DC-707 POS Interface Manual

Copyright ©2013, MicroTec
Taarngade 9 • DK-7600 Struer • Denmark

All Rights Reserved

This document may not be reproduced or transmitted in any form by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system without written permission.

Table of Contents

Contents ____ *ii*

Introduction ____ *3*

MicroTec Basic POS Interface ____ *3*

Electronic specification ____ **3**

MicroTec Basic POS Protocol ____ *5*

Data format: ECU to POS ____ **5**

Data format: POS to ECU ____ **7**

DC-707 POS settings ____ *9*

Enable POS and set up a PLU Base ____ **9**

PLU Table ____ **10**

Introduction

The MicroTec Basic Cash Register Interface allows a Cash Register or POS system to interface with DC-707 dispensing units via a serial port for ringing drinks in an intelligent manner. By installing or enabling the MicroTec Basic driver in the DC-707 dispensing unit and complying with the following specification, the DC-707 dispensing system will send PLU (price look up) information to a cash register and optionally wait for acknowledgement from the register prior to dispensing.

A DC-707 dispensing unit will be referred to as an ECU (electronic control unit).

Different manufacturers may refer to their sales terminals as ECR, POS, Point of Sale or other terminology. This specification will use POS to refer to any sales terminal capable of interfacing to a DC-707 ECU dispenser unit via a serial port.

PLU is a numeric string that is used to indicate what was dispensed. It is known in some POS systems as NLU.

MicroTec Basic POS Interface

Electronic specification

The MicroTec Basic Interface will communicate with a register via a RS-232 data link with the following parameters:

Baud Rate : 9600 bps
Data Bits : 8
Parity : None
Stop Bits : 1
Handshaking : None

The only lines always used by the DC-707 ECU are TxD, RxD, and GND. These are defined as follows:

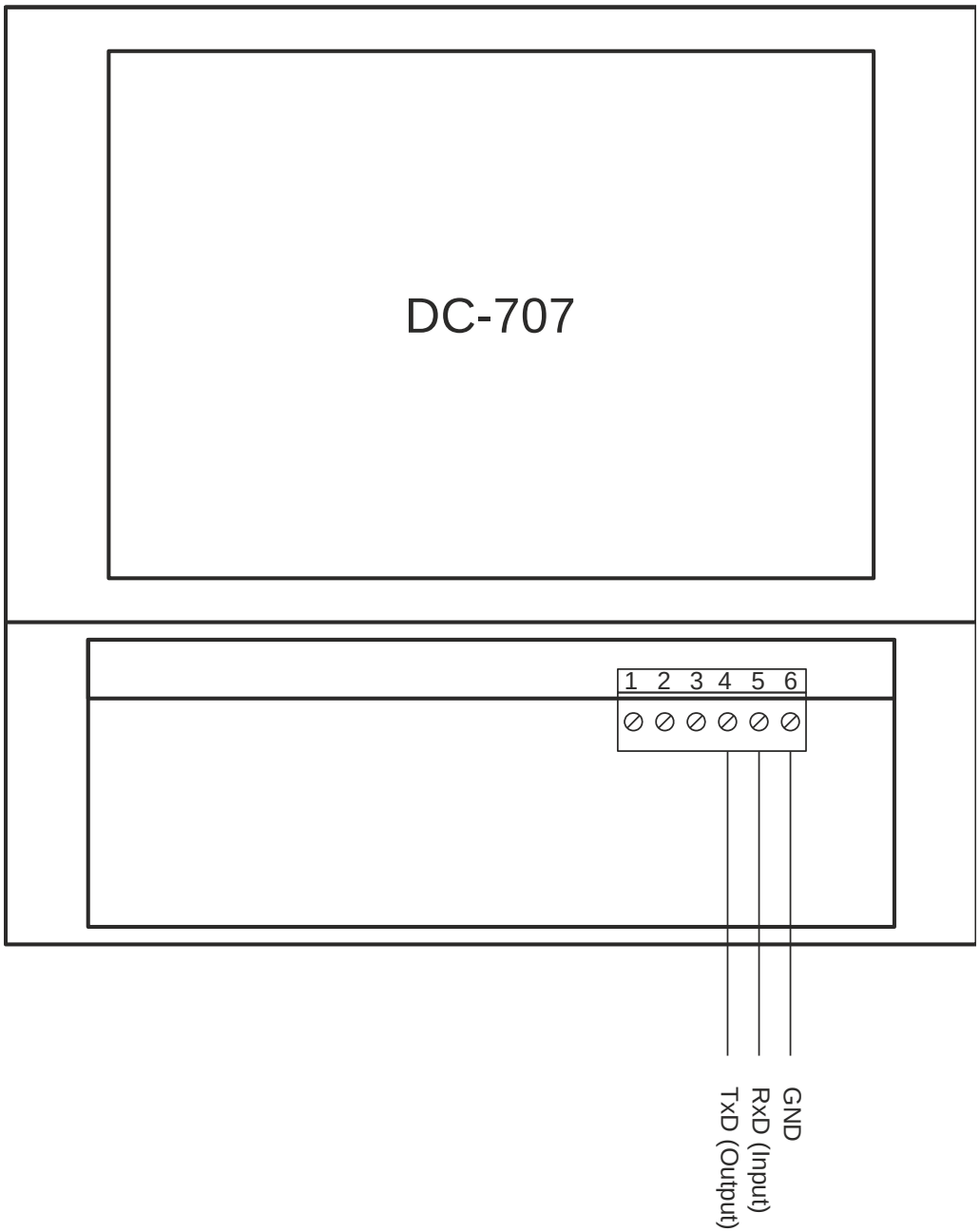
TxD : TRANSMIT DATA
Data transmitted from DC-707

