GA964 - NV11

ccTalk Communication Protocol Specification (Additional Commands)

INTELLIGENCE IN VALIDATION

Innovative Technology
INTELLIGENCE IN VALIDATION

Automated Transactions
INTELLIGENCE IN VALIDATION

Bellis Technology
INTELLIGENCE IN VALIDATION

CONTENTS

Copyright Innovative Technology Ltd 2012

1 REVISION HISTORY

Issue	Date	Firmware	By	Notes
A		NA	GC	First draft for review
B	08/09/2010		SR	Technical Review
1	10/09/2010	3.31	SR	First Release
1.1	23/09/2010		SR	Added detail on command description.
2	08/10/2010	3.31 P21	SR	Added commands 33, 34 & 36
2.1	20/01/2011		SR	Updated examples
3	15/03/2011	3.33	SR	Reordered commands Add empty-only details Added commands 19, 20 and 35
3.1	01/07/2011	3.34	SR	Corrected length on command 35 Reordered commands
4	29/09/2011	3.36	SR	Add chapter on DES note dispense, command 18
5	27/03/2012	3.39	SR	Added multi denomination commands. Added command 17, replacing command 18.
6	10/08/2012	3.41	SR	Added flag 2, bit 4 to command 20 response.

Copyright Innovative Technology Ltd 2012

2 INTRODUCTION

This document details the additional commands used by the NV11 in ccTalk. This document should be read in conjunction with the full ccTalk specification. Only the commands in addition to the standard BNV ccTalk specification are documented. The aim of this document is to provide an interface for the NV11 that operates in ccTalk systems.

The note validator commands conform to the standard ccTalk specification and build upon it to provide recycling functionality. The NV11 can be configured to store either single denomination notes or multiple denomination notes. Each note to be recycled for later payout will be stored in the Note Float recycler rather than the stacker, until the unit is full or a programmable limit is reached.

Please see the full NV11 manual for a detailed operation description.

If in doubt please contact Innovative Technology for details of any changes.

Innovative Technology Ltd has a policy of continual product improvement. As a result the products supplied may vary from the specification described here.

Copyright Innovative Technology Ltd 2012

3 COMMAND SUMMARY

[Header] Command	Parameters	Reply
[17] - Request Encrypted Expanded Status [0x11]	SEL, CHAL	See section 5.4.1
[19] – Force Empty NV11 Store to Cash Box [0x13]	SEL	EVENT
[20] - Request Expanded NV11 status [0x14]	SEL	FLAG, EVENT, REMAIN, PAID, UNPAID, STORED, STORE, FLAG2
[21] - Clear Total Count [0x15]	SEL	ACK / NAK
[22] - Pump RNG [0x16]	R1 – R8	ACK
[23] - Request Cipher Key [0x17]	None	Key1 – Key8
[24] - Request Variable set [0x18]	SEL	KEY, MC
[25] – Modify Variable Set [0x19]	SEL, KEY	ACK
[26] – Request Total Count [0x1A]	SEL	IN1-IN3, OUT1-OUT3,STR1-STR3
[27] – Enable Payout [0x1B]	KEY	ACK / NAK
[28] – Dispense NV11 bills [0x1C]	SEL, SEC1-SEC8, BILL CNT	EVENT / NAK
[29] - Request NV11 status [0x1D]	SEL	FLAG, EVENT, REMAIN, PAID, UNPAID, STORED, STORE
[30] – Emergency stop [0x1E]	SEL, FUNC	REMAIN
[31] – Empty NV11 Store to Cash Box [0x1F]	SEL	EVENT
[32] – Modify variable MC set [0x20]	SEL, MC	ACK / NAK
[33] - Request RC version [0x21]	SEL	VERSION
[34] - Request RC Count [0x22]	SEL	COUNT
[35] – Modify RC Count [0x23]	SEL, COUNT	ACK / NAK
[36] - Request Current Count [0x24]	SEL	COUNT

Copyright Innovative Technology Ltd 2012

4 COMMUNICATION FORMAT

Baud rate= 9600bps, Data= 8 bit, No parity, Stop bit= 1bit

4.1 COMMAND FORMAT FROM HOST TO NV11:

[Dst] [Len] [CRC (LSB)] [Header] [Data1] ... [DataN] [CRC (MSB)]

