Citect for Windows, Version 6.xx, 7.xx, 8.xx

JControl driver, User information

Contents

1. User information

1.1 Device Application notes

1.1.1 Device Manufacturer

	Detail
Manufacturer	Johnson Controls
Device name	JControl
Communications method	Serial RS232 direct or RS485 multidrop via converter IU-9100. The driver acts as master.

1.1.2 Device Definition

Johnson Controls has informed that the protocol is exactly the same for all units below. The only difference is that the item addresses for similar signals are different between the models.

Models
DX-9100
DC-9100
XT/XP-9100
TC-9100
SC-9100
DR-9101

1.2 Communications/Hardware Configuration

The serial method of communication to the Johnson family of intelligent digital and analogue input/output controllers, uses the JCONTROL protocol. Using this method you can connect to single controllers or to multiple controllers as in the following diagrams:
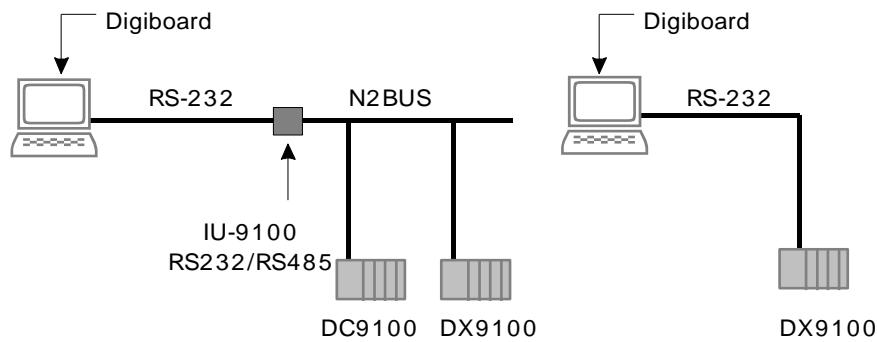
1.2.1 Connection details for download/upload of DX-9100
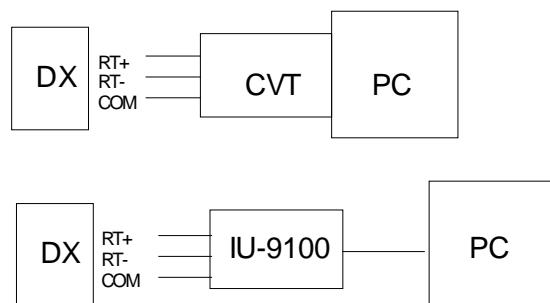
1.2.2 Wiring Diagram IU-9100 to PC Connection
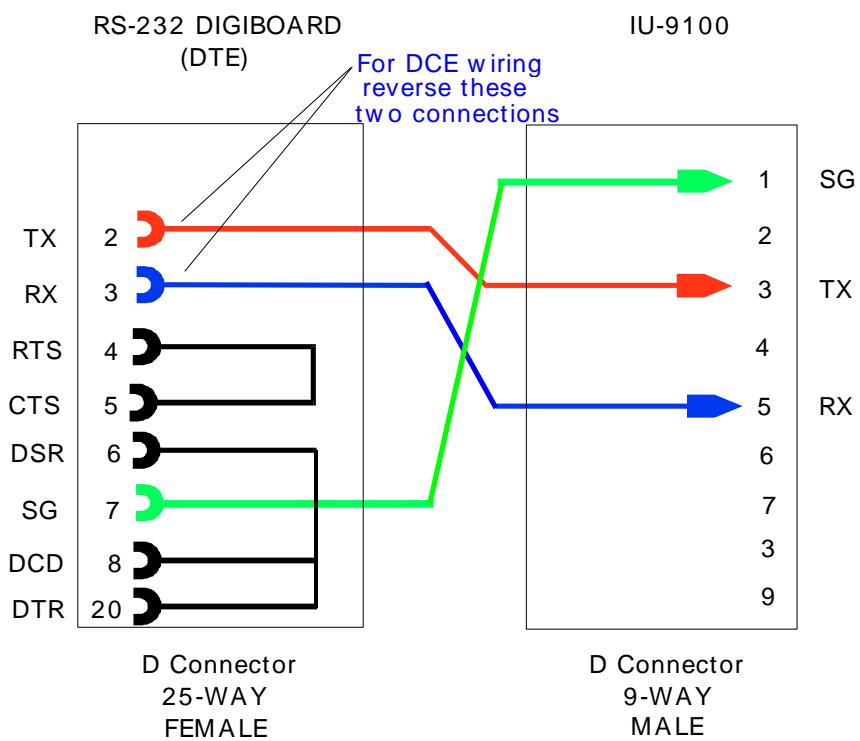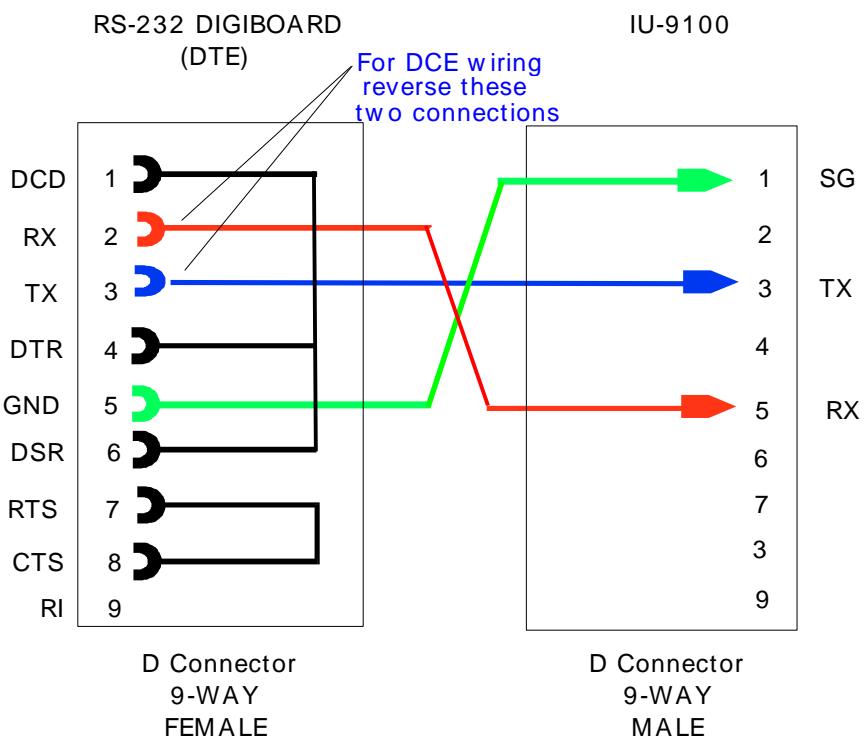
1.2.3 Wiring Diagram Direct Download DX9100

