LiCONiC INSTRUMENTS

Remote Operation

for

StoreX

Precision Incubator Series

STX40, STX100, STX200, STX500, STX1000
IC, HC, HR, DC, DR, DF
SA, BT, IT
STP1000

Important Notice

The Information contained in this document is subject to change without notice.

Information provided by LiCONiC AG is believed to be accurate and reliable. However, the user is responsible for the proper and correct use of the product. No license is granted by acquisition of the product for any patent or patent rights of LiCONiC. If the user does not follow the instructions given in this manual, LiCONiC does not take any responsibility for injuries or damages caused by the LiCONiC product.

Contact Address

LiCONiC AG
Industriestrasse 8
FL – 9493 Mauren
Principality of Liechtenstein

Telephone: +423 373 63 39
Telex: +423 373 53 59
E-mail: info@liconic.com
Internet: www.liconic.com

Before operating the instrument, the user must read and understand this manual.

Installation and Servicing

Installation, servicing and reinstallation of the instrument shall only be performed by System Integrators and / or service personnel authorized by LiCONiC AG.

LiCONiC regularly offers Operator Training Courses. We highly recommend the attendance of such a course prior to working with a LiCONiC Instrument.

Table of Contents

1. Remote Operation

1.1 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

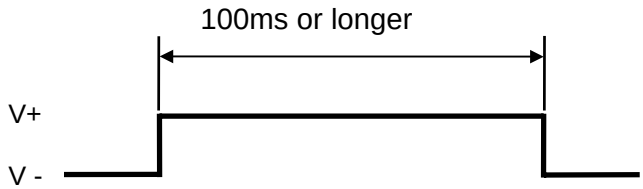
The example program shows how the comport is initialized under MS-DOS. For details refer to the MS-DOS manuals.

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

1.2 Command Transmission Procedure

1.2.1 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;
```

1.2.2 Command Definition

A command is any string that is recognized and interpreted by the controller logic. Commands consist of one or more ASCII sub-strings separated by a space character.

1.2.3 Operation Definition

Operations are physically executable commands. Operations are triggered by one or by a set of commands. A set of commands consists of two or more commands separated by an ASCII carriage return character.

1.2.4 Response Definition

A Response is a string sent by the controller logic following a previously sent command.

- Command: PC --> Controller
- Resonse: PC <-- Controller

1.2.5 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;
```

1.2.6 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	CR *cr*	CC *cr lf*
Send Commands (see below)		
Close Communication	CQ *cr*	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.

1.2.7 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*

1.2.8 List of Operations

Operations are physically executable commands. Operations are triggered by one or by a set of commands. A set of commands consists of two or more commands separated by an ASCII carriage return character. Prior completing a set of commands for an operation the Ready bit (1915) must be ‘1’ except for Reset command (ST 1900) and the Soft Reset Command (ST 1800).

Operation	Command Sequence
Reset	ST 1900
Soft Reset	ST 1800
Continue Access *(used in conjunction with special configuration flags)*	ST 1902
Abort access *(used in conjunction with special configuration flags)*	ST 1903
Position Carrousel	WR DM0 x
Lift Positioning (when 1910 set only)	WR DM5 y
Initialize	ST 1801
Import Plate	WR DM0 x WR DM5 y ST 1904
Export Plate	WR DM0 x WR DM5 y ST 1905
Pick Plate	WR DM0 x WR DM5 y ST 1908
Place Plate	WR DM0 x WR DM5 y ST 1909

Get Plate	WR DM0 x WR DM5 y ST 1907
Put Plate	WR DM0 x WR DM5 y ST 1906
BarCode Search	ST 1710
Short Import	WR DM10 p
Short Export	WR DM15 n

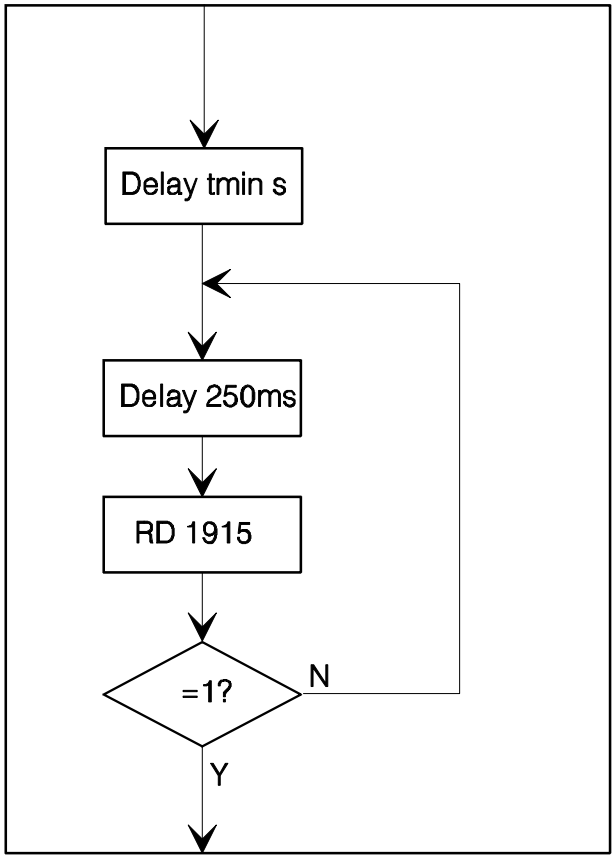
1.2.9 System Status

The Handling response on Ready-Polling ("x") is '0' (handling not ready to accept commands) or '1' (handling ready, command may be sent). In its idle status the handling sets the Ready Bit to '1'.

	Command	Response
Read Ready-Bit	RD *sp* 1915 *cr*	x *cr lf*

When polling the Ready Bit after sending a command, wait at least 200ms before requesting the Ready Status. We recommend to wait 100ms – 200ms between polling sequences.

