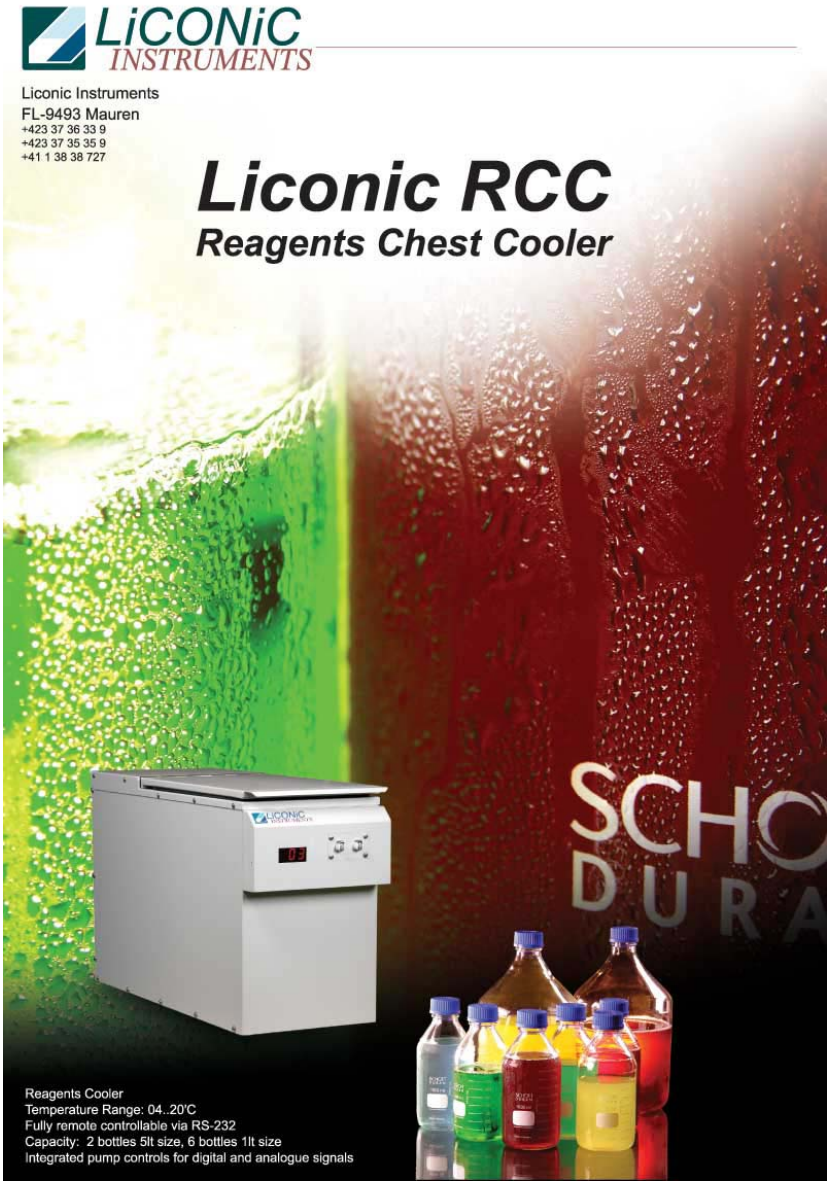
RCC
Reagents Chest Cooler



Technical Description
May, 09th 2004

RCC Reagents Chest Cooler

CONTENT

1. General Description 3
1.1. Scope 3
1.2. Principal of Operation 3
2. Output Specifications 3
2.1. Digital Outputs 3
2.2. Analog Outputs 3
2.3. Pump Interface Connector 3
2.4. Pump Interface Connector Pin Layout 3
3. Serial Communication Port Description 4
3.1. RS-232 Serial Port Configuration 4
3.2. Command Transmission Procedure 4
3.2.1. Brake Signal 4
3.2.2. Command Syntax 5
3.2.3. Open / Close Communication 6
3.2.4. Controller Error Messages 7
4. Command Descriptions 8
4.1. Digital Pump Commands 8
4.2. Analog Pump Commands 8
4.3. Temperature Related Commands 8
4.4. Output Command Table 9
5. Electrical Specifications 9
5.1. Electrical Specifications Table 9
6. Mechanical Specifications 9
6.1. Mechanical Specifications Table 9
6.2. Chiller Specifications Table 9
6.3. Drawings 10

RCC Reagents Chest Cooler

1. General Description

1.1. Scope

The RCC is a air convection cooler.