RS-232 DIGIBOARD
(DTE)

For DCE wiring
reverse these
two connections

DX9100

DCD	1
RX	2
TX	3
DTR	4
GND	5
DSR	6
RTS	7
CTS	8
RI	9

1	
2	RX
3	TX
4	
5	GND
6	
7	
3	
9	

D Connector
9-WAY
FEMALE

D Connector
9-WAY
FEMALE

RS-232 DIGIBOARD
(DTE)

For DCE wiring
reverse these
two connections

DX9100

TX	2
RX	3
RTS	4
CTS	5
DSR	6
SG	7
DCD	8
DTR	20

1	
2	RX
3	TX
4	
5	SG
6	
7	
3	
9	

D Connector
25-WAY
FEMALE

D Connector
9-WAY
FEMALE
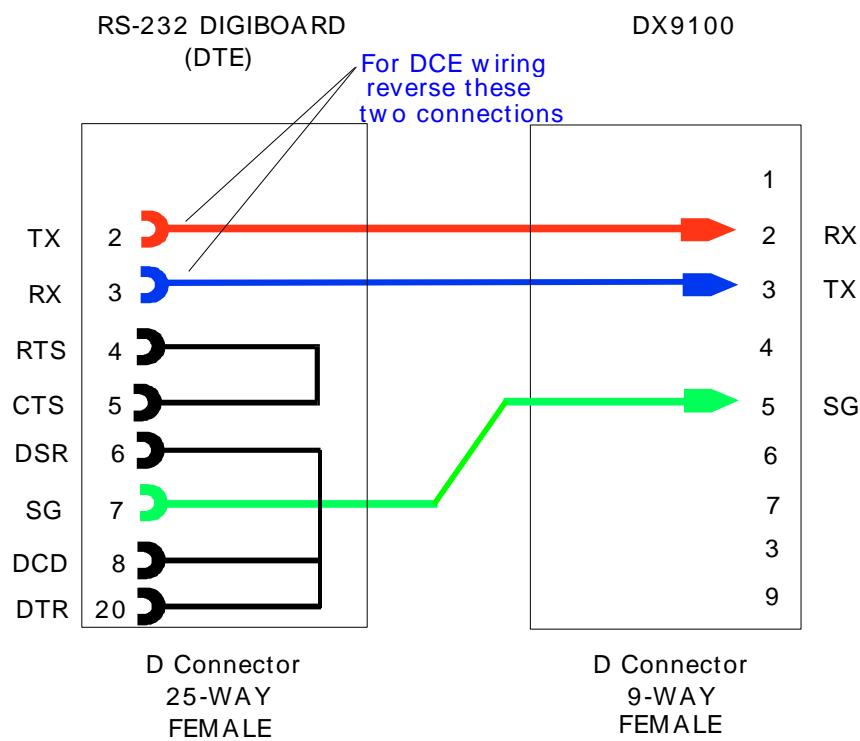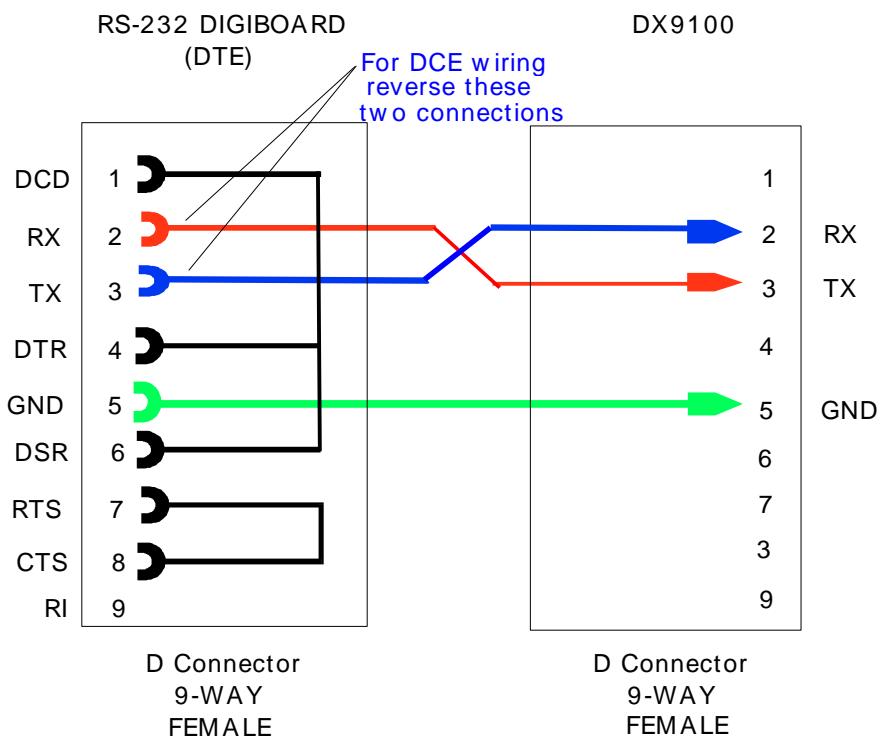
1.2.4 I/O Device Settings

The Johnson factory sets all communications parameters in the units.

1.3 Special Requirements

The JCONTROL driver for Johnson Control equipment deliveries by begcomm Communication AB, Brunnehagen 109, 417 47 Göteborg, Sweden. All copyrights and licenses belongs to begcomm Communication AB.

1.4 Maximum Request Length

The driver works with single items. The item structure especially in the DX series is not good for block readings.

1.5 Reference: Communications forms
1.5.1 Serial communication

1.5.1.1 Boards form

Field	Default	Allowable values
Board Name	This field is user defined.	
Board Type	COMX	
Address	0	
I/O Port	BLANK	
Interrupt	BLANK	
Special Opt	BLANK	
Comment	This field is user defined and is not used by the driver.	

1.5.1.2 Ports form

Field	Default	Allowable values
Port Name	This field is user defined.	
Port number		
Board name	Refers to the board previously defined in 'boards' form.	
Baud rate	9600	
Data bits	8	
Stop bits	1	
Parity	NONE	
Special Opt	BLANK	
Comment	This field is user defined and is not used by the driver.	

1.5.2 Ethernet communication

1.5.2.1 Boards form

Field	Default	Allowable values
Board Name	This field is user defined.	
Board Type	TCPIP	
Address	0	
I/O Port	BLANK	
Interrupt	BLANK	
Special Opt	BLANK	
Comment	This field is user defined and is not used by the driver.	

