometer, use P1EA=0 for example.

## LC-DCC RFID Train Location

The following table contains information related to the RFID train location system under development.

| Command | Controller Response | Description |
|---|---|---|
| L | *None* | Send RFID location information to application. |

This command is a location command sent from the RFID train location system and is forwarded by the LC-DCC controller to the connected PC and Android phone or tablet.

The format of the message is:

L<location><type><identifier> followed by a line feed, all numbers are hexadecimal.

For example

L00139876 <line feed>

Where 001 is the location of the RFID detector

Where 3 is the type, this allows engines and rolling stock to be identified separately

Where 9876 is always a four hex digit number for the engine/rolling stock identification

# Version Change History

**April 2020**

Added support for two ADS1115 and up to eight potentiometers.  Added PQ (packet queue) command.  Added 'H' command for broadcast all engines stop.

**April 2019**

Initial version.