1.2. Principal of Operation

A chest with a cover that can be opened manually contains the reagents bottles. A evaporator is located inside the chest. A ventilator driven by a brushless DC motor is located next to the evaporator. A defrost valve allows injection of hot gas form the compressor into the evaporator.

2. *O*utput Specifications

2.1. Digital Outputs

Output type	NPN ground switching	
Rated output load	0.25	ADC
Max. output voltage	30	V
Off leakage current	<.5	uA
On resistance	<1	R
Switching time	<2	ms

2.2. Analog Outputs

Analog output range	4 .. 20	mA
Output impedance	0.5	R
Max. load resistance	400	R
Resolution	4	uA
Accuracy	<1	% FS
Conversion speed	<10	ms
Insulation	Photocoupler (cannels are *not* individually insulated)	--

2.3. Pump Interface Connector

Connector type	D-SUB
Number of pins	9
Polarity	female

2.4. Pump Interface Connector Pin Layout

Pin number	Description
1	Ground
2	N.c.
3	N.c.
4	Pump activate
5	Pump flow direction
6	Analog signal
7	Analog ground
8	N.c.
9	N.c.

3. Serial Communication Port Description

3.1. RS-232 Serial Port Configuration

ASCII data format
Full duplex
PC: Delimiter CR (Chr 13h)
PLC: Delimiter CR,LF (Chr 13h,10h)
9600 Baud
8 Data bits
1 Stop bit
Parity even

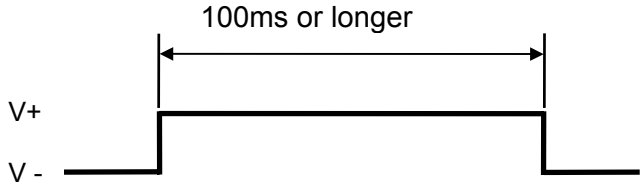
The example program shows how the comport is initialized under MS-DOS. For details refer to the MS-DOS manuals.

```
FUNCTION STX_InitCom(pN:INTEGER):INTEGER;
  VAR n:INTEGER;
  BEGIN
    n:=ModeCom(pN,9600,'E',8,1);
    STX_InitCom:=n
  END;
```

3.2. Command Transmission Procedure

3.2.1. Brake Signal

The controller serial port is reset by sending a 'Brake Signal'. The specification of the Brake Signal is given below.



In most cases the Brake Signal can be omitted. The example program shows how a Brake Signal can be implemented under MS-DOS. The example program consists of two procedures where the inner procedure "LCR_Brake" is called by the outer procedure ("STX_ComReset"). Note that this program will directly address the comport chip. There may be operating systems that will not allow the chip to be addressed directly.

```
PROCEDURE STX_ComReset(pN:INTEGER);
  PROCEDURE LCR_Brake(rAdr,tme:INTEGER);
    BEGIN
      Port[rAdr]:=Port[rAdr] OR $40; Delay(tme); {Bit6=Brake}
      Port[rAdr]:=Port[rAdr] AND $BF; Delay(2);
    END;
  BEGIN
    CASE pN OF
      1: LCR_Brake($3FB,500);
      2: LCR_Brake($2FB,500);
    END
  END;
```

3.2.2. Command Syntax

For communication only a few commands are required. A command is an ASCII-string which is sent to the controller. Response is an ASCII string sent by the controller. Note that each command is prompted by a Response string.

A command consists of command segments. The first command segment defines the intention of the command. Command segments are separated by Space (ASCII 20h). Response Segments are separated by comma (ASCII 1Ch). The table below gives a list of abbreviations used later on.