Field	Description
Dst	Destination Address, (In case of Bill validator, 40 [0x28])
Len	Number of data bytes in the message
Header	Command
Data	Data attached to Header, specified by Len (1-N)
CRC	[Dst]+[Len]+[Header]+[Data1]+...+Calculated value of CRC16 against Data N

4.2 RESPONSE FORMAT FROM THE NV11 TO HOST:

ACK message

[Dst] [00] [CRC (LSB)] [00] [CRC (MSB)]

NAK message

[Dst] [00] [CRC (LSB)] [05] [CRC (MSB)]

BUSY message

[Dst] [00] [CRC (LSB)] [06] [CRC (MSB)]

Response

[Dst] [Len] [CRC (LSB)] [00] [Data1] ... [DataN] [CRC (MSB)]

Field	Description
Dst	Destination Address, in case of Host 1 [01]
Len	Number of data bytes in the message
Header	Reply message [00]
Data	Data attached to Header, specified by Len (1-N)
CRC	[Dst]+[Len]+[Header]+[Data1]+...+Calculated value of CRC16 against Data N If Encryption setting is enabled, encryption is done against: [CRC(LSB)]+[Header]+[Data1]+...+[DataN]+[CRC(MSB)]

Copyright Innovative Technology Ltd 2012

5 COMMAND DETAILS

The commands for the NV11 are detailed in four sections:

- Routing commands
- Status commands
- Encryption commands
- Compatibility commands.

5.1 STATUS COMMANDS

Commands to get the status of the NV11 (e.g. counters, routing in single denomination operation).

5.1.1 [20] - REQUEST EXPANDED NV11 STATUS [0X14]

Command to retrieve the NV11 status and counters during dispense or empty operation.

Note:
Expands on [29] - Request NV11 status [0x1D], also includes byte (FLAG2) for expanded status reporting.

Send: **[40] [01] [CRC (LSB)] [20] [SEL] [CRC (MSB)]**
Reply: **[01] [07] [CRC (LSB)] [00] [FLAG] [EVENT] [REMAIN] [PAID] [UNPAID] [STORED] [STORE] [FLAG2] [CRC (MSB)]**

Field	Description
FLAG	Bits 0-7 indicate status of NV11: B0 – Note Flag connect 1 = not connected B1 – NV11 Full 1 = full B2 – NV11 Empty 1 = empty B3 – Reserved B4 – Unit enable 1 = disabled B5 – NV11 error occurred 1 = error B6 – Entrance bill remain 1 = remained B7 – Validator error occurred 1 = error
EVENT	Event counter
REMAIN	Number of bills available to payout
PAID	Number of bills paid out
UNPAID	Number of bills not yet paid out
STORED	Number of bills routed from store into stacker
STORE	Number of bills not yet stacked
FLAG2	Bits 0-7 indicate extended status of NV11: B0 – Validator is initialising 1 = Validator not yet initialised B1 – Note Float initialising 1 = Note Float not yet initialised B2 – Empty only 1 = Note Float in 'Empty Only' state B3 – Note inserted during payout 1 = Note detected, NV11 halted B4 – Multi-denomination enabled 1 = Multi-denomination enabled B5 – Reserved B6 – Reserved B7 – Reserved

Copyright Innovative Technology Ltd 2012

5.1.2 [26] – REQUEST TOTAL COUNT [0X1A]

Command to get the total number of stored, dispensed and retrieved bills from the NV11.

Send: **[40] [01] [CRC (LSB)] [26] [SEL] [CRC (MSB)]**
Reply: **[01] [09] [CRC (LSB)] [IN1(LSB)] [IN2] [IN3 (MSB)] [OUT1 (LSB)] [OUT2] [OUT3 (MSB)] [STR1 (LSB)] [STR2] [STR3 (MSB)] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.
IN	Counter for notes routed into Note Float (1-3, Total number of stored bills)
OUT	Counter for notes paid out of Note Float (1-3, Total number of dispensed bills)
STR	Counter for notes routed to cashbox from Note Float (1-3, Total number of retrieved bills)

5.1.3 [21] - CLEAR TOTAL COUNT [0X15]

Command to clear the total number of the stored, dispensed and retrieved bills in the NV11.