Ready Polling



A sample program will explain the polling sequence in detail. Note that monitoring a flag change either from '0' to '1' or from '1' to '0' may also be advisable.

```
PROCEDURE STX_WaitReady(portNbr:INTEGER);
  BEGIN
    DelayMs(300);
    WHILE (STX_Read(portNbr,1915)='0') AND NOT(KeyPressed) DO
      DelayMs(200)
  END;
```

Note !
- An Operation may only be executed when Ready Bit equals '1' (RD 1915 -> '1').
- Allow >200ms delay between last command sent for execution of an operation and the first Ready Bit request.
- Use 100ms..200ms delay between requests.

1.3 Commands

1.3.1 Basic Commands

	Command	Response
Reset	**ST *sp* 1900 *cr***	**OK *cr lf***
Read Ready Flag	**RD *sp* 1915 *cr***	**x[1] *cr lf***
Read Error Flag	RD *sp* 1814 *cr*	x *cr lf*
Read Plate-Ready Flag	RD *sp* 1815 *cr*	x *cr lf*
Initialize Handling System	**ST *sp* 1801 *cr***	**OK *cr lf***
Set stacker slot position m (0..m)	**WR *sp* DM0 *sp* m *cr***	**OK *cr lf***
Set Handler level position n (1..n)	**WR *sp* DM5 *sp* n *cr***	**OK *cr lf***
Set Handler Stacker Pitch	WR *sp* DM23 *sp* p *cr*	OK *cr lf*
Open Gate[2]	ST *sp* 1901 *cr*	OK *cr lf*
Close Gate[3]	ST *sp* 1902 *cr*	OK *cr lf*
Terminate Access	ST *sp* 1903 *cr*	OK *cr lf*
Import plate to m,n*)	**ST *sp* 1904 *cr***	**OK *cr lf***
Export plate from m,n*)	**ST *sp* 1905 *cr***	**OK *cr lf***
Set plate to x-fer station m,n*)	ST *sp* 1906 *cr*	OK *cr lf*
Get plate from x-fer station m,n*)	ST *sp* 1907 *cr*	OK *cr lf*
Pick plate form stacker, level m,n*)	ST *sp* 1908 *cr*	OK *cr lf*
Place plate to stacker, level m,n*)	ST *sp* 1909 *cr*	OK *cr lf*
Position z-Lift for BCR reading m,n*)	ST *sp* 1910 *cr*	OK *cr lf*
Activate Shovel Transfer Sensor (70º C HT Version)	ST *sp* 1911 *cr*	OK *cr lf*
Read Swap Station position	RD *sp* 1912 *cr*	x *cr lf*
Rotate Swap Station 180 degree	ST *sp* 1912 *cr*	OK *cr lf*
Rotate Swap Station back to home position	RS *sp* 1912 *cr*	OK *cr lf*
Read Shaker status	RD *sp* 1913 *cr*	OK *cr lf*
Activate Shaker	ST *sp* 1913 *cr*	OK *cr lf*

[1] x = Response expect '0' or '1'
[2] Continue access on Gate Mode 1
[3] Continue access on Handshake Mode

Stop Shaker	RS *sp* 1913 *cr*	OK *cr lf*
Read 2nd Transfer Station Sensor	RD *sp* 1807 *cr*	x *cr lf*
Read Cassette Plate Presence Sensor	RD *sp* 1808 *cr*	x *cr lf*
Read User Door Switch	RD *sp* 1811 *cr*	x *cr lf*
Read Shovel Plate Sensor	RD *sp* 1812 *cr*	x *cr lf*
Read Transfer Station Plate Sensor	RD *sp* 1813 *cr*	x *cr lf*
Read Actual Temperature (1/10º Celsius)	**RD DM982 *cr***	**ttttt[1] *cr lf***
Read Set Temperature value (1/10º C; e.g. 370 = 37.0º Celsius)	RD DM890 *cr*	sssss[2] *cr lf*
Set operation temperature (1/10º C)	**WR DM890 *sp* ttttt *cr***	**OK *cr lf***
Read Actual Humidity (1/10% RH)	**RD DM983 *cr***	**ttttt *cr lf***
Read Set Temperature value (1/10% RH; e.g. 900 = 90.0% RH)	RD DM893 *cr*	sssss *cr lf*
Set operation humidity (1/10% RH)	**WR DM893 *sp* ttttt *cr***	**OK *cr lf***
Read Actual CO2 level (1/100% vol.)	**RD DM984 *cr***	**ttttt *cr lf***
Read Set CO2 level (1/100% vol. 500 = 5.0% CO2)	RD DM894 *cr*	sssss *cr lf*
Set operation CO2 level (1/100% vol.)	**WR DM894 *sp* ttttt *cr***	**OK *cr lf***
Read Actual N2 / O2 level (1/100% vol.)	**RD DM985 *cr***	**ttttt *cr lf***
Read Set N2 / O2 level (1/100% vol. 9000 = 90.0% N2)	RD DM895 *cr*	sssss *cr lf*
Set operation N2 / O2 level (1/100% vol.)	**WR DM895 *sp* ttttt *cr***	**OK *cr lf***
Read Actual N2 / O2 level (1/100% vol.) [3]	**RD DM986 *cr***	**ttttt *cr lf***
Read Set N2 / O2 level (1/100% vol. 9000 = 90.0% O2) [4]	RD DM896 *cr*	sssss *cr lf*
Set operation N2 / O2 level (1/100% vol.) [5]	**WR DM896 *sp* ttttt *cr***	**OK *cr lf***

The stacker slot is selected by sending m=1..2. The handler is positioned towards the specified slot. Note that stacker levels are numbered starting at the bottom level upwards. Bottom level is 1.

The Activate Handler-Command initializes the handling. Use the Activate Handler Command after cold-start or after sending a Reset Command. The Activate Handler Command should not be used as an 'on-off' function.
Functions marked with *) can only be used with selected stacker slots. The value of the number of stackers is stored in DM29. The value of the number of levels is stored in DM25. When using the Position z-Lift for BCR reading command the ST 1910 command has to be sent only once. Once set, the handler is positioned by simply writing the carrousel and level position into DM0 and DM5.

