

MTX Node Ring Notes

Revision: 29 March 2015

Acknowledgement

This document could not have been produced without the aid of the NODE documentation and NODE ROM disassembly provided by Tony Brewer.

Disclaimer

Some of the information below is based upon the authors interpretation of the NODE ROM listing. No assurances as to the accuracy of this document are given.

Node Variables

Name	Address	Description
		Space between (STKLIM) and (LSTBOT) for mail
		14*(NRING) bytes of space for node names Node names list grows downwards in memory, with IDSELF & ASCIND (own name and ID) forming the top entry
IDSELF	F000	Name of node
ASCIND	F00C	Numeric node identifier as hex string
RINDEX	F00E	Numeric node identifier as a byte value
RNGFLG	F00F	03 = In operation
RCVFLG	F010	Receiver status: 0 = Not in packet 1 = Start of packet received 2 = Data packet 3 = Data packet, >=5 characters, first byte of ID matches 4 = Data packet, not addressed to node 5 = Data packet >=5 characters, addressed to node 6 = Control or program packet
TFLAG	F011	Transmit status
TMRFLG	F012	Timer flag
PTYPE	F013	Return packet type
SYNFLG	F014	Packet syntax flag: 0 = No error 2 = Declared packet length does not match received length 4 = Checksum error 8 = Packet type 5, length not 3 bytes
RCIND	F015	receiver buffer pointer
RPOINT	F017	no. of chars. in receiver buffer

Name	Address	Description
RCERR	F018	special receive condition flag
FRMERR	F019	packet framing error flag: Non zero if start of packet received part way.
TINDEX	F01A	transmitter buffer pointer
MRKFLG	F01C	transmitter mark flag
TCOUNT	F01D	counter for timer
TCNST	F01F	ring time constant
ABTFLG	F021	abort flag
BRCFLG	F022	Broadcast flag (1 = Broadcast data received, 0 = Unicast data received)
SUSPFLG	F023	set while ring suspended
PROGFLG	F024	non-zero during program loading
RNGERR	F025	set during recovery from ring error
RCMMD	F026	ring executive command byte, set by NI
RETFLG	F027	return packet code
MBXFLG	F028	Mail box in use flag
MAILC2	F029	Copy of MAILCNT. Tested during print spooling.
CTBFLG	F02A	set while NI is writing to CTBUF
TKNFLG	F02B	set while node has token
TKNHLD	F02C	hold token if set
TCOMMAND	F02D	transmitted data packet token
RNGSTAT	F02E	ring status for NODE STAT
NRING	F02F	number of nodes in ring
NSEND	F030	number of sender nodes
NDTYP	F031	Node type: 2 = Master node
ACCEPT	F032	data accept flags
ENABLE	F033	Command enable flags. Set to FF if master packet received.
CTBCNT	F034	no. of entries in CTB
CRBCNT	F035	no. of entries in CRB
RSENDER	F036	last data received sender no.
RTYPE	F037	last data received type
ERRSTAT	F038	Error trapping status
ERRNUM	F039	Error number [copy of ERRFLG]
GSBFLG	F03A	GOSUB flag
ADBAUD	F03B	channel A default baud rate
CTCAD	F03C	channel A baud rate
MAILCNT	F03D	number of entries in mail box

Name	Address	Description
IRSTAT	F03E	Copy of INSRCV for NODE STAT
NNSTAT	F03F	Copy of NEWNAME for NODE STAT
PKTCNT	F040	Non-token packet counter
TKNCNT	F041	Token counter
INSRCV	F042	instruction received flag
NEWNAME	F043	new name flag
JTYPE	F044	received packet type
ENTER	F045	enter ring on next token, if set
ERRF2	F046	An error code??
ASCIN2	F047	Copy of ASCIND – Hex encoded node ID
CTCBD	F049	Time constant for CTC channel 2
DBFMT	F04A	DART Channel B configuration Bits 15-14: Bits per character (0x0000 = 5, 0x8000 = 6, 0x4000 = 7, 0xC000 = 8) Bits 3-2: Stop bits (0x0040 = 1, 0x0080 = 1.5, 0x00C0 = 2) Bits 1-0: Parity (0x0000 = None, 0x0001 = Odd, 0x0003 = Even)
NDYRSV	F04C	Number of bytes reserved for NODDY page
ERRTRAP	F04E	zero if error trap off
ERRFLG	F04F	Error message number
ERRLIN	F050	line number to GOTO on error
NLIST	F052	first byte in identifier space
LSTBAS	F054	bottom of list space
SRCND	F057	source node for data packet
PKTFLG	F058	data packet no. flag
DSPCNT	F059	offset for display buffer
SCRNFLG	F05A	flag for display screen
TIMEFLG	F05B	flag for time on display screen
RUNFLG	F05C	flag for auto-run after program load
NOENT	F05D	Non zero value prevents node from entering ring on receipt of enter (0x34) packet
OLDCALC	F064	stores (CALCST)
OSTKLM	F066	Holds previous value of STKLIM when ring is suspended
CRBBEG	F068	Pointer to start of Command Receive Buffer
CTBPSV	F06A	Saves CTBPTR while receiving a program
CTBPTR	F06C	next free byte in Command Transmitter Buffer
CRBPTR	F06E	next free byte in Command Receiver Buffer
TXPEND	F070	Pointer to pending transmission command