Command Segment	Mnemonics
Communication Request	CR
Communication Quit	CQ
Communication Clear	CC
Communication Finished	CF
Set	ST
Reset	RS
Read	RD
Write	WR
Write Set	WS
Data Memory	DM
Timer	T
Space ASCII 20h	*sp*
Line Feed ASCII 0Ah	*lf*
Carriage Return ASCII 0Dh	*cr*

The following example program shows how a string sent to the StoreX controller can be generated and sent. The “STX_SendStr” will allow a string “s” to be sent through port “pN”. The result of the function may be used for error handling. Note that the string “s” does not require any delimiter. The delimiter is added inside the function. The “auxStrAut” may be any low level or DOS procedure that supports the transmission through the comport.

```
FUNCTION STX_SendStr(pN:INTEGER; s:STRING):INTEGER;
  VAR ss:STRING;
  BEGIN
    STX_SendStr:=0;
    ss:=Concat(s,cr);
    auxStrOut(pN,ss)
  END;
```

Since every command is prompted by the StoreX it makes sense to introduce a procedure that handles this send-receive sequence. A possible solution is given below.

```
FUNCTION STX_ReadBackStr(pN:INTEGER; s:STRING):STRING;
  CONST tries=2;
  VAR i,n,m,err:INTEGER; w:WORD; s0,s1:STRING; c,kp:CHAR;
  BEGIN
    IF NOT(kbdEsc) THEN
      BEGIN
        EmptyAux(pN);
        i:=-1;
        s0:=s;
```

```
        m:=Pos('-',s0);
        IF m>0 THEN
          BEGIN
            Delete(s0,m,1);
            s1:=Copy(s0,m,Length(s0));
            Val(s1,n,err);
            Delete(s0,m,Length(s0));
            w:=-n; Str(w,s1); s0:=s0+s1
          END;
        REPEAT
          Inc(i);
          EmptyAux(pN); auxStrOut(pN,s0+cr);
          IF i>3 THEN DelayMs(100);
          auxStrIn(pN,s1,5,lf);
          Delete(s1,PRED(Length(s1)),2)
        UNTIL (s1[1]<>'E') OR (i>tries) OR KbdEsc;
        STX_ReadBackStr:=s1;
      END
    ELSE
      STX_ReadBackStr:=''
  END;
```

3.2.3. Open / Close Communication

Prior to communication with the controller, the communication has to be opened. Before communication is opened, the controller only accepts the Open Communication Command (CR). For improved safety, it is recommended that communication is closed (CQ) when no communication is required for a long period of time.

	Command	Response
Open Communication *Send Commands (see below)* Close Communication	CR *cr* CQ *cr*	CC *cr lf* CF *cr lf*

The two example programs show how to open and close communication to the StoreX.

```
FUNCTION STX_OpenCom(pN:INTEGER):STRING;
  BEGIN
    STX_OpenCom:=STX_ReadBackStr(pN,'CR')
  END;

Time out functions and communication error can be trapped at this
level.

FUNCTION STX_CloseCom(pN:INTEGER):STRING;
  BEGIN
    STX_CloseCom:=STX_ReadBackStr(pN,'CQ')
  END;
```

These examples show how simple communication becomes when using the “STX_ReadBack” procedure. The following example explains how often used sequences are programmed. The ‘Set-‘procedure sets an internal relay (or flag). The value of the flag becomes ‘1’. The ‘Reset-‘procedure resets an internal relay (or flag) . The value of the flag becomes ‘0’. Flags can be set, reset or read.

```
FUNCTION STX_Set(pN,rel:INTEGER):STRING;
  VAR sR:STRING;
  BEGIN
    Str(rel,sR);
    STX_Set:=STX_ReadBackStr(pN,'ST '+sR)
  END;

FUNCTION KV_Reset(pN,rel:INTEGER):STRING;
  VAR sR:STRING;
  BEGIN
    Str(rel,sR);
    STX_Reset:=STX_ReadBackStr(pN,'RS '+sR)
  END;

FUNCTION STX_Read(pN,rel:INTEGER):STRING;
  VAR sR:STRING;
  BEGIN
    Str(rel,sR);
    STX_Read:=STX_ReadBackStr(pN,'RD '+ sR)
  END;
```

The Set and Reset will return an 'OK'-response if operation is successful. The Read-procedure will return a '0' or '1'.