Send: **[40] [01] [CRC (LSB)] [21] [SEL] [CRC (MSB)]**
Reply ACK : **[01] [00] [CRC (LSB)] [00] [CRC (MSB)] or**
Reply NAK: **[01] [00] [CRC (LSB)] [05] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.

5.1.4 [33] - REQUEST RC VERSION [0X21]

Command to request recycler version.

Send: **[40] [01] [CRC (LSB)] [33] [SEL] [CRC (MSB)]**
Reply: **[01] [04] [CRC (LSB)] [00] [VERSION] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.
VERSION	4 byte ASCII denoting recycler version.

5.1.5 [34] - REQUEST RC COUNT [0X22]

Command to request (if set) the limit on the maximum number of notes that can be stored in the recycler.

Send: **[40] [01] [CRC (LSB)] [34] [SEL] [CRC (MSB)]**
Reply: **[01] [01] [CRC (LSB)] [00] [COUNT] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.
COUNT	Will return 0, denoting notes will be accepted until unit is full.

Copyright Innovative Technology Ltd 2012

5.1.6 [36] - REQUEST CURRENT COUNT [0X24]

Command to request the number of bills in the recycler.

Send: **[40] [01] [CRC (LSB)] [36] [SEL] [CRC (MSB)]**
Reply: **[01] [02] [CRC (LSB)] [00] [COUNT LSB] [COUNT MSB] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.
COUNT	16bit number denoting the number of bills in the recycler, split over 2 bytes.

5.1.7 [24] - REQUEST VARIABLE SET [0X18]

Request denomination to be recycled by NV11 when in single denomination operation.

Note: Key is returned for compatibility reasons only. The value can be set using command [25].

Send: **[40] [01] [CRC (LSB)] [24] [SEL] [CRC (MSB)]**
Reply: **[01] [02] [CRC (LSB)] [00] [KEY] [MC] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.
KEY	Cipher Key (1-8)
MC	Denomination setting (1-16)

Copyright Innovative Technology Ltd 2012

5.2 ROUTING COMMANDS

These commands govern the route or movement of the bills in the NV11.

5.2.1 [32] – MODIFY VARIABLE MC SET [0X20]

Command to set the bill to recycle in single denomination operation. Respond NAK when the desired denomination data is not available in the NV11 or contains notes of a different denomination.

Note: Ensure Note Float is emptied of notes before sending this command.

Send: **[40] [02] [CRC (LSB)] [32] [SEL] [MC] [CRC (MSB)]**
Reply ACK: **[01] [00] [CRC (LSB)] [00] [CRC (MSB)] or**
Reply NAK: **[01] [00] [CRC (LSB)] [05] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.
MC	Denomination setting (1-16). Set to 0 to route all bills to cashbox.

EXAMPLE

With a EUR dataset containing:

Channel	Denomination
1	€5
2	€10
3	€20
4	€50

To set the unit to store & payout EUR20 notes, you would need to first empty the unit of any existing notes. Then send the command:

40 02 [CRC LSB] 32 00 03 [CRC MSB]

Where:

SEL = 00 as described above
MC = 03 as it is channel 3 to be paid out.

5.2.2 [27] – ENABLE PAYOUT [0X1B]

Command to enable or disable NV11 payout feature. Notes inserted will always be stored if denomination matches that of the NV11, regardless of this setting.

Send: **[40] [01] [CRC (LSB)] [27] [KEY] [CRC (MSB)]**
Reply ACK: **[01] [00] [CRC (LSB)] [00] [CRC (MSB)] or**
Reply NAK: **[01] [00] [CRC (LSB)] [05] [CRC (MSB)]**

Field	Description
KEY	Enable/Disable setting, if value is not 165, unit will not be available for payout: [165] = enable [0xA5] <>[165] = disable <>[0xA5]

Copyright Innovative Technology Ltd 2012

5.2.3 [28] – DISPENSE NV11 BILLS [0X1C]

Command to dispense bills from the NV11.
If ccTalk BNV encryption is being used:
Fill security code fields with '48' [0x30].
If ccTalk BNV encryption is **not** being used:
The security codes are created using a cipher Key. Please contact support@innovative-technology.co.uk for details.