1.5.2.2 Ports form for tcp

Field	Default	Allowable values
Port Name	This field is user defined.	
Port number		
Board name	Refers to the board previously defined in 'boards' form.	
Baud rate	BLANK	Not used by the driver
Data bits	BLANK	Not used by the driver
Stop bits	BLANK	Not used by the driver
Parity	BLANK	Not used by the driver
Special Opt	-i192.168.100.136 –p10001 -t	
Comment	This field is user defined and is not used by the driver.	

1.5.2.3 Ports form for udp

Field	Default	Allowable values
Port Name	This field is user defined.	
Port number		
Board name	Refers to the board previously defined in 'boards' form.	
Baud rate	BLANK	Not used by the driver
Data bits	BLANK	Not used by the driver
Stop bits	BLANK	Not used by the driver
Parity	BLANK	Not used by the driver
Special Opt	-i192.168.100.136 –p10001 -u	
Comment	This field is user defined and is not used by the driver.	

1.5.3 I/O Devices form

Field	Default	Allowable values
Name	This field is user defined, and is not used by the driver.	
Number	Must be unique.	
Address	241	The range is between 0 and 255
Protocol	JCONTROL	
Port name	Refers to the port previously defined in 'ports' form.	
Comment	This field is user defined and is not used by the driver.	

1.6 Reference

1.6.1 Data types JControl

IO Device Type	Citect data format	Citect data types	Description/Special Usage/Limitations/ Valid Ranges
Single Item	SIN*x*	REAL	Read / Write. Johnson Control special floating point number (16 bit).
Single Item	SIB*x*	BYTE	Read / Write.
Single Item	SIB*x.y*	DIGITAL	Read / Write.
Single Item	SIW*x*	UINT	Read / Write.
Single Item	SIW*x.z*	DIGITAL	Read / Write.
Single Item	SIL*x*	LONG	Read / Write. Allowed value is maximum 2,147,483,647
Real Time Clock	RTC*t*	BCD	Read / Write.
Daylight Savings	DS*d*	BCD	Read / Write.
Exception Days	ED*e*	BCD	Read / Write.
Time Schedule	TS*m.w*	BCD	Read / Write.
Time Schedule	TS*m.w*	UINT	Read / Write.

Where:
- *x*   Item number, 0 - FFFF
- *y*   Bit number, 1 - 8
- *z*   Bit number, 1 - 16
- *t*   Item number, 0 - 7
- *d*   Item number, 0 - 1
- *e*   Item number, 0 - 3B
- *m*   Time schedule module, 0 - 7
- *w*   Item in time schedule module, 0 - 1C

1.6.2 Examples data types JControl

1.6.2.1 Single Item

Data Type   BYTE
Address   SIB0
Comment   Item for Device Model (All models)

Data Type   UINT
Address   SIW21
Comment   Status Word 1 in DC9100 (Digital input)

Data Type   DIGITAL
Address   SIW21.1
Comment   Logic Input 1 (DC9100)

Data Type   DIGITAL
Address   SIB6.1
Comment   Digital Input 1 in (DX9100)

Data Type   REAL

Address   SIN4C7
Comment   Analog Input Value 1 (DX9100)

Data Type   LONG
Address   SILD
Comment   DI1 Pulse Counter (DX9100)

1.6.2.2 Real Time Clock

Data Type   BCD
Address   RTC6
Comment   Real Time Clock Minutes (DX9100)

1.6.2.3 Daylight Savings

Data Type   BCD
Address   DS0
Comment   Daylight Savings Start date (DX9100)

1.6.2.4 Exception Days

Data Type   BCD
Address   ED2
Comment   Exception Days Start Date #02 (DX9100)

1.6.2.5 Time Schedule

Data Type   BCD
Address   TS1.1B
Comment   Time Schedule module 1, End Time Event #08 (DX9100)

Data Type   UINT
Address   TS1.1C
Comment   Time Schedule module 1, Enable Days Event #08 (DX9100)

1.6.3 Data types N2Open

IO Device Type	Citect data format	Citect data types	Description/Special Usage/Limitations/ Valid Ranges
Internal Parameter ADF	IPADF*x*	REAL	Read / Write
Internal Parameter ADI	IPADI*x.y*	INT	Read / Write.
Internal Parameter BD	IPBD*x.z*	BYTE	Read / Write.
Override Internal ADF	OIPADF*x*	REAL	Read / Write.
Override Internal ADI	OIPADI*x.y*	INT	Read / Write.
Override Internal BD	OIPBD*x.z*	BYTE	Read / Write