The Activate Plate Sensor command is used in HT-units only. In HT units the Shovel Plate Sensor is deactivated by default. The Activate Plate Sensor command has to be sent prior to reading the sensor status (ST 1911). This command will turn the sensor on for ~2 seconds. The sensor can be read during this period of time. After turning the sensor on wait approximately 0.1 of a second before reading the sensor with the RD 1812 command.
The Error-Flag is set when the handling cannot finish a movement in time. Check the Error-Flag by sending "RD 1814" when the Ready-Bit does not become '1'.

[1] t = actual value 5 characters (Word=16Bit)
[2] s = set value 5 characters (Word=16Bit)
[3] Used when O2 *and* N2 option is present, N2 gas is connected to N2-, O2 is connected to O2-inlet
[4] Used when O2 *and* N2 option is present, N2 gas is connected to N2-, O2 is connected to O2-inlet
[5] Used when O2 *and* N2 option is present, N2 gas is connected to N2-, O2 is connected to O2-inlet

The Plate Ready flag (1815) is set when the system has cleared the plate from the Transfer Station on executing an Import command or when a plate is placed on the Transfer Station during an Export command. This Plate Ready flag allows increased access speeds. The read Plate Ready (RD 1815) returns '1' until the read Ready flag (RD 1915) is set to '1'.

1.3.2 Examples of Usage of Basic Commands

The following example strings show the usage of the basic commands. To initialize the system after a cold-start or a reset you may send:

Command	Response	Comment
ST *sp* 1801 *cr*	OK *cr lf*	Initialize Command

Prior to sending an initialization command you may request the system status by reading the Ready Flag:

Command	Response	Comment
RD *sp* 1915 *cr*	'1' *cr lf*	System Ready
RD *sp* 1915 *cr*	'0' *cr lf*	System Busy

To import a plate from the Transfer Station to level 10 in the stacker at slot 2. The command to be sent is:

Command	Response	Comment
WR *sp* DM0 *sp* 2 *cr*	OK *cr lf*	Position rotation at slot 2
WR *sp* DM5 *sp* 10 *cr*	OK *cr lf*	Select level 10 in stacker
ST *sp* 1904 *cr*	OK *cr lf*	Import Command to start import process

To export a plate form level 22 in the stacker at slot 1 and place it on the Transfer Station, the command strings to be sent are:

Command	Response	Comment
WR *sp* DM0 *sp* 1 *cr*	OK *cr lf*	Position rotation at slot 1
WR *sp* DM5 *sp* 22 *cr*	OK *cr lf*	Select level 22 in stacker
ST *sp* 1905 *cr*	OK *cr lf*	Export Command to start export process

If you simply want to remove a plate from the shovel by placing it on the transfer station you may send:

Command	Response	Comment
WR *sp* DM0 *sp* 1 *cr*	OK *cr lf*	Position rotation at any position; must be defined
WR *sp* DM5 *sp* 1 *cr*	OK *cr lf*	Select any level
ST *sp* 1906 *cr*	OK *cr lf*	Start execution of setting plate from the shovel to the Transfer Station

Remember that Slot Position and Level Position must also be set at the "ST 1906" and "ST 1907" command.

In order to transport one plate from level 15 of stacker 2 to level 17 of the same stacker 2 the following sequence has to be sent:

Command	Response	Comment
WR *sp* DM0 *sp* 2 *cr*	OK *cr lf*	Position rotation at slot 1
WR *sp* DM5 *sp* 17 *cr*	OK *cr lf*	Pick plate from level 17 of stacker 2

ST *sp* 1908 *cr*	OK *cr lf*	Start execution of Pick plate from level 17 of stacker on the Shovel
RD *sp* 1915 *cr*	'0' *cr lf*	System busy executing command
RD *sp* 1915 *cr*	'0' *cr lf*	System Ready for next command
WR *sp* DM5 *sp* 15 *cr*	OK *cr lf*	Select level 15
ST *sp* 1909 *cr*	OK *cr lf*	Place command to place plate from shovel to the stacker

Please note that the slot position value in DM0 and / or the level information in DM5 remains in the Data Memories after completion of command. Therefore in above example the "WR DM0 1" needs not to be sent again.

1.3.3 Extended Commands

	Command	Response
Read Error Flag (default =0)	RD *sp* 1814 *cr*	x *cr lf*
Reset “Water Low” Alarm	RS 1505 *cr*	OK *cr lf*
Reset CO2 (Gassing 1) Timeout Error	RS 1504 *cr*	OK *cr lf*
Read Two Way Communication Activation Flag (default =0)	RD *sp* 1200 *cr*	x *cr lf*
Set Two Way Communication Activation Flag (default =0)	ST *sp* 1200 *cr*	OK *cr lf*
Reset Two Way Communication Activation Flag (default =0)	RS *sp* 1200 *cr*	OK *cr lf*
Read Flag: Send Handling Progress Status <SPX> (default =0)	RD *sp* 1201 *cr*	x *cr lf*
Set Flag: Send Handling Progress Status <SPX> (default =0)	ST *sp* 1201*cr*	OK *cr lf*
Reset Flag: Send Handling Progress Status <SPX> (default =0)	RS *sp* 1201 *cr*	OK *cr lf*
Read Flag: Send Climate Data <CxD> (default =0)	RD *sp* 1213 *cr*	x *cr lf*
Set Flag: Send Climate Data <CxD> (default =0)	ST *sp* 1213 *cr*	OK *cr lf*
Reset Flag: Send Climate Data <CxD> (default =0)	RS *sp* 1213 *cr*	OK *cr lf*
Read Flag: Send Plate Sensor Status Changes <SNX> (default =0)	RD *sp* 1214 *cr*	x *cr lf*
Set Flag: Send Plate Sensor Status Changes <SNX> (default =0)	ST *sp* 1214 *cr*	OK *cr lf*
Reset Flag: Send Plate Sensor Status Changes <SNX> (default =0)	RS *sp* 1214 *cr*	OK *cr lf*
Read Flag: Send System Status Byte Changes <SYX> (default =0)	RD *sp* 1215 *cr*	x *cr lf*
Set Flag: Send System Status Byte Changes <SYX> (default =0)	ST *sp* 1215 *cr*	OK *cr lf*
Reset Flag: Send System Status Byte Changes <SYX> (default =0)	RS *sp* 1215 *cr*	OK *cr lf*
Read Auto-End-Access Flag (default =1)	RD *sp* 1600 *cr*	x *cr lf*
Set Auto-End-Access Flag (default =1)	ST *sp* 1600 *cr*	OK *cr lf*
Reset Auto-End-Access Flag (default =1)	RS *sp* 1600 *cr*	OK *cr lf*