When using multiple denomination recycling, CNT must be set to 1 as only a single bill can be dispensed per command. A NACK will be returned if this condition is not satisfied.

Send: **[40] [0A] [CRC (LSB)] [28] [SEL] [SEC1] [SEC2] [SEC3] [SEC4] [SEC5] [SEC6] [SEC7] [SEC8] [CNT] [CRC (MSB)]**
Reply: **[01] [01] [CRC (LSB)] [00] [EVENT CNT] [CRC (MSB)] or**
Reply: **[01] [00] [CRC (LSB)] [05] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.
SEC	Encryption bytes 1-8 of security code – described above.
CNT	Number of to be dispensed (1-30).
EVENT CNT	Event counter, incremented when notes are moved from the Note Float store.

5.2.4 [30] – EMERGENCY STOP [0X1E]

Command to perform emergency stop while the NV11 is dispensing / storing bills.
Parameter FUNC is reserved and only included for compatibility (default 0).

Send: **[40] [02] [CRC (LSB)] [30] [SEL] [FUNC] [CRC (MSB)]**
Reply: **[01] [01] [CRC (LSB)] [00] [REMAIN] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.
FUNC	Reserved - included for compatibility only.
REMAIN	Number of bills still to be dispensed

5.2.5 [35] – MODIFY RC COUNT [0X23]

Command to set the limit on the maximum number of notes that can be stored in the recycler.

Send: **[40] [01] [CRC (LSB)] [35] [SEL] [COUNT] [CRC (MSB)]**
Reply: **[01] [01] [CRC (LSB)] [00] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.
COUNT	Set to 0 (default) for maximum slots available in the recycler. Any other value will limit the number of notes stored. Excess notes will be routed to the cashbox. Values higher than the slot limit will return NACK.

5.2.6 [31] – EMPTY NV11 STORE TO CASH BOX [0X1F]

Command to store all the bills in the NV11 to the validator store to the cash box.

This command will respond with an NACK and not succeed if NV11 is in 'Empty Only' State. See commands 19 and 20 for more details.

Send: **[40] [01] [CRC (LSB)] [31] [SEL] [CRC (MSB)]**
Reply: **[01] [01] [CRC (LSB)] [00] [EVENT] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.
EVENT	Event counter, (1-255). Counter returns 0 after Reset or Power-On.

5.2.7 [19] – FORCE EMPTY NV11 STORE TO CASH BOX [0X13]

Command to store all the bills in the NV11 to the validator store to the cash box.

This command only for use when in 'Empty Only' State.
See command 20 for details of flag indicating this state.

Send: **[40] [01] [CRC (LSB)] [19] [SEL] [CRC (MSB)]**
Reply: **[01] [01] [CRC (LSB)] [00] [EVENT] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.
EVENT	Event counter, (1-255). Counter returns 0 after Reset or Power-On.

Copyright Innovative Technology Ltd 2012

5.3 CIPHER ENCRYPTION COMMANDS

Cipher encryption commands are not required if using standard ccTalk BNV encryption. These commands setup the key for use with the dispense bills command.

They are also used for DES encryption, see section 5.4.

5.3.1 [22] - PUMP RNG [0X16]

This command 'pumps' the random number generator of the slave device with an 8 byte random number and is part of the dispense encryption algorithm.

Send: **[40] [08] [CRC (LSB)] [22] [RND1] [RND2] [RND3] [RND4] [RND5] [RND6] [RND7] [RND8] [CRC (MSB)]**
Reply: **[01] [00] [CRC (LSB)] [00] [CRC (MSB)]**

Field	Description
RND	Random Value

5.3.2 [23] - REQUEST CIPHER KEY [0X17]

This command requests a cipher key from the slave device and is part of the dispense encryption algorithm.