Where:

x	Item number, 1 - 255
y	Bit number, 0 - 15
z	Bit number, 1 - 8

1.6.4 Examples data types N2Open

1.6.4.1 Internal Parameter

Data Type   REAL
Address   IPADF1
Comment

Data Type   INT
Address   IPADI1
Comment

Data Type   DIGITAL
Address   IPADI1.1
Comment

Data Type   BYTE
Address   IPBD1
Comment

Data Type   DIGITAL
Address   IPBD1.1
Comment

1.6.4.2 Override Internal Parameters

Data Type   REAL
Address   OIPADF1
Comment

Data Type   INT
Address   OIPADI1
Comment

Data Type   DIGITAL
Address   OIPADI1.1
Comment

Data Type   BYTE
Address   OIPBD1
Comment

Data Type   DIGITAL
Address   OIPBD1.1
Comment

1.6.5 Hints, Tips, and Frequently asked questions

- Take care to use bit writing if you have a PLC program running in the DUC. The driver is first reading the word and masking the bit and afterward sending it back to the unit. If the PLC program make a change in the same word during the time the driver is manipulating this word the PLC changing can be overwritten.
- The Johnson Controls floating point values uses only sixteen bit. Use therefore maximum only one decimal in the presentation.
- If you want to set output 3 to 8 in DX-9100 direct from Citect you have to first enable the output. You have also to put the bit for Supervisory System Active SUP W(16) to active state and refreshes the DX9100 with 120 minutes time-out. The best way to do this is from Cicode. This is a watchdog function, which belongs to the security system. The project developer has the responsibility for this procedure, not the driver itself.
- You can change the Counters size in DX-9100 between 16 bits or 32 bits direct from Citect with B(4) in DXS1.
- With tcpip it can sometimes be good to increase the timeout parameter to a very high value e.g. 12000. The reason for this is that if you should have problem with disturbances the tcpip part can make retries before Citect make a timeout. This is typical for protocols without a synchronize mechanism

1.7 Driver reference

	Detail
Driver name	AQUACOM
Maximum array size[1]	256

1.7.1 Driver generated error codes

Driver Error Code (Hexadecimal)	Mapped to (Generic Error label)	Meaning of Error Code
101	**GENERIC_INVALID_COMMAND**	Not existing command mode
102	**GENERIC_INVALID_DATA**	Data not matching the item or function type
181	**GENERIC_BAD_PARAMETER**	Not existing item or function
182	**GENERIC_CMD_CANCELED**	Temporarily impossible to access the item
183	**GENERIC_ACCESS_VOILATION**	Not programmable item
184	**GENERIC_INVALID_DATA**	Table programmed with illegal items
185	**GENERIC_INVALID_DATA**	Trend programmed with illegal item
186	**GENERIC_INVALID_COMMAND**	Invalid Functional Module
187	**GENERIC_ADDRESS_RANGE_ ERROR**	Exceeding Addressing Range
188	**GENERIC_NO_RESPONSE**	Undefined Address after gate
189	**GENERIC_NO_RESPONSE**	No answer from Device after gate
18A	**GENERIC_ACCESS_VOILATION**	Password Protection Active
190	**GENERIC_HARDWARE_ERROR**	I=C-bus Error
191	**GENERIC_HARDWARE_ERROR**	Hardware not available
192	**GENERIC_BAD_PARAMETER**	Illegal item number
193	**GENERIC_INVALID_DATA**	Counters unreliable
194	**GENERIC_HARDWARE_ERROR**	Power supply unreliable

[1] Equivalent to 'Maximum Request Length'

1.7.2 Standard Parameters

Parameter	Default	Allowable values	Description
Block (bytes)	4		(Not used by the driver)
Delay (mS)	5		(30000/BR mS (BR=Baud Rate bit/sec))
MaxPending	2	2	
Polltime (mS)	0	0	
TimeOut (mS)*	1000	1000 to 32000	
Retry	1	0 to 3	
WatchTime (Sec)	30		

1.7.2.1 Block
The optimum size of data for a read request in bytes.

1.7.2.2 Delay
The period, in milliseconds, to wait between receiving a response and sending the next command.

1.7.2.3 MaxPending
The maximum number of pending commands that the driver holds ready for immediate execution on each channel.

1.7.2.4 Polltime
The polling time (in milliseconds). If the PollTime is set to 0 then the driver will work in interrupt mode.