Read Gate Mode 1 Flag (default=0), wait behind Gate for ST 1901	RD *sp* 1601 *cr*	x *cr lf*
Set Gate Mode 1 Flag (default =0)	ST *sp* 1601*cr*	OK *cr lf*
Reset Gate Mode 1 Flag (default =0)	RS *sp* 1601 *cr*	OK *cr lf*
Read Gate Mode 2 Flag (default=0), close Gate after timeout	RD *sp* 1602 *cr*	x *cr lf*
Set Gate Mode 2 Flag (default =0)	ST *sp* 1602 *cr*	OK *cr lf*
Reset Gate Mode 2 Flag (default =0)	RS *sp* 1602 *cr*	OK *cr lf*
Read Gate Mode 3 Flag (default=0), never close Gate automatically	RD *sp* 1603 *cr*	x *cr lf*
Set Gate Mode 3 Flag (default =0)	ST *sp* 1603 *cr*	OK *cr lf*
Reset Gate Mode 3 Flag (default =0)	RS *sp* 1603 *cr*	OK *cr lf*
Read Input- and Output Transfer Station selection	RD *sp* 1607 *cr*	x *cr lf*
Swap Input and Output Transfer Station	ST *sp* 1607 *cr*	OK *cr lf*
Swap Transfer Stations to default position	RS *sp* 1607 *cr*	OK *cr lf*
Read Handshake Mode flag (default =0)	RD *sp* 1610 *cr*	x *cr lf*
Activate Handshake Mode (default =0)	ST *sp* 1610 *cr*	OK *cr lf*
De-activate Handshake Mode (default =0)	RS *sp* 1610 *cr*	OK *cr lf*
Read Plate Trace selection	RD *sp* 1611 *cr*	x *cr lf*
Plate Trace on, stop on plate handling error	ST *sp* 1611 *cr*	OK *cr lf*
Plate Trace off, do not check plate	RS *sp* 1611 *cr*	OK *cr lf*
Read Wait Plate selection	RD *sp* 1612 *cr*	x *cr lf*
Wait Plate on, stop at x-fer station until plate present resp. x-fer station is free	ST *sp* 1612 *cr*	OK *cr lf*
Wait Plate off, do not check x-fer station status	RS *sp* 1612 *cr*	OK *cr lf*
Read Plate Verify selection	RD *sp* 1613 *cr*	x *cr lf*
Plate verify on, stop at x-fer station to allow BCR read	ST *sp* 1613 *cr*	OK *cr lf*
Plate verify off	RS *sp* 1613 *cr*	OK *cr lf*
Read User Door Lock door[1] status	RD *sp* 1701 *cr*	x *cr lf*
Lock User door[2]	ST *sp* 1701 *cr*	OK *cr lf*
Unlock User door[3]	RS *sp* 1701 *cr*	OK *cr lf*
Read LED / Beeper Alarm status	RD *sp* 1702 *cr*	x *cr lf*
Activate LED- / Beeper Alarm	ST *sp* 1702 *cr*	OK *cr lf*
De-activate LED- / Beeper Alarm	RS *sp* 1702 *cr*	OK *cr lf*
Do BarCode Search Scan	ST *sp* 1710 *cr*	OK *cr lf*
Scan one cassette[4]	ST *sp* 1711 *cr*	OK *cr lf*
Turn BCR on	ST *sp* 1712 *cr*	OK *cr lf*
Turn BCR off	ST *sp* 1713 *cr*	OK *cr lf*
BCR Reset	ST *sp* 1714 *cr*	OK *cr lf*
Read Actual Slot Position n (0..2)	RD *sp* DM1 *cr*	nnnnn *cr lf*
Read Handler z-Offset (default = 600)	RD DM20 *cr*	ddddd *cr lf*
Read Handler dz Pick- & Place-Movement in Stacker (default = 500)	RD DM21 *cr*	ddddd *cr lf*
Read Handler In-Transfer z-Position (default ~42'000)	RD DM22 *cr*	ddddd *cr lf*
Read Handler z-Pitch (default =1925)	RD DM23 *cr*	ddddd *cr lf*

[1] available if door lock option present
[2] available if door lock option present
[3] available if door lock option present
[4] Requires 2-way communication feature