Name	Address	Description
MBXBOT	F072	pointer to first byte below mailbox
MBPPTR	F074	Pointer to character in mailbox to print
MBXTOP	F076	Pointer to top of mailbox
MBTOFS	F078	Offset (from (STKLIM)) of top of mailbox
MBPOFS	F07A	Offset (from (STKLIM)) in mailbox of character to print
MBPCNT	F07C	Number of bytes in mailbox to print
PRNTFLG	F07E	mail print flag
PRNDVC	F07F	printer device flag
QDELFLG	F080	Delete mail flag
CHAR	F081	character to be printed
	F082 -F091	This space appears to be unused
		The following user jumps are all initialised with RET, but can be modified to customise NODE operation
NODEUJ0	F092	Character received. Character in A
NODEUJ1	F095	Start of packet decode
NODEUJ2	F098	Packet processed
NODEUJ3	F09B	About to transmit character. Character in A
NODEUJ4	F09E	Unicast data received?
NODEUJ5	F0A1	Processing Transmit Control Blocks
NODEUJ6	F0A4	User timer routine. A replacement for USERINT, which is hooked by NODE ROM
NODEUJ7	F0A7	User keyboard routine. A replacement for USERIO, which is hooked by the NODE ROM
NODEUJ8	F0AA	Process NODE command. DE = Start of line
NODEUJ9	F0AD	About to display (HL) on NODE status line
NODEUJA	F0B0	Called by NODE DISC
NODEUJB	F0B3	Called by NODE USER
NODEUJC	F0B6	Unused ?
NODEUJD	F0B9	Unused ?
NODEUJE	F0BC	Unused ?
	F0BF	Start of NODE code in RAM
	F0EF	Relative offset of a JR nstruction. Changed from #00 to #03 when loading a program
NODEPG	F189	NODE ROM page (modifies data value of a LD A,nn instruction)
	F2FB	End of NODE code in RAM
DSPINI	F2FD	Display line initialised flag
IDSLFA	F2FE	End of node names list

Name	Address	Description
CTBUF	F300	Command Transmitter Buffer (256 bytes)
CRBUF	F400	Command Receive Buffer (512 bytes)
TRBUF	F600	Transmit buffer
RCVBUF	F700	Receive buffer
DSPBUF	F800	Node variable: Node message buffer
DSPTXT	F802	Start of text in message buffer
DSPTIM	F820	Location of time in message buffer

Node Names List

The node names list grows down in memory from IDSELF, which forms the first entry. NLIST points to the base of the list. Each entry is 14 bytes and consists of:

Offset	Size	Contents
0	12	Name of the node (ASCII text)
12	2	Node ID (hex encoded)

Transmit Command Block

This specifies data to be assembled for transmission. CTBPTR or TXPEND gives location of this data

Offset	Size	Contents
0	1	Length of block
1	1	Destination ID
2	*	Data to be included in packet header (TCB length - 8)
	3	Address and page of data to send
	3	Length of data to send

Packet Format

Control Packets

Offset	Length	Content	Description
0	1	0x02	Start of packet
1	1	Packet type	0x30 ("0") = Suspend ring 0x31 ("1") = Initiate packet 0x32 ("2") = Master packet 0x33 ("3") = Identifier packet (gives name of node) 0x34 ("4") = Enter packet (join all nodes to ring)

Offset	Length	Content	Description
			0x35 (“5”) = Null packet (next character is EOP) 0x37 (“7”) = Ring abort / ring error 0x38 (“8”) = Master abort / master error
4	12	Source Name	Name of sending node
16	2	Source ID	ID of sending node (hex encoded)
18	2	Hop count?	Number of nodes in the ring?
*	4	Checksum	Hex encoded 16 bit value = minus sum of the packet contents, including packet type
*	1	0x1A	End of packet (EOP)