RxD : RECEIVE DATA
Data received by DC-707

GND : SIGNAL GROUND

The RS232 Connector of the DC-707 ECU is a 6 Pin screw terminal connector with pin definitions as follows:

PIN	Signal		
1			
2			
3			
4	Transmit Data	(TxD)	OUTPUT
5	Receive Data	(RxD)	INPUT
6	Signal Ground	(GND)	



MicroTec Basic POS Protocol

Data format: ECU to POS

The DC-707 ECU will send a variable length packet of data prior to dispensing a drink. This packet contains Start (STX) and End (ETX) of transmission bytes as well as a PLU number in ASCII, modifiers and trailers, and a Logical Redundancy Check byte (LRC). The STX, ETX, PLU, and LRC are always sent in a packet but the modifier and trailer bytes are optional. The transmission packet will look as follows with items in ()'s being optional.

STX (Modifiers) PLU (Trailers) ETX LRC

FIELD DEFINITIONS:

STX 02H Start of Transmission

This byte signals the start of a packet of data.

PLU Price Look Up

The PLU field is a 1 byte or longer sequence which signifies what is being dispensed. The length of the PLU is the number of digits needed to express the PLU. There is no zero padding. A PLU cannot be zero. The range of PLU values varies depending on the type of DC-707 ECU. These values are sent out as an ASCII sequence, most significant byte first, according to the following table:

PLU – ASCII Mapping Table

Numerical Value	ASCII Value in Hex
0	30H
1	31H
2	32H
3	33H
4	34H
5	35H
6	36H
7	37H
8	38H
9	39H
Example : PLU = 4598 Packet Data = 34H 35H 39H 38H	

MODIFIER AND TRAILER FIELDS

NOTE: This driver does not implement any escape sequence. Therefore, modifiers and trailers must avoid using STX or ETX as a character.

Modifiers and trailer bytes are a series of byte values. The only byte values that are not allowed in these fields is the NULL value 00H, STX value 02H and ETX value 03H. These fields are used to modify the action on a PLU or cause an action to occur. For example, a modifier with a value of

05H may be used to signal that the current drink is a small portion of a given PLU value. A trailer byte of 0DH might be used to indicate that the drink should be entered into the register. Modifiers and trailers are optionally programmed at the DC-707 ECU and it is up to the designer of the POS system to decide whether they may be useful to his or her system and the significance of the values. Most existing systems do not use the modifier and trailer information.

Some DC-707 ECUs can be configured to send different modifiers or trailers based on price level or size. The same modifier and trailer sequence will be sent for all pours dispensed as price level X and portion size Y. For a DC-707 ECU that supports three price levels (A, B, C) and three sizes (S, R, L), this leads to 9 distinct Modifier and Trailer sequences.

Other modifiers and trailers will be the same for all pours.

ETX 03H End of Transmission

This byte signals the end of the packet data.

LRC Logical Redundancy Check

This one byte value is calculated by performing an XOR function on all of the bytes in the data stream other than the LRC itself. This means the STX, ETX, PLU, and all modifier and trailer bytes are included in the LRC calculation.

NO ESCAPE CHARACTER is used

There is no escape character used by the MicroTec Basic interface. Escape characters are used to distinguish data from control bytes. The control bytes in this case are STX and ETX. Since the PLU is in ASCII and the LRC occurs after the ETX, there is no possibility that a 02H or 03H may occur in the data stream between the STX and ETX unless one of these characters was programmed as a modifier or trailer. STX and ETX must not be used in the Modifier and Trailer fields. Otherwise, the POS may incorrectly interpret these as STX or ETX control fields.

The LRC does not need to be escaped since it is always the single byte immediately following the ETX.

EXAMPLE LRC CALCULATION

Example Data Packet #1

The data to send in this case is a PLU of 135 along with a modifier equal to 14H and a trailer of 21H. The LRC is not calculated yet so we leave it as a ?? value.

Logical Data Packet

STX	MODIFIER	PLU	TRAILER	ETX	LRC
02H	14H	135	21H	03H	??

Since the PLU must be sent as an ASCII value we expand the PLU 135 into a 3-byte sequence.

Preliminary Data to Send

STX	MODIFIER	PLU DATA	TRAILER	ETX	LRC
02H	14H	31H 33H 35H	21H	03H	??