Read Handler Out-Transfer z-Position1	RD DM24 *cr*	ddddd *cr lf*
Read Max. Number of Levels (default = 22)	RD DM25 *cr*	ddddd *cr lf*
Read Handler dz Pick- & Place-Movement at Transfer Station (default = 800)	RD DM26 *cr*	ddddd *cr lf*
Read BCR z-Lift Read Position offset (default ~200)	RD DM27 *cr*	ddddd *cr lf*
Read Handler dz Pick- & Place-Movement at Out-Transfer Station (default = 800)2	RD DM28 *cr*	ddddd *cr lf*
Read Number of Cassettes	RD DM29 *cr*	ddddd *cr lf*
Read Handler z-Offset right carrousel (STX1000)	RD DM35*r*	ddddd *cr lf*
Read shaker speed (applies on systems with shaker option only) (default =25)	RD DM39 *cr*	ddddd *cr lf*
Read carrousel rotation speed (default =50)	RD DM38 *cr*	ddddd *cr lf*
Read number of carrousel positions	RD DM46 *cr*	ddddd *cr lf*
Read Lift z-Offset of upper carrousel (applies on Bi-level option only) (default=12'400)	RD DM47 *cr*	ddddd *cr lf*
Read number of levels of the (lower) carrousel (applies on Bi-Level option only) (default=22)	RD DM48 *cr*	ddddd *cr lf*
Read Handler Left Stacker position (default ~70)	RD DM80 *cr*	ddddd *cr lf*
Read Handler Right Stacker position (default ~940)	RD DM81 *cr*	ddddd *cr lf*
Read Handler Transfer Station position (default ~3500)	RD DM82 *cr*	ddddd *cr lf*
Set Handler z-Offset	WR DM20 *sp* d *cr*	OK *cr lf*
Set Handler dz Pick- & Palce-Movement	WR DM21 *sp* d *cr*	OK *cr lf*
Set Handler In-Transfer z-Position	WR DM22 *sp* d *cr*	OK *cr lf*
Set Handler z-Pitch	WR DM23 *sp* d *cr*	OK *cr lf*
Set Handler Out-Transfer z-Position3	WR DM24 *sp* d *cr*	OK *cr lf*
Set Handler dz Pick- & Place-Movement at Transfer Station	WR DM26 *sp* d *cr*	OK *cr lf*
Set BCR z-Lift Read Position offset	WR DM27 *sp* d *cr*	OK *cr lf*
Set Handler dz Pick- & Place Movement at Out-Transfer z-Position4	WR DM28 *sp* d *cr*	OK *cr lf*
Set Handler z-Offset right carrousel (STX1000)	WR DM35 *sp* d *cr*	OK *cr lf*
Set Carrousel rotation speed (range=1..80)	WR DM38 *sp* d *cr*	OK *cr lf*
Set Shaker Speed (range=1..50)	WR DM39 *sp* d *cr*	OK *cr lf*
Set Lift z-Offset of upper carrousel	WR DM47 *sp* d *cr*	OK *cr lf*
Set number of levels of lower carrousel	WR DM48 *sp* d *cr*	OK *cr lf*
Set left radial handler turn position at 1st stacker	WR DM80 *sp* d *cr*	OK *cr lf*
Set right radial handler turn position at 2nd stacker	WR DM81 *sp* d *cr*	OK *cr lf*
Set radial handler turn position at transfer station	WR DM82 *sp* d *cr*	OK *cr lf*

d = data (Word=16Bit)
x = 0,1 (Word=16Bit)

1 Handler: DM24 = DM22
2 Handler: DM26 = DM28
3 StoreX: DM24 = DM22
4 StoreX: DM26 = DM28

t = actual value (Word=16Bit)
s = set value (Word=16Bit)

Note !
- Z-height data memory values may be different on 3rd generation instruments due to different reduction ration of the lift drive
- Divide DM-values by four for 3rd generation instruments
- Always read DM values prior any change

1.3.4 Special Commands STP1000

Custom commands are commands defined for a specific application for. These commands will work on instruments supporting the specific application. These instruments may differ in their hardware configuration .

Special Commands for STP1000 support addressing the Tube Puncher Module (TPM).

	Command	Response
Select external TPM device for plate transfer	ST 1104 *cr*	OK *cr lf*
Select standard transfer station for plate transfer	RS 1104 *cr*	OK *cr lf*
Select lower level at TPM for plate transfer (Daughter Plate)	ST 1105 *cr*	OK *cr lf*
Select upper level on TPM for plate transfer (Mother Plate)	RS 1105 *cr*	OK *cr lf*
Read lift transfer height for plate transfer at TPM	RD DM41 *cr*	ddddd *cr lf*
Read transfer height for plate transfer at upper level (Mother Plate)	RD DM42 *cr*	ddddd *cr lf*
Read transfer height for plate transfer at lower level (Daughter Plate)	RD DM43 *cr*	ddddd *cr lf*
Set lift transfer height for plate transfer at TPM to h1	WR DM41 *sp* h1 *cr*	OK *cr lf*
Set plate transfer height for plate transfer at upper level (Mother Plate) in TPM to h2	WR DM42 *sp* h2 *cr*	OK *cr lf*
Set plate transfer height for plate transfer at lower level (Daughter Plate) in TPM to h3	WR DM43 *sp* h3 *cr*	OK *cr lf*

1.3.5 Cassette Configuration Look-up Table[1]

Up to 20 types of cassette may be stored in the Cassette Type Look-up Table (CTT). The Data Memory area of the Cassette Type Look-up Table is DM230..DM250. At this time 14 cassettes types are pre-configured:

Used	TypeNr.	Type	Pitch	Address	Content
+	0	MTP	23mm	DM230	788

[1] FW Version 7.1 and later only

+	1	DWP	50mm	DM231	1713
+	2		17mm	DM232	582
+	3		28mm	DM233	959
+	4		33mm	DM234	1131
+	5	DiTi200 ul	72mm	DM235	2467
+	6	DiTi1000 ul	110mm	DM236	3769
+	7	NTP	11mm	DM237	377
+	8		21mm	DM238	719
+	9		63mm	DM239	2158
+	10		20mm	DM240	
+	11		29mm	DM241	
+	12		35mm	DM242	
+	13		59mm	DM243	
+	14		75mm	DM244	
x	15			DM245	User
x	16			DM246	User
x	17			DM247	User
x	18			DM248	User
x	19			DM249	User
x	20			DM250	User

Although we recommend to use pre-defined cassettes types unchanged, the user may alter pre-configured cassettes types.

Each cassette location has an assigned Data Memory location that represents its type of cassette and the numbers of levels of its cassette. These values are stored in the Cassette Configuration Table (CCT). The Cassette Configuration Table is located in DM251..DM499. DM251 represents cassette location number 1, DM252 cassette location number 2 .. A total of 250 cassettes can be configured.