Program & Data Packets

Offset	Length	Content	Description
0	1	0x02	Start of packet
1	1	Packet type	0x39 (“9”) = Broadcast (to all nodes) 0x41 (“A”) = Unicast (to specific node)
2	2	Packet length	Hex encoded
4	2	Destination ID	Numeric identifier in hex
6	2	Source ID	Numeric identifier in hex
8	4	Packet Number	High bit is last packet flag
12	*	Data	Hex encoded binary data
*	4	Checksum	Hex encoded 16 bit value = minus sum of the binary length and binary data
*	1	0x1A	End of packet

Data for CALL command

Offset	Length	Content	Description
0	1	0x8E	CALL command
1	2	Address	Address to call
3	1	Page	ROM/RAM page to call
4	2	Parameter	Value to pass in HL
6	1	0x01	Length of dummy data
7	1	Dummy	Dummy data required by TCB

Data for EXT command

Offset	Length	Content	Description
0	1	0x8F	EXT command
1	1	Accept	The data accept bits
2	1	Enable	The command enable bits

Offset	Length	Content	Description
3	1	0x01	Length of dummy data
4	1	Dummy	Dummy data required by TCB

Data for FLAG command

Offset	Length	Content	Description
0	1	0x90	FLAG command
1	1	Flag	The flag to set
2	1	0x01	Length of dummy data
3	1	Dummy	Dummy data required by TCB

Data for MESSAGE command

Offset	Length	Content	Description
0	1	0x91	MESSAGE command
1	1	Length	Length of message
2	*	Message	Text of message (max. 30 characters)

Data for POST command

Offset	Length	Content	Description
0	1	0x92	SNODDY command or response to RCV request
1	8	Source	Source variable name (type encoded – first letter at end, with high bit set), followed by a space
9	2	Length	Length of NODDY page
11	1	Block length	Length of data in this packet (max. 64 bytes)
12	*	Data	Block of NODDY data (multiple packets sent if necessary to transfer all data)

Data for MSEND command

Offset	Length	Content	Description
0	1	0x93	MSEND command or response to MRCV request
1	1	Mode	0x00 = MSEND 0x01 = Response to MRCV
2	2	Address	Source address
4	1	Page	ROM/RAM page of source
5	2	Address	Destination address
7	1	Source	Destination page
8	2	Length	Length of memory to transfer
10	1	Block length	Length of data in this packet (max. 64 bytes)
11	1	Data	Block of data (multiple packets sent if necessary to transfer all data)

Data for SEND command

Offset	Length	Content	Description
0	1	0x94	SEND command or response to RCV request
1	1	Mode	0x00 = SEND 0x01 = Response to RCV
2	8	Source	Source variable name (type encoded – first letter at end, with high bit set), followed by a space
10	8	Destination	Destination variable name (type encoded – first letter at end, with high bit set), followed by a space
18	2	Length	Length of source variable
20	1	Block length	Length of data in this packet (max. 64 bytes)
21	*	Data	Block of data (multiple packets sent if necessary to transfer all data)

Data for SNODDY command

Offset	Length	Content	Description
0	1	0x95	SNODDY command or response to RCV request
1	8	Name	Name of NODDY page. Maximum of 8 characters. High bit set on last character of name, and followed by space.
9	2	Length	Length of NODDY page
11	1	Block length	Length of data in this packet (max. 64 bytes)
12	*	Data	Block of NODDY data (multiple packets sent if necessary to transfer all data)

Data for PROGRAM command (first part)

Offset	Length	Content	Description
0	1	0x96	PROGRAM command (System Variables)
1	1	Flag	0xB4 = RUN 0x9D = LLIST 0x00 = None
2	1	LSTPG	Last page of memory
3	2	0xFA7B	Start address of system variables sent (VARNAM)
5	1	LSTPG	Last page of memory
6	3	Length	Length of system variables.
9	1	Block length	Length of data in this packet (max. 64 bytes)
10		Data	System variables between VARNAM (0xFA7B) and (SYSTOP) (typically (0xFB4B) (multiple packets sent if necessary to transfer all data)

Data for PROGRAM command (second part)