Send: **[40] [00] [CRC (LSB)] [23] [CRC (MSB)]**
Reply: **[01] [08] [CRC (LSB)] [00] [KEY1] [KEY2] [KEY3] [KEY4] [KEY5] [KEY6] [KEY7] [KEY8] [CRC (MSB)]**

Field	Description
KEY	Cipher Key (1-8)

Copyright Innovative Technology Ltd 2012

5.4 DES ENCRYPTION FOR NOTE PAYOUT

Command	Operation	Response
[22] - Pump RNG [0x16] →		
8 bytes of data		
[RND 1-8] bytes of random data		
		← Ack/Nack
[23] - Request Cipher Key [0x17] →		
		← Cipher Key – 8 bytes
	Decrypt cipher with **DES key** XOR each byte with number of notes to payout Encrypt with **DES key** to create **dispense key**	
[28] – Dispense NV11 bills [0x1C] →		
10 bytes of data:		
[SEL] Select - 0		
[SEC 1-8] Dispense Key		
[NUM] Number of notes		
		← Ack/Nack

During the payout operation, the status and progress of the payout can be checked using command [17] - Request Encrypted Expanded Status [0x11].

Copyright Innovative Technology Ltd 2012

5.4.1 [17] - REQUEST ENCRYPTED EXPANDED STATUS [0X11]

Retrieves the NV11 status and counters during a payout or empty operation.

Note:
This replaces command [18] - Request Encrypted Status [0x12], providing an additional bit in flag 2 representing multi-currency enable.

Send: **[40] [01] [CRC (LSB)] [17] [SEL] [CHAL1] [CHAL2] [CHAL3] [CRC (MSB)]**
Reply: **[01] [07] [CRC (LSB)] [ACK] [DES CRC (LSB)] [CHAL1] [FLAG] [EVENT] [REMAIN] [PAID] [UNPAID] [STORED] [STORE] [FLAG2] [RND1] [RND2] [CHAL2] [RND3] [CHAL3] [DES CRC (MSB)] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.
CHAL 1- 3	3 challenge bytes, which are included for security. These will be returned in the reply and should be checked.
ACK	Acknowledge byte – NACK if correct data is not provided in the command.
DES CRC	CRC-16 of the 14 bytes between [DES CRC (LSB)] and [DES CRC (MSB)]
FLAG	Bits 0-7 indicate status of NV11: B0 - Note Flag connect 1 = not connected B1 - NV11 Full 1 = full B2 – NV11 Empty 1 = empty B3 – Reserved B4 – Unit enable 1 = disabled B5 – NV11 error occurred 1 = error B6 – Entrance bill remain 1 = remained B7 – Validator error occurred 1 = error
EVENT	Event counter
REMAIN	Number of bills available to payout
PAID	Number of bills paid out
UNPAID	Number of bills not yet paid out
STORED	Number of bills routed from store into stacker
STORE	Number of bills not yet stacked
FLAG2	Bits 0-7 indicate extended status of NV11: B0 – Validator is initialising 1 = Validator not yet initialised B1 – Note Float initialising 1 = Note Float not yet initialised B2 – Empty only 1 = Note Float in 'Empty Only' state B3 – Note inserted during payout 1 = Note detected, NV11 halted B4 – Multi-Denomination recycling 1 = mode enabled B5 – Reserved B6 – Reserved B7 – Reserved
RND 1-3	Random bytes of data

Copyright Innovative Technology Ltd 2012

5.5 MULTIPLE DENOMINATION

By default the NV11 will store in the Note Float recycler up to 30 notes of a single denomination ready for payout. For example EUR 10 is recycled and all other notes are stacked to the cashbox below. Firmware version 3.39 released in March 2012 allows multiple denominations of notes to be stored in the Note Float recycler and paid out or stacked. To enable this functionality, additional commands are required and these are described in this section.

5.5.2 OVERVIEW

The notes are still stored in the same way - a Last In First Out mechanism. Therefore in order to payout the correct note, it is possible some notes may need to be stacked to the cashbox, prior to the required note being paid out.