Data memories are 16 bit oriented. They can be read or written. The following examples show how to use the data memories.

```
FUNCTION STX_ReadDataMemory(pN,nbr:INTEGER):STRING;
  VAR sR:STRING;
  BEGIN
    Str(nbr,sR);
    STX_ReadDataMemory:=STX_ReadBackStr(pN,'RD DM'+ sR)
  END;

FUNCTION STX_WriteDataMemory(pN,nbr:INTEGER; valu:WORD):STRING;
  VAR sR,sV:STRING;
  BEGIN
    Str(nbr,sR); Str(valu,sV);
    STX_ReadDataMemory:=STX_ReadBackStr(pN,'WR DM'+ sR + ' ' + sV)
  END;
```

Note that the Response on the above Write procedure is always 'OK'. The Read procedure will return a five-character-string.

3.2.4. Controller Error Messages

The following Error Codes are sent by the PLC. These error codes indicate system-errors and are not the same as the Instruments own error-messages (refer to "Handling Error Messages")

Error	Comment	Response
Relay Error	Undefined timer, counter, data memory, check if requested unit is valid	E0 *cr lf*
Command Error	Invalid Command, check if communication is opened by CR, check command sent to controller, check for interruptions during string transmission	E1 *cr lf*
Program Error	Firmware lost, reprogram controller	E2 *cr lf*
Hardware Error	Controller hardware error, turn controller ON/OFF, controller is faulty and has to be replaced	E3 *cr lf*

Write Protected Error	Unauthorized Access	E4 *cr lf*
Base Unit Error	Unauthorized Access	E5 *cr lf*

4. Command Descriptions

4.1. Digital Pump Commands

Command	Description	Response	Comment
ST 500	Activate pump 1 motion	OK	
RS 500	Stop pump 1 motion	OK	
RD 500	Read Status of pump 1 motion Stop	0	
RD 500	Read Status of pump 1 motion Running	1	
ST 503	Activate pump 2 motion	OK	
RS 503	Stop pump 2 motion	OK	
RD 503	Read Status of pump 2 motion Stop	0	
RD 503	Read Status of pump 2 motion Running	1	
ST 502	Reverse pump 1 motion	OK	
RS 502	Pump 1 forward motion	OK	
RD 502	Read direction of pump 1 motion forward	0	
RD 502	Read direction of pump 1 motion reverse	1	
ST 505	Reverse pump 2 motion	OK	
RS 505	Pump 2 forward motion	OK	
RD 505	Read direction of pump 2 motion forward	0	
RD 505	Read direction of pump 2 motion reverse	1	

4.2. Analog Pump Commands

Command	Description	Response	Comment
ST 2800	Activate Analog Output pump 1	OK	Analog output random analog value (+/- 10V) prior command sent
WR DM996 x	Write analog value pump 1 [0 .. 4000]	OK	
RD DM996	Read analog value pump 1	x	
ST 2801	Activate Analog Output pump 2	OK	Analog output random analog value (+/- 10V) prior command sent
WR DM997 y	Write analog value pump 2 [0 .. 4000]	OK	
RD DM997 y	Read analog value pump 2	y	

4.3. Temperature Related Commands

Command	Description	Response	Comment
WR DM890 z	Set operating temperature [0..200]	OK	Value in Deg. 1/10 C. E.g..100=10'C
RD DM890	Read operating temperature value	z	
RD DM982	Read actual temperature	t	

4.4. Output Command Table

Pin number	Description	Pump 1	Pump 2
1	Ground		
2	N.c.		
3	N.c.		
4	Pump activate	ST 500 / RS 500	ST 500 / RS 503
5	Pump flow dir.	ST 502 / RS 502	ST 502 / RS 505
6	Analog signal	ST 2800 / WR DM996 x	ST 2801 / WR DM997 x
7	Analog ground		
8	N.c.		
9	N.c.		

5. Electrical Specifications

5.1. Electrical Specifications Table

Rated Supply Voltage	230	V
Supply Voltage Frequency	50	Hz
Power Consumption typ.	120	W
Power Consumption max.	160	W
Supply Current typ.	0.7	A
Supply Current max.	0.9	A
Fuse	4	AT