Each entry of the Cassette Configuration Table contains the information about the type of the cassette and the number of levels of the cassette. This information is combined in one word (16Bit), where the lower byte contains the number of levels of the cassette and the higher byte the type of the cassette.

Example:

CrPos	CsType	Type	DM23x		ValHex	ValDec
1		5/33	4	251	$040F	1039
2		8/23	0	252	$001C	28
3		2/23	0	253	$0016	22
4		8/17	2	254	$021C	540
5		5/33	4	255	$040F	1039
6		7/72	5	256	$0507	1287
7		0/50	1	257	$010A	266

A cassette is addressed through the Cassette Configuration Table by entering a negative value in DM0 (16Bit). ‘WR DM0 65535 would position the carrousel to location carrousel 1. When addressing the plate level (WR DM5 xx) the system automatically calculates the correct height of the plate by using the information of the Cassette Configuration Table.

1.3.6 Short Access Operations

The following commands allow extremely short and simple command sequences. When using short commands the plates are numbered from 1 to the maximum plate capacity. The maximum plate capacity is the value in DM25 (number of levels) multiplied with the value in DM29 (number of stackers). The way the plates are numbered can be selected. By default the plates are numbered beginning at the lowest level of stacker 1 to the top level of stacker 1, continuing at the lowest level of stacker 2 ending at the top level of stacker 2 (Vertical Mode). The optional numbering starts at the lowest level of stacker 1, lowest level of stacker 2, second level of stacker 1, second level of stacker 2, third level of stacker 1 etc. (Horizontal Mode).

	Command	Response
Import plate n	WR *sp* DM10 *sp* n *cr*	OK *cr lf*
Export plate n	WR *sp* DM10 *sp* -n *cr*	OK *cr lf*
Export plate n (unsigned alternative)	WR *sp* DM15 *sp* n *cr*	OK *cr lf*
Select Vertical Numbering Mode (default)	ST *sp* 1604 *cr*	OK *cr lf*
Select Horizontal Numbering Mode	RS *sp* 1604 *cr*	OK *cr lf*

One Short Access Command can be sent while the prior access is being executed. This second command is stacked and executed after termination of the first command.

1.4 Handling Status / Error Messages

Error Commands are used when the handling detects an internal error (e.g. when loading a plate to an occupied location). In case of error the Error Flag (1814) is set from '0' to '1'. The exact cause of an error can be found in the data memory 200 (DM200). For each type of error an error code is set in DM200. The list below shows the meaning of the error code. Errors are read by reading the content of DM200. On a time-out, first the Error Flag is read (RD 1814). Then DM200 is read in order to find the cause of error. An error is reset by sending the Reset Command (ST 1900).

Errors DM200=1xx are Import Plate Errors, errors DM200=2xx are Export Plate Errors.

	Command	Response
Read Error Flag (default =0)	RD *sp* 1814 *cr*	x *cr lf*
Read Error Code	RD *sp* DM200 *cr*	x *cr lf*

The following tables will list the StoreX status messages.

Error	Description	Code
General Handling Error	Handling action could not be performed in time.	00001 *cr lf*
Gate Open Error	Gate could not reach upper position or Gate did not reach upper position in time	00007 *cr lf*
Gate Close Error	Gate could not reach lower position or Gate did not reach lower position in time	00008 *cr lf*
General Lift Positioning Error	Handler-Lift could not reach desired level position or does not move	00009 *cr lf*
User Access Error	Unauthorized user access in combination	00010 *cr lf*

	with manual rotation of carrousel	
Stacker Slot Error	Stacker slot cannot be reached	00011 *cr lf*
Remote Access Level Error	Undefined stacker level has been requested	00012 *cr lf*
Plate Transfer Detection Error	Export operation while plate is on transfer station	00013 *cr lf*
Lift Initialization Error	Lift could not be initialized	00014 *cr lf*
Plate on Shovel Detection	Trying to load a plate, when a plate is already on the shovel	00015 *cr lf*
No Plate on Shovel Detection	Trying to remove or place plate with no plate on the shovel	00016 *cr lf*
No recovery	Recovery was not possible	00017 *cr lf*

Error	Description	Code
Import Plate Stacker Positioning Error	Carrousel could not reach desired radial position during Import Plate procedure or Lift could not reach transfer level during Import Plate procedure.	00100 *cr lf*
Import Plate Handler Transfer Turn out Error	Handler could not reach outer turn position at transfer level during Import Plate procedure.	00101 *cr lf*
Import Plate Shovel Transfer Outer Error	Shovel could not reach outer position at transfer level during Import Plate procedure.	00102 *cr lf*
Import Plate Lift Transfer Error	Lift did not reach upper pick position at transfer level during Import Plate procedure.	00103 *cr lf*
Import Plate Shovel Transfer Inner Error	Shovel could not reach inner position at transfer level during Import Plate procedure.	00104 *cr lf*
Import Plate Handler Transfer Turn in Error	Handler could not reach inner turn position at transfer level during Import Plate procedure.	00105 *cr lf*
Import Plate Lift Stacker Travel Error	Lift could not reach desired stacker level during Import Plate procedure.	00106 *cr lf*
Import Plate Shovel Stacker Front Error	Shovel could not reach front position on stacker access during Plate Import procedure.	00107 *cr lf*
Import Plate Lift Stacker Place Error	Lift could not reach stacker place level during Import Plate procedure.	00108 *cr lf*
Import Plate Shovel Stacker Inner Error	Shovel could not reach inner position at stacker plate placement during Import Plate procedure.	00109 *cr lf*
Import Plate Lift Travel Back Error	Lift could not reach zero level during Import Plate procedure.	00110 *cr lf*
Import Plate Lift Init Error	Lift could not be initialized after Import Plate procedure.	00111 *cr lf*