Offset	Length	Content	Description
0	1	0x97	PROGRAM command (Program VA)
1	1	Flag	0xB4 = RUN 0x9D = LLIST 0x00 = None
2	3	0x00, 0x00, 0x00	Start VA (Zero)
5	3	Length	Length of program & array data
8	1	Block length	Length of data in this packet (max. 64 bytes)
9	*	Data	Program virtual memory (program & array data) (multiple packets sent if necessary to transfer all data)

Data for PROGRAM command (third part)

Offset	Length	Content	Description
0	1	0x98	PROGRAM command (Program variables)
1	1	Flag	0xB4 = RUN 0x9D = LLIST 0x00 = None
2	2	VARNAM	Address of start of variable names table
4	1	LSTPG	Last page of memory
5	3	Length	Length of variable name and data tabs
8	2	Block length	Length of data in this packet (max. 64 bytes)
10	*	Data	Variable name and value tables (multiple packets sent if necessary to transfer all data)

Data for MRCV command

Offset	Length	Content	Description
0	1	0x99	MRCV command
2	2	Address	Source address
4	1	Page	ROM/RAM page of source
5	2	Address	Destination address
7	1	Source	Destination page
8	2	Length	Length of memory to transfer
10	1	0x01	Length of dummy data
11	1	Dummy	Dummy data required by TCB

Data for Acknowledge packet

Offset	Length	Content	Description
0	1	0xA9	Acknowledge data receipt
2	1	0x01	Length of dummy data

Offset	Length	Content	Description
3	1	Status	Number of status message to display

Example Ring Traffic

Ring Initiation (Two nodes over named pipes)

At node "Alpha"

Type in: node name,"alpha"

Transmit 0 <Ctrl+Z>, 0 NULL packets, 0 Out of packet characters

Transmit Ring Packet:

119ALPHA 0101FC5D

Receive 0 <Ctrl+Z>, 0 NULL packets, 0 Out of packet characters

Receive Ring Packet:

119ALPHA 0102FC5C

Transmit 61 <Ctrl+Z>, 0 NULL packets, 0 Out of packet characters

Transmit Ring Packet:

219ALPHA 0201FC5B

Receive 0 <Ctrl+Z>, 0 NULL packets, 0 Out of packet characters

Receive Ring Packet:

219ALPHA 0202FC5A

Receive 17687 <Ctrl+Z>, 527 NULL packets, 0 Out of packet characters

Receive Ring Packet:

319BETA 0201FC84

Transmit 19675 <Ctrl+Z>, 528 NULL packets, 0 Out of packet characters

Transmit Ring Packet:

319BETA 0202FC83

At node "Beta"

Receive 0 <Ctrl+Z>, 0 NULL packets, 0 Out of packet characters

Receive Ring Packet:

119ALPHA 0101FC5D

Transmit 0 <Ctrl+Z>, 0 NULL packets, 0 Out of packet characters

Transmit Ring Packet:

119ALPHA 0102FC5C

Receive 61 <Ctrl+Z>, 0 NULL packets, 0 Out of packet characters

Receive Ring Packet:

219ALPHA 0201FC5B

Transmit 0 <Ctrl+Z>, 0 NULL packets, 0 Out of packet characters

Transmit Ring Packet:

219ALPHA 0202FC5A

Type in: node name,"beta"

Transmit 17687 <Ctrl+Z>, 527 NULL packets, 0 Out of packet characters

Transmit Ring Packet:

319BETA 0201FC84
Receive 19675 <Ctrl+Z>, 528 NULL packets, 0 Out of packet characters
Receive Ring Packet:
319BETA 0202FC83

Testing Commands (Two nodes over named pipes)