1.7.2.5 Timeout
Specifies how many milliseconds to wait for a response before regarding a request as having failed

1.7.2.6 Retry
The number of times to retry a request which has timed out after no response.

1.7.2.7 WatchTime
The frequency that the driver uses to check the communications link to the I/O Device.

1.7.3 Driver Specific Parameters

| Parameter	Default	Allowable values	Description
RegisterTcpip	0	0=serial 1=tcpip -1=messagebox	Obsolete from version 3.02.00.001

1.7.3.1 RegisterTcpip (Obsolete from version 3.02.00.001)

The Citect parameter RegisterTcpip (Obsolete from version 3.02.00.001) must be set to 1 or -1 if it shall be possible to use tcpip. Default is 0 for serial. If you are setting this parameter to -1 you will see the following messagebox

If you choose **Yes** you have to continue to fill in your tcpip code you have got from Beijer Electronics AB. If you choose **No** you have to put in the serial code or if you choose Cancel you can choose Demo mode. If you don't want to have this extra messagebox you can set the parameter [JCONTROL]

RegisterTcpip to 0 or to1 in Citect.ini there 0 forces your choice to serial and 1 to tcpip.

In runtime you can read out the type of registration in the statistics parameter 18 = 1 "TcpIp registered" or 19=1 "Serial registered" in the Kernel driver window.
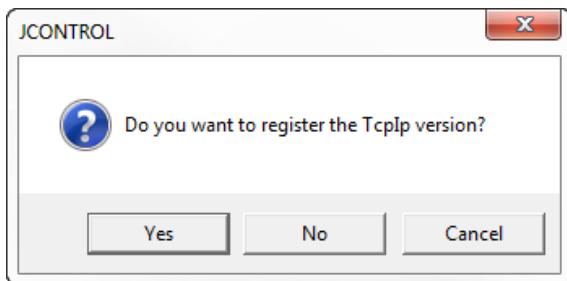
1.7.4 Driver generated statistics

Number	Label	Description
0	Frames Transmitted	Number of frames transmitted
1	Frames Received	Number of frames received
2	Received Interrupts	Number of interrupts
3	Write Requests	Number of write requests
4	Read Requests	Number of read requests
5	Frame Accepted	Number of accepted received frames
6	Frame With Error	Number of received frames with errors
7	Read Modify Write	Number of errors in received frames
8		
9		
10		
11		
12		
13		
14		
15		
16		
17		
18	TcpIp registered	Value = 1 when tcpip version is registered (Obsolete from version 3.02.00.001)
19	Serial registered	Value = 1 when serial version is registered (Obsolete from version 3.02.00.001)

1.7.5 Debug messages

1.7.5.1 Initialisation:

Item 0 is used for initialisation. In the answer it is possible to read what type of unit and version is used. In the example below the answer is 15 and 12 for the two multidropped units. (15H = DX9100 version 2.x and 12H = DC9100 version 2)
Sat Feb 19 21:17:12 2000 31:27:03.825 Transmit Length 14
>0180007A0405<d>
Sat Feb 19 21:17:12 2000 31:27:03.867 Receive Length 10
A15D51545<d>
Sat Feb 19 21:17:12 2000 31:27:03.955 Transmit Length 14
>0080007E0105<d>
Sat Feb 19 21:17:12 2000 31:27:03.987 Receive Length 10
A12DB124C<d>

1.7.5.2 Reading:

In the example item 4C7 (C704) is readout and the item address is the value for analogue input 1.
Sat Feb 19 21:20:19 2000 31:30:10.903 Transmit Length 16
>0184C704892583<d>
Sat Feb 19 21:20:19 2000 31:30:10.942 Receive Length 12
A58554C06B4<d>

1.7.5.3 Writing:

Sat Feb 19 21:25:56 2000 31:35:47.978 Transmit Length 20
>01D801050013A4684A<d>
Sat Feb 19 21:25:56 2000 31:35:48.015 Receive Length 2
A<d>

1.7.5.4 Error:

Here we are trying to read a non-existing item 7F00. The answer is NAK and the errorcode is 81.
Sat Feb 19 21:32:27 2000 31:42:18.681 Transmit Length 16
>0184007FE9118A<d>
Sat Feb 19 21:32:27 2000 31:42:18.714 Receive Length 4
N81<d>
Sat Feb 19 21:32:27 2000 31:42:18.714 Error: Bad user parameters
  READ   001d PORT2_BOARD1   IODev   SIW7F00(32512)   1
  Generic 000029 Driver 00000385 (0x00000181)

2. Analysis

2.1 JControl overview

This section documents the requirements of the protocol itself.