Error	Description	Code
Export Plate Lift Stacker Travel Error	Carrousel could not reach desired radial position during Export Plate procedure or Lift could not reach desired stacker level during Exoprt Plate procedure.	00200 *cr lf*
Export Plate Shovel Stacker Front Error	Shovel could not reach front position on stacker access during Plate Export procedure.	00201 *cr lf*
Export Plate Lift Stacker Import Error	Lift could not reach stacker pick level during Export Plate procedure.	00202 *cr lf*
Export Plate Shovel Stacker Inner Error	Shovel could not reach inner position at stacker plate pick during Export Plate procedure.	00203 *cr lf*
Export Plate Lift Transfer Positioning Error	Lift could not reach transfer level during Export Plate procedure.	00204 *cr lf*
Export Plate Handler Transfer Turn out Error	Handler could not reach outer turn position at transfer level during Export Plate procedure.	00205 *cr lf*
Export Plate Shovel Transfer Outer Error	Shovel could not reach outer position at transfer level during Export Plate procedure.	00206 *cr lf*
Export Plate Lift Transfer Place Error	Lift did not reach lower place position at transfer level during Export Plate procedure.	00207 *cr lf*
Export Plate Shovel Transfer Inner Error	Shovel could not reach inner position at transfer level during Export Plate procedure.	00208 *cr lf*
Export Plate Handler Transfer Turn in Error	Handler could not reach inner turn position at transfer level during Export Plate procedure.	00209 *cr lf*
Export Plate Lift Travel Back Error	Lift could not reach Zero position during Export Plate procedure.	00210 *cr lf*
Export Plate Lift Initializing Error	Lift could not be initialized after Export Plate procedure.	00211 *cr lf*

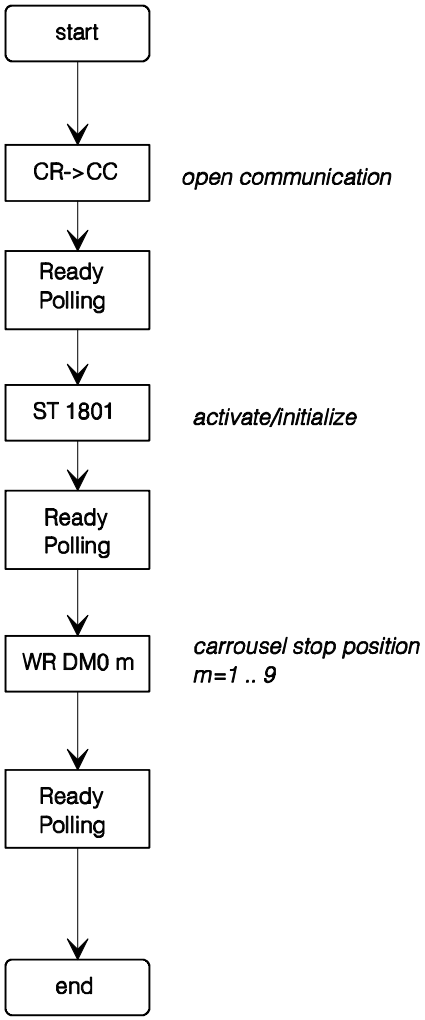
Error	Description	Code
Exit Plate Errors (1906)	Errors as above but in conjunction with Remove Plate Command	003xx *cr lf*
Barcode Read Errors (1910)	Errors as above but in conjunction with BCR Command	004xx *cr lf*
Place Plate Errors (1909)	Errors as above but in conjunction with Place Plate Command	005xx *cr lf*
Enter Plate Errors (1907)	Errors as above but in conjunction with Set Plate Command	006xx *cr lf*
Pick Plate Errors (1908)	Errors as above but in conjunction with Get Plate Command	007xx *cr lf*

The following example will show how the Ready Polling can be combined with the continuous system status request. The program below will display details of the handling actions of the StoreX handler. In combination with data base detailed status reports can be output real-time to the operator all time.

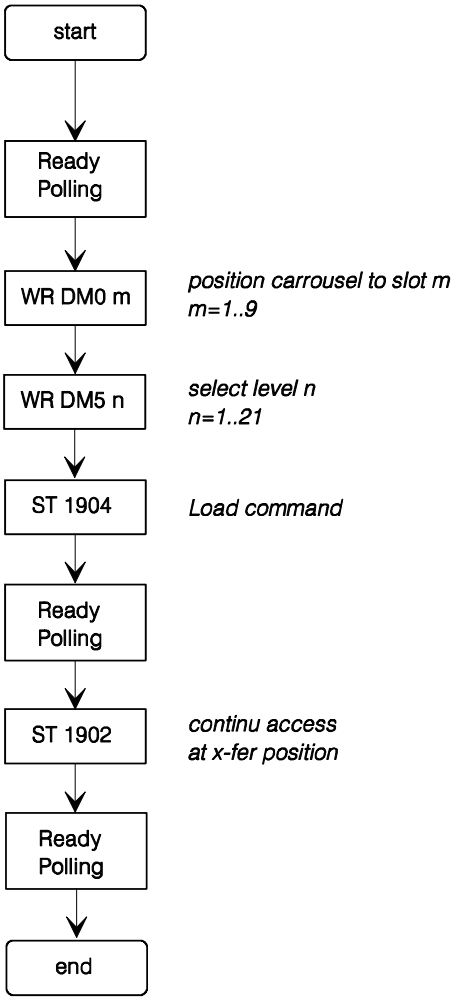
```
PROCEDURE STX_WaitReadyTrace(portNbr:INTEGER);
  VAR c:CHAR; x,y,err:INTEGER; n:WORD;
  BEGIN
    WRITE(' ');
    x:=WhereX; y:=WhereY;
    DelayMs(300);
    WHILE (STX_Read(prtNbr,1915)='0') AND NOT(KeyPressed) DO
      BEGIN
        DelayMs(100);
        GotoXY(x,y);
      Val(STX_ReadDataMemory(portNbr,200,n,err); WRITE(n:3)
         END
  END;
```