6. Mechanical Specifications

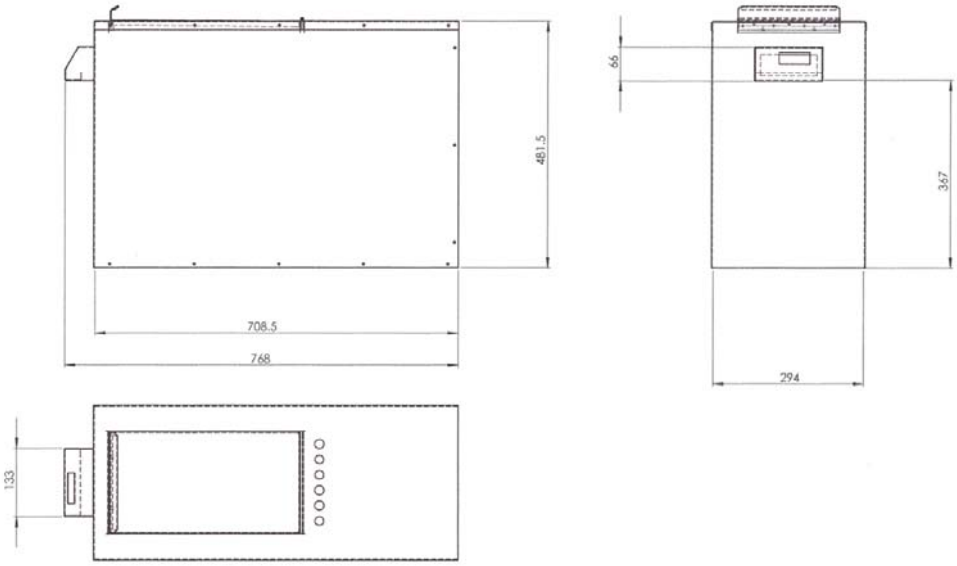
6.1. Mechanical Specifications Table

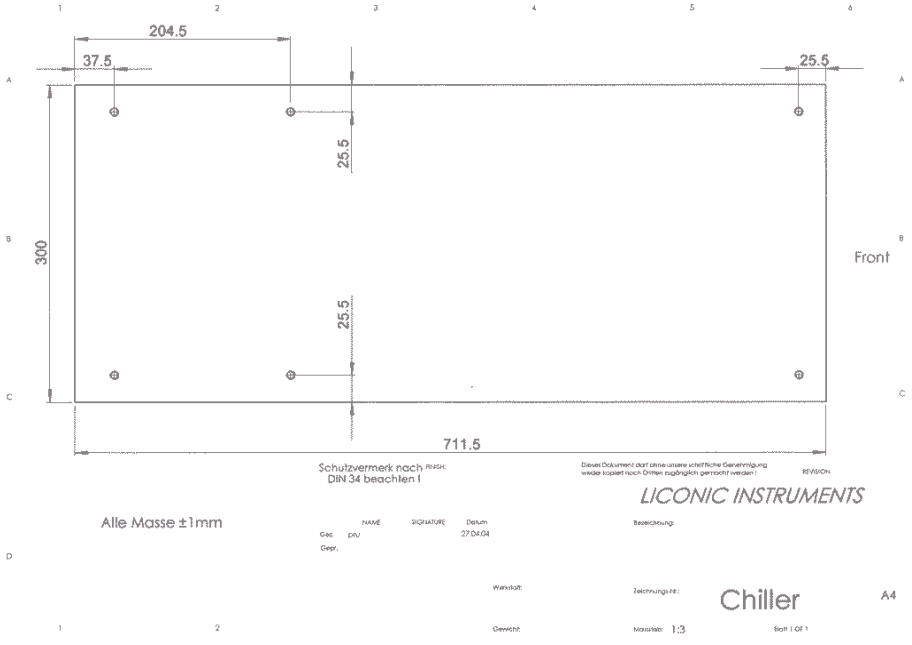
Capacity Volume	30	l
Capacity 5l Schott Duran 5000ml	2	Pcs
Capacity 1l Schott Duran 1000ml	6	Pcs
Dimensions (wxdxh)	294x768x480	mm
Weight		kg
Noise level		dBA

6.2. Chiller Specifications Table

Volume		cm3
Cooling Fluid	R134a	
Protection	Overcurrent / Overtemperature	--

6.3. Drawings


481.5
66
367
708.5
768
294
133


204.5
37.5
25.5
25.5
300
Front
25.5
711.5
Schutzvermerk nach DIN 34 beachten !
LICONIC INSTRUMENTS
Alle Masse ±1mm
Chiller
A4