2.1.1 IO Device Online Test

The item 0 is read out from the device at initialization. The driver makes the standard control about checksums, errors and so on. If the message is accepted the driver will tell Citect the device is online. In the answer from the unit is the device code laying. It's easy to see the type and revision for the unit. The digits after A in the protocol is the device code A15D51545<d> e.g. 15 stand for DX9100 Revision 2.x

Device Code	Description	Revision
01H	DR9100 - Room Controller	1
11H	DR9100 - Room Controller	2
02H	DC9100 - Plant Controller	1
12H	DC9100 - Plant Controller	2
03H	IM9100 - Modem Interface	1
04H	DO9100 - Digital Optimizer	1
05H	DX9100 - Digital Controller	1.x
15H	DX9100 - Digital Controller	2.x
25H	DX9100 - Digital Controller	3
06H	TC9100 - Room Controller	1
08H	XT9100 - Extension Module	1
18H	XTM905 - Extension Module	1

2.1.2 Message Structure

2.1.2.1 Single item read message format:

">"	ADR	CMD	ITEM	BCC	CHECKSUM	"Cr"

Positive answer:

A	DATA	BCC	CHECKSUM	"Cr"

Negative answer:

N	ERROR	"Cr"

2.1.2.2 Single item write message format:

">"	ADR	CMD	ITEM	DATA	BCC	CHECKSUM	"Cr"

Positive answer:

A	"Cr"

Negative answer:

N	ERROR	"Cr"

2.1.2.3 Functional module read message format:

">"	ADR	CMD	MOD	INDEX	BCC	CHECKSUM	"Cr"

Positive answer:

A	DATA	BCC	CHECKSUM	"Cr"

Negative answer:

N	ERROR	"Cr"

2.1.2.4 Functional module write message format:

">"	ADR	CMD	MOD	INDEX	DATA	BCC	CHECKSUM	"Cr"

Positive answer:

A	"Cr"

Negative answer:

N	ERROR	"Cr"

Where:

>	Start Command Character
ADR	Two digits hexadecimal transmitted as two ASCII characters, indicates which of the 256 System 91 units on the serial link is being addressed, ranging from 00 to 0FFH.
CMD	Two digits hexadecimal Command Code.
ITEM	Two digits hexadecimal or one word hexadecimal indicating which of the possible items, defined in the All Items List, is interested in the transaction. Address up to 0FFFFH.
DATA	N bytes data according to item or function type
BCC	Block check sum word obtained executing of all transmitted ASCII from ADR to DATA included.
CHECKSUM	Checksum byte obtained executing the sum, modulo 256, of all transmitted ASCII from ADR to BCC included.
Cr	Termination character, ASCII '0DH'
ERROR	Error code see section 1.7.1
MOD	One byte hexadecimal indicating which Functional Module is interested in the transaction.
INDEX	One to three bytes hexadecimal for selection of parameters within the functional module; depending on the selected functional module some of the digits are not used or assume special meanings.

2.1.3 Pages

Different units uses different pages. DC9100 uses page 0 to 3. DX9100 uses page 0 to 3 and the extended area. DX9100 can use the whole range from address 0000 to FFFF together with the command 84/C4. The driver looks at the item address and changes the read/write command by itself. It's therefore possible to mix e.g. DC and DX units on the same multidropline.

	Item address	Read Command	Write Command
Page 0	00 to FF	80	C0
Page 1	100 to 1FF	81	C1
Page 2	200 to 2FF	82	C2
Page 3	300 to 3FF	83	C3
Extended	400 to FFFF	84	C4

2.2 DBFs
2.2.1 Help.dbf

TYPE	DATA	FILTER
PROTOCOL	JCONTROL	
PROTOCOL	JCONTROL2	

2.2.2 Protdir.dbf

TAG	FILE	BIT_BLOCK	MAX_LENGTH	OPTIONS
JCONTROL	JCONTROL	256	256	0x0DBf
JCONTROL2	N2OPEN			0x0DBf

2.2.3 JControl.dbf Entries

TEMPLATE	UNIT_TYPE	RAW_TYPE	BIT_WIDTH	LOW	HIGH	COMMENT
SIN%<16X	0x00000001	2	32	0	65535	Single Item Real, 0 - FFFF
SIB%<16X[.%u(1,1,8)]	0x00000002	8	8	0	65535	Single Item Byte, 0 - FFFF
SIW%<16X[.%u(1,1,16)]	0x00000003	1	16	0	65535	Single Item Integer, 0 - FFFF
SIL%<16X	0x00000004	4	32	0	65535	Single Item Long, 0 - FFFF
RTC%<16X	0x00000005	3	16	0	7	Real Time Clock, 0 - 7
DS%<16X	0x00000006	3	16	0	1	Daylight Savings, 0 - 1
ED%<16X	0x00000007	3	16	0	59	Exception Days, 0 - 3B
TS%<16.%+X%*256	0x00000008	3	16	0	28	Time Schedule, 0 - 7. 0 - 1C
TS%<16.%+X%*256	0x00000008	1	16	0	28	Time Schedule, 0 - 7. 0 - 1C