1.5 Program Examples

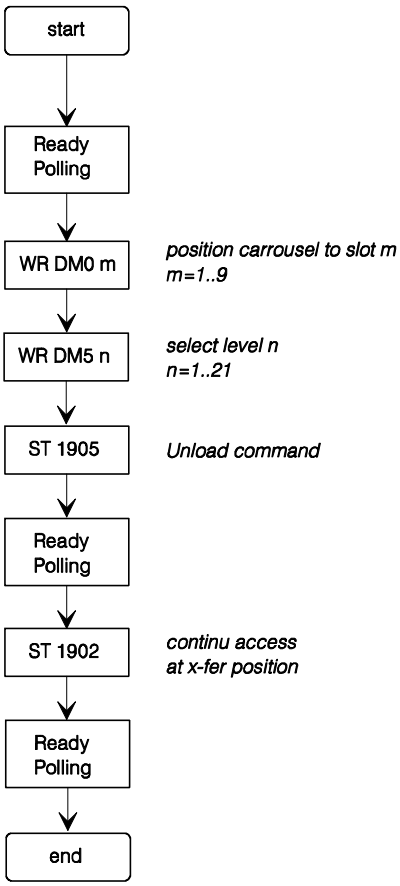
Cold Start with Positioning



Load Access to Slot m Level n



Unload Access from Slot m Level n



A universally usable procedure which can be used for most StoreX commands is given below. Use this procedure after initializing the instrument only.

```
PROCEDURE STX_DoPlate(pN,slot,level:INTEGER; command:STRING);
  BEGIN
    IF NOT(KeyPressed) THEN
      BEGIN
        STX_WaitReady(pN);
        STX_WriteDataMemory(pN,0,slot);
        STX_WriteDataMemory(pN,5,level);
        STX_Set(pN,command);
        STX_WaitReadyTrace(portNbr)
      END;
  END;
```

2. Utility Software

```
LiCONiC R&D                                                    01-06-96
C.G.Malin                                                      09:04:01

            HBI-UserSoft    *   V 0.04, 25.05.96
_____________________________________________________________________
Heraeus AG                                          HBI 2001-3
UserName                                                      S-Nbr


              Direct Commands                       1
              Monitor Flags                         2
              Macros                                3
              Teach Positioning Times               4
              Random Positioning                    5
              Random Access Cycles                  6
              Random Fast Access                    7
              Quit                                  0

_____________________________________________________________________
              Enter [0..7]                          _
```

2.1.1 Direct Commands

Direct entering of commands (ASCII-Characters) and sending to the Handling Controller. The commands can be entered as capital letters or Prompt entries by Return.

```
Enter direct commands to KV-PLC, press q to quit
_
```

2.1.2 Monitor Flags

```
P0          0000        Rot.EN      0500
Pn          0001        GateEN      0501
GtClse      0002        GateEN      0502
GtOpen      0003        Rot.Dir     0503
            0004        GtTmOut     0504
F.Door      0005        AccsLED     0505
SW 2E0      0006
SW 2E1      0007
SW 2E2      0008
SW 2E3      0009

Acs.        1100
Key Valid
InPos
Ready 1915

RotPos. DM01    Accs.  DM00     AccV.  DM02
SetSpd. DM91    aSlpe. DM92     bSlpe. DM93     intvl. DM94
_
```

Status Flag of the Handling Controller. The status message “0“ means inactive, the status message “1“ means active.

Code	Status Flag	Comment
P0	0000	0-initiator of carrousel
Pn	0001	Position-initiator of carrousel
GtClse	0002	Gate closed switch
GtOpen	0003	Gate opened switch
KeySw	0004	Key switch
Gn LED	0005	Green LED
SW 2E0	0006	Manual positioning switch bit0
SW 2E1	0007	Manual positioning switch bit1
SW 2E2	0008	Manual positioning switch bit2
SW 2E3	0009	Manual positioning switch bit3
Acs.	1100	Carrousel access (remote or manual)
Key Valid		Manual positioning switch valid
Ready	1915	Ready-Bit
Rot.Pos.	DM1	Actual carrousel position
Access.	DM0	Carrousel set position
AccV.	DM2	Access accepted

2.1.3 Macros

Sending of complete, preprogrammed command sequences.

```
Function              Command          Key

Rot. Position         WR DM0 x         1..9
Enable Rotation       WR DM0 0            0
Gate Open             ST 1901             O
Gate Close            ST 1902             C
End Access            ST 1903             E
Shaker ON             ST 1907             S
Shaker OFF            RS 1907             F
CarAct ON             ST 1801             A
CarAct OFF            RS 1801             D
Command Reset         ST 1900             R

Quit                  --               esc
Ready [1915]

Please Select ->    _
```

Code	Comment
0	No access, carrousel rotation enabled
1.9	Position carrousel
O	Gate open (only when carrousel is positioned)
C	Gate closed
R	Reset Handling Controller
Q	Exit macro menu
Accs.	Carrousel access (remote or manual)
Ready	Ready-Bit

2.1.4 Teach Positioning Times

Program to measure positioning times of the carrousel.
The program starts positioning automatically. As a result a table containing the measured positioning times is presented.

2.1.5 Random Positioning

Program for positioning the carrousel randomly without gate movements.
After entering the access intervals the program continuously simulates accesses at random positions until the “q-Key“ is pressed. As a result a table containing statistical data is presented.

If an error of positioning time larger then 0.4 seconds compared with the calibrated positioning time is observed a positioning error is assumed and monitored.

2.1.6 Random Access Cycles

Program for positioning the carrousel randomly including gate movements.
After entering the access intervals the program continuously simulates accesses at random positions until the “q-Key“ is pressed. As a result a table containing statistical data is presented.

If an error of positioning time larger then 0.4 seconds compared with the calibrated positioning time is observed a positioning error is assumed and monitored.

2.1.7 Random Fast Access

Program for positioning the carrousel randomly including combined positioning-gate movements commands.
After entering the access intervals the program continuously simulates accesses at random positions until the “q-Key“ is pressed. As a result a table containing statistical data is presented.

If an error of positioning time larger then 0.4 seconds compared with the calibrated positioning time is observed a positioning error is assumed and monitored.