Now we proceed with the LRC calculation:

Data Type		Hex	Binary			
				LRC		
STX		02H	00000010b	00H	00000000b	Starting LRC
				02H	00000010b	
Modifier		14H	00010100b	02H	00000010b	Intermediate LRC
				14H	00010100b	
PLU 135m	(1)	31H	00110001b	16H	00010110b	Intermediate LRC
				31H	00110001b	
	(3)	33H	00110011b	27H	00100111b	Intermediate LRC
				33H	00110011b	
	(5)	35H	00110101b	14H	00010100b	Intermediate LRC
				35H	00110101b	
Trailer		21H	00100001b	21H	00100001b	Intermediate LRC
				21H	00100001b	
ETX		03H	00000011b	00H	00000000b	Intermediate LRC
				03H	00000011b	
				03H	00000011b	Final LRC

The LRC calculates to 03H

Final Data to Send

STX	MODIFIER	PLU DATA	TRAILER	ETX	LRC
02H	14H	31H 33H 35H	21H	03H	03H

Data format: POS to ECU

With the MicroTec Basic driver installed, the DC-707 ECU will send the above packet of data and then wait for a response from the POS system. This response may be one of two things. Either an acknowledge byte (ACK) or a negative acknowledge (NAK). It is the responsibility of the POS system to verify that the correct data has been received from the DC-707 ECU by checking the LRC byte. If the data is not correct, the POS must respond with a NAK. If the data is correct then the POS must check to see if this request for a drink contains a valid PLU. If it does not, the POS must respond with a NAK. The POS may also want to make sure that the register is ready to ring a drink on its terminal. If everything is correct on the register end then the register should respond with an ACK.

The DC-707 ECU may be set to pour with or without release.

If pour without release is set, the DC-707 ECU will pour as soon as the PLU packet has been sent regardless or the response from the POS. The packet will not be resent even if a NAK is received.

If pour with release is set, then the DC-707 ECU will not dispense the drink until an ACK has been received. If the ECU does not receive a response within a user-defined time, the DC-707 ECU will act as if it had received a NAK. At this point the ECU may retry or the bartender may again request the pour of a drink. **During the time from the start of packet to the reception of an ACK the DC-707 ECU will not pour. It is important that the POS designer keep this time as short as possible.**

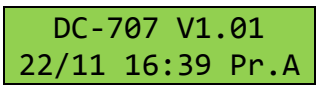
POS Response Values

Response	Hex Values
ACK	**06H**
NAK	**15H**

DC-707 POS settings

Enable POS and set up a PLU Base

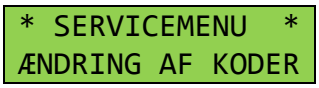
When the ECU sends pour information to a POS system, it uses a pre-defined PLU for the pourer code/size set up from a base PLU you select.



Enter the service code.



The service menu will now appear.



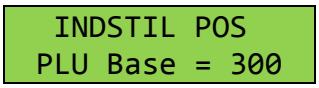
Press the * -key until the menu for POS settings is displayed.



Press the # -key to change the POS settings.



By default the POS communication is disabled. Press the * -key to enable POS communication.
Press the * -key until the desired PLU Base is displayed e.g. “PLU Base = 300”.



Press the # -key to save the selected PLU Base.
Press the # -key to exit the service menu.

PLU Table

Find the column for the base PLU you've selected to determine the specific PLU for each pourer/size.

	PLU Base 100	PLU Base 200	PLU Base 300	PLU Base 400	PLU Base 500	PLU Base 600	PLU Base 700	PLU Base 800	PLU Base 900
Code 1, Size 1	101	201	301	401	501	601	701	801	901
Code 2, Size 1	102	202	302	402	502	602	702	802	902
Code 3, Size 1	103	203	303	403	503	603	703	803	903
Code 4, Size 1	104	204	304	404	504	604	704	804	904
Code 5, Size 1	105	205	305	405	505	605	705	805	905
Code 6, Size 1	106	206	306	406	506	606	706	806	906
Code 7, Size 1	107	207	307	407	507	607	707	807	907
Code 1, Size 2	111	211	311	411	511	611	711	811	911
Code 2, Size 2	112	212	312	412	512	612	712	812	912
Code 3, Size 2	113	213	313	413	513	613	713	813	913
Code 4, Size 2	114	214	314	414	514	614	714	814	914
Code 5, Size 2	115	215	315	415	515	615	715	815	915
Code 6, Size 2	116	216	316	416	516	616	716	816	916
Code 7, Size 2	117	217	317	417	517	617	717	817	917
Code 1, Size 3	121	221	321	421	521	621	721	821	921
Code 2, Size 3	122	222	322	422	522	622	722	822	922
Code 3, Size 3	123	223	323	423	523	623	723	823	923
Code 4, Size 3	124	224	324	424	524	624	724	824	924
Code 5, Size 3	125	225	325	425	525	625	725	825	925
Code 6, Size 3	126	226	326	426	526	626	726	826	926
Code 7, Size 3	127	227	327	427	527	627	727	827	927