2.2.4 N2Open.dbf Entries

TEMPLATE	UNIT_TYPE	RAW_TYPE	BIT_WIDTH	LOW	HIGH	COMMENT
IPADF%<16U	0x00000010	2	32	1	255	Internal Parameter ADF, Real
IPADI%<16U[.%u(0,0,15)]	0x00000011	1	16	1	255	Internal Parameter ADI, Int
IPBD%<16U[.%u(1,1,8)]	0x00000012	8	8	1	255	Internal Parameter BD, Byte
OIPADF%<16U	0x00000013	2	32	1	255	Over Int Parameter ADF, Real
OIPADI%<16U[.%u(0,0,15)]	0x00000014	1	16	1	255	Over Int Parameter ADI, Int
OIPBD%<16U[.%u(1,1,8)]	0x00000015	8	8	1	255	Over Int Parameter BD, Byte

2.2.5 PROTERR.dbf Entries

PROTOCOL	MASK	ERROR	MESSAGE	REFERENCE	ACTION	COMMENT
JCONTROL	0	101	Not existing command mode			
JCONTROL	0	180	Data not matching the item or function type			
JCONTROL	0	181	Not existing item or function			
JCONTROL	0	182	Temporarily impossible to access the item			
JCONTROL	0	183	Not programmable item			
JCONTROL	0	184	Table programmed with illegal items			
JCONTROL	0	185	Trend programmed with illegal item			
JCONTROL	0	186	Invalid Functional Module			
JCONTROL	0	187	Exceeding Addressing Range			
JCONTROL	0	188	Undefined Address after gate			
JCONTROL	0	189	No answer from Device after gate			
JCONTROL	0	18A	Password Protection Active			
JCONTROL	0	190	I=C-bus Error			
JCONTROL	0	191	Hardware not available			
JCONTROL	0	192	Illegal item number			
JCONTROL	0	193	Counters unreliable			
JCONTROL	0	194	Power supply unreliable			
JCONTROL	FF	100	*JCONTROL error	DC page 2.4, DX page 6-5, TC page 3-3, XT page 10/35		

2.3 Development resources

2.3.1 Contacts

begcomm Communication AB
Brunnehagen 109
S-417 47 Göteborg
Phone: +46(0)70-5305769
Email: info@begcomm.com
Website: www.begcomm.com

2.3.2 Documents

- Johnson Controls DX9100 Specifications Rev 3.14	1-Sep-1997
- Johnson Controls DC9100 Specifications Ver 1.03	16-Jun- 1988
- Johnson Controls TC9100 Specifications Rev 3.0	22-Dec-1995
- Johnson Controls XT9100 Design specification Ver 3.6	17-Feb-1992
- METASYS N2 SYSTEM PROTOCOL SPECIFICATION FOR VENDORS	05/10/99

2.3.3 Driver Version History

Version	Modified By	Details
2.00.00.001	Bertil Göransson	Original
2.00.01.001	Bertil Göransson	Cleaning of receive buffer before transmit
2.00.01.002	Bertil Göransson	New ProtocolID
2.01.00.001	Bertil Göransson	Tcplp implemented with special registration facility
2.02.00.001 B1	Bertil Göransson	N2Open implemented. Only Internal Parameters Read/Write
2.02.00.002 B1	Bertil Göransson	Override Internal Parameters Read/Write for N2Open implemented
2.02.01.001	Bertil Göransson	Updated for Citect Citect V6
2.02.02.001	Bertil Göransson	TimeChannels (TS) updated for Citect V7
3.00.00.001	Bertil Göransson	Internal information changed to begcomm. New GUID
3.01.00.001	Bertil Göransson	Ported to VS2010
3.02.00.001	Bertil Göransson	Changed license system to only one license type for both TCP and serial communication. N2Open is implemented in the documentation. N2Open is not undocumented anymore.
3.02.00.002	Bertil Göransson	Serial ID implemented in the Protection project. Ported to VS2017.