Command 38 will return the number of notes stored and the channel number of the note in each position. An example is shown below:

TX: [40] [0] [195] **[38]** [107]

RX: [1] [5] [57] [0] **[4] [2] [3] [1] [2]** [16]

[04] Indicates a count of 4 notes

[02] 4th accessible note	Channel 2 = EUR 10
[03] 3rd accessible note	Channel 3 = EUR 20
[01] 2nd accessible note	Channel 1 = EUR 5
[02] 1st accessible note	Channel 2 = EUR 10

In order to payout the EUR20 note, the EUR10 and EUR5 notes need to be stored in the cashbox. This is done using command 37 (stack notes) with an option of 2, instructing the NV11 unit to stack 2 notes to the cashbox.

TX: [40] [2] [133] **[37] [0] [2]** [170]

Commands 20 (unencrypted) or 17 (DES encrypted) can be used to check the status of the stack operation.

Once the stack operation is complete, sending command 38 again will return the updated set or stored notes. In this example the following is returned:

TX: [40] [0] [195] **[38]** [107]

RX: [1] [3] [19] [0] **[2] [2] [3]** [147]

[02] Indicates a count of 2 notes

[02] 2nd accessible note	Channel 2 = EUR 10
[03] 1st accessible note	Channel 3 = EUR 20

At this time the dispense command can be used and the EUR 20 note will be paid out.

It is advisable to take into account the time to stack a note when deciding how to pay the full value. It may be that paying out 2 x EUR 5 and 1 x EUR 10 would be more efficient that stacking 3 notes to get to EUR 20.

Note:
When using multiple denomination mode, only one bill can be dispensed each time command [28] – Dispense NV11 bills [0x1C] is used. Using a count > 1 in command 28 will return a NACK.

Copyright Innovative Technology Ltd 2012

5.5.3 [37] – STACK BILLS [0X25]

Stacks one or more bills from the recycler to the cashbox.

Send: **[40] [01] [CRC (LSB)] [37] [SEL] [COUNT] [CRC (MSB)]**
Reply: **[01] [01] [CRC (LSB)] [00] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.
COUNT	Number of bills to send to the cashbox. Values higher than the number of notes stored will return NACK.

5.5.4 [38] – GET NOTE POSITIONS [0X26]

Returns the count of notes stored in the recycler and the channel of each stored note.

Send: **[40] [01] [CRC (LSB)] [38] [SEL] [CRC (MSB)]**
Reply: **[01] [01] [CRC (LSB)] [00] [COUNT] [NOTE N] ... [NOTE1] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.
COUNT	Number of bills stored in recycler.
NOTE N	The channel number of the N^{th} accessible note in the recycler, where N = COUNT.
NOTE 1	The channel number of first accessible note stored in the recycler.

5.5.5 [39] – SET NOTE ROUTES [0X27]

Enables multi-denomination bill recycling and determines which bills will be routed to the recycler, and which bills to the cashbox. The routing is set through a bit array made up of 2 bytes:

Each bit represents a channel, bit 0 is LSB (least significant bit):

[mask 1]	[mask 2]
Bit 0 = bill 1	Bit 0 = bill 9
...	...
Bit 7 = bill 8	Bit 7 = bill 16

The route of each channel is represented by the value of each bit:
0 = bill stacked
1 = bill recycled

Send: **[40] [01] [CRC (LSB)] [39] [SEL] [MASK1] [MASK2] [CRC (MSB)]**
Reply: **[01] [01] [CRC (LSB)] [00] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.
MASK 1	Bit array defining routing of channels 1 - 8
MASK 2	Bit array defining routing of channels 7 - 16

Copyright Innovative Technology Ltd 2012

5.5.6 [40] – GET NOTE ROUTES [0X28]

Indicates the routing of bills in multi denomination recycling. The routing is indicated through a bit array made up of 2 bytes as detailed above.

The route of each channel is represented by the value of each bit:
0 = bill stacked
1 = bill recycled

Send: **[40] [01] [CRC (LSB)] [40] [SEL] [CRC (MSB)]**
Reply: **[01] [01] [CRC (LSB)] [00] [MASK1] [MASK2] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.
MASK 1	Bit array indicating routing of channels 1 - 8
MASK 2	Bit array indicating routing of channels 7 - 16

Copyright Innovative Technology Ltd 2012

6 COMPATIBILITY COMMANDS

Commands only included for compatibility with the specification. These have no effect on the NV11 operation.