Type in: node name,"alpha"
Transmit 0 <Ctrl+Z>, 0 NULL packets, 0 Out of packet characters
Transmit Ring Packet:
119ALPHA 0101FC5D
Receive 0 <Ctrl+Z>, 0 NULL packets, 0 Out of packet characters
Receive Ring Packet:
119ALPHA 0102FC5C
Transmit 139 <Ctrl+Z>, 0 NULL packets, 0 Out of packet characters
Transmit Ring Packet:
219ALPHA 0201FC5B
Receive 0 <Ctrl+Z>, 0 NULL packets, 0 Out of packet characters
Receive Ring Packet:
219ALPHA 0202FC5A
Type in: node enter
Transmit 11741 <Ctrl+Z>, 361 NULL packets, 0 Out of packet characters
Transmit Ring Packet:
419ALPHA 0101FC5A
Receive 19043 <Ctrl+Z>, 361 NULL packets, 0 Out of packet characters
Receive Ring Packet:
419ALPHA 0102FC59
Receive 0 <Ctrl+Z>, 0 NULL packets, 0 Out of packet characters
Receive Ring Packet:
3192 0201FD0E
Transmit 88 <Ctrl+Z>, 1 NULL packets, 0 Out of packet characters
Transmit Ring Packet:
3192 0202FD0D
Receive 13336 <Ctrl+Z>, 325 NULL packets, 0 Out of packet characters
Receive Ring Packet:
319BETA 0201FC84
Transmit 27089 <Ctrl+Z>, 325 NULL packets, 0 Out of packet characters
Transmit Ring Packet:
319BETA 0202FC83
Type in: node message,"beta","hello"
Transmit 37642 <Ctrl+Z>, 736 NULL packets, 0 Out of packet characters
Transmit Ring Packet:
A1F02010180910548454C4C4FFD53
Receive 32920 <Ctrl+Z>, 737 NULL packets, 0 Out of packet characters
Receive Ring Packet:
A1701020100A9011BFF20
Type in: let m\$="you have mail"
Type in: node post,"beta",m\$
Transmit 112232 <Ctrl+Z>, 1382 NULL packets, 0 Out of packet characters
Transmit Ring Packet:
A4302010180928D2020202020200D000D594F552048415645204D41494CF99

C

Receive 32481 <Ctrl+Z>, 1382 NULL packets, 0 Out of packet characters

Receive Ring Packet:

A1701020100A9011DFF1E

Type in: 10 print "Testing"

Type in: node program,"beta",run

Transmit 119988 <Ctrl+Z>, 1872 NULL packets, 0 Out of packet characters

Transmit Ring Packet:

AA50201010096B4017BFA01D000004000C001C001C001C04BFB00000000000C9C
900000001FF80FFD74BFB48FDC367F1C37FF1C90000C900000F40000F4000004
00F400000000000000000000000000000000000000000000E8BD

Receive 44796 <Ctrl+Z>, 1872 NULL packets, 0 Out of packet characters

Receive Ring Packet:

A1701020100A90100FF3B

Transmit 905 <Ctrl+Z>, 1 NULL packets, 0 Out of packet characters

Transmit Ring Packet:

AA50201020096B4017BFA01D0000040000000000000000000000000000000000
00F0000004000010000000F40000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000FAE6

Receive 0 <Ctrl+Z>, 1 NULL packets, 0 Out of packet characters

Receive Ring Packet:

A1701020100A90100FF3B

Transmit 599 <Ctrl+Z>, 1 NULL packets, 0 Out of packet characters

Transmit Ring Packet:

AA50201030096B4017BFA01D0000040000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000FB84

Receive 0 <Ctrl+Z>, 1 NULL packets, 0 Out of packet characters

Receive Ring Packet:

A1701020100A90100FF3B

Transmit 532 <Ctrl+Z>, 1 NULL packets, 0 Out of packet characters

Transmit Ring Packet:

A450201048096B4017BFA01D0000010000000000000D8FA0000000040800000F
901

Receive 0 <Ctrl+Z>, 1 NULL packets, 0 Out of packet characters

Receive Ring Packet:

A1701020100A90100FF3B

Transmit 569 <Ctrl+Z>, 1 NULL packets, 0 Out of packet characters

Transmit Ring Packet:

A450201018097B401000000100000100F000A00902254455354494E4722FF40F
981

Receive 0 <Ctrl+Z>, 1 NULL packets, 0 Out of packet characters

Receive Ring Packet:

A1701020100A90100FF3B

Transmit 699 <Ctrl+Z>, 1 NULL packets, 0 Out of packet characters

Transmit Ring Packet:

A270201018098B40100C00101000001FFFC46

Receive 0 <Ctrl+Z>, 1 NULL packets, 0 Out of packet characters

Receive Ring Packet:

A1701020100A90126FF15

Type in: node suspend

Transmit 41058 <Ctrl+Z>, 778 NULL packets, 0 Out of packet characters

Transmit Ring Packet:

019ALPHA 0101FC5E

Receive 18599 <Ctrl+Z>, 778 NULL packets, 0 Out of packet characters

Receive Ring Packet:

019ALPHA 0102FC5D

Type in: node cont

Type in: node ext,"beta",255,255

Transmit 77103 <Ctrl+Z>, 897 NULL packets, 0 Out of packet characters

Transmit Ring Packet:

A1B020101808FFFFFF0100FCD3

Receive 36865 <Ctrl+Z>, 897 NULL packets, 0 Out of packet characters

Receive Ring Packet:

A1701020100A90119FF22