6.1.1 [25] – MODIFY VARIABLE SET [0X19]

Supported for compatibility reasons. Does not affect setup of NV11.
Data field is ignored.

6.1.2 [29] - REQUEST NV11 STATUS [0X1D]

Command is redundant due to addition of command 20, however is included for compatibility reasons.

Retrieves the NV11 status and counters.

Send: **[40] [01] [CRC (LSB)] [29] [SEL] [CRC (MSB)]**
Reply: **[01] [07] [CRC (LSB)] [00] [FLAG] [EVENT] [REMAIN] [PAID] [UNPAID] [STORED] [STORE] [CRC (MSB)]**

Field	Description
FLAG	Bits 0-7 indicate status of NV11: B0 - Note Flag connect 1 = not connected B1 - NV11 Full 1 = full B2 – NV11 Empty 1 = empty B3 – Reserved B4 – Unit enable 1 = disabled B5 – NV11 error occurred 1 = error B6 – Entrance bill remain 1 = remained B7 – Validator error occurred 1 = error
EVENT	Event counter
REMAIN	Number of bills available to payout
PAID	Number of bills paid out
UNPAID	Number of bills not yet paid out
STORED	Number of bills routed from store into stacker
STORE	Number of bills not yet stacked

Copyright Innovative Technology Ltd 2012

6.1.3 [18] - REQUEST ENCRYPTED STATUS [0X12]

Command is redundant due to addition of command 17, however is included for compatibility reasons.

Retrieves the NV11 status and counters during a payout or empty operation.

Send: **[40] [01] [CRC (LSB)] [18] [SEL] [CHAL1] [CHAL2] [CHAL3] [CRC (MSB)]**
Reply: **[01] [07] [CRC (LSB)] [ACK] [DES CRC (LSB)] [CHAL1] [FLAG] [EVENT] [REMAIN] [PAID] [UNPAID] [STORED] [STORE] [RND1] [RND2] [RND3] [CHAL2] [RND4] [CHAL3] [DES CRC (MSB)] [CRC (MSB)]**

Field	Description
SEL	Recycler selection - set to 0.
CHAL 1- 3	3 challenge bytes, which are included for security. These will be returned in the reply and should be checked.
ACK	Acknowledge byte – NACK if correct data is not provided in the command.
DES CRC	CRC-16 of the 14 bytes between [DES CRC (LSB)] and [DES CRC (MSB)]
FLAG	Bits 0-7 indicate status of NV11: B0 - Note Flag connect 1 = not connected B1 - NV11 Full 1 = full B2 – NV11 Empty 1 = empty B3 – Reserved B4 – Unit enable 1 = disabled B5 – NV11 error occurred 1 = error B6 – Entrance bill remain 1 = remained B7 – Validator error occurred 1 = error
EVENT	Event counter
REMAIN	Number of bills available to payout
PAID	Number of bills paid out
UNPAID	Number of bills not yet paid out
STORED	Number of bills routed from store into stacker
STORE	Number of bills not yet stacked
RND 1-4	Random bytes of data

Copyright Innovative Technology Ltd 2012

7 GLOSSARY

7.1 ABRIEVIATIONS USE IN SPECIFICATION

CNT	Count
RNG	Random Number generated
MC	Denomination Setting
ACK	Acknowledge [00]
NAK	Reject [05]
SEL	Select denomination (where applicable)
DST	Destination
CRC	Cyclic Redundancy Check – see full ccTalk manual.

7.2 'EMPTY ONLY' STATE

The NV11 unit may at times not accept further notes into the recycler regardless of the routing setup; these notes are instead stacked to the cashbox. Often this will only be the case if the limit of notes to store has been reached. The other case this may happen is if the recycler has entered a note out of expected position or other discrepancies that could lead to a failed or incorrect payout.

Copyright Innovative Technology Ltd 2012