SDS CCU Interface USL-MS

File: SDS_CcuSw Interface USL-MS.docx

Document Status:	released
Classification:	confidential
Author:	Paul Geiter
Project Number:	1044130023
Version / Revision:	1.1
Date of Storage:	December 13, 2017

Distribution list:

Name:	Address:	Mail:	Name:	Address:	Mail:
Kurt Eggimann	UTCH	kurt.eggimann@uster.com	Martin Cao	UTCN	martin.cao@uster.com
Rowan Sinden	UTCH	rowan.sinden@uster.com	Karen Chen	UTCH	karen.chen@uster.com
			Gang Liu	UTCH	gang.liu@uster.com

Revision history:

Rev.	Protocol version	Reason for revision	Release	
			Date:	Visa:
0.1	V1	Created	29.09.17	GEIT
0.2	V1	Reworked after VC with Sentinel team in Uster Suzhou	27.10.17	GEIT
1.0	V1	Reviewed with Gang Liu and released for implementation	17.11.17	GEIT
1.1	V1	Correction in “Response format”. Added machine status “startup”.	13.12.17	GEIT

Reference to other documents:

Pos.	Name of document and/or description		
		Date:	Author:

Table of content

Figures

Tables

1 Introduction

1.1 Purpose

This document describes the data exchange between the USTER® *QUANTUM 3* Clearer Central Unit (CCU) and the USTER® *SENTINEL* Machine Station (MS). The goal is to create a business value by combining the data gained from the ring spinning machine and the data gathered by the yarn clearer. In combination with the spindle tracking system of the winding machine each spinning cop can be qualified and the spin position can be alerted in case of quality problems.

1.2 Terms and abbreviations

CCU	Clearer Central Unit
MA	Machine (data per machine)
MS	Machine station of USTER® *SENTINEL*
SDS	System Design Specification
SP	Spindle on the ring spinning machine (data per spindle)
UDP	User Datagram Protocol
USL	USTER® *SENTINEL EXPERT*

1.3 Open points

List of open points in this document.

- ☐ Extend the list of alarm codes with winder alarms (Murata) (chapter 5).
- ☐ Define tests for cop shape and spindle quality calculation (chapters 9.6 and 9.7)

2 Network

2.1 Network topology

The overall system will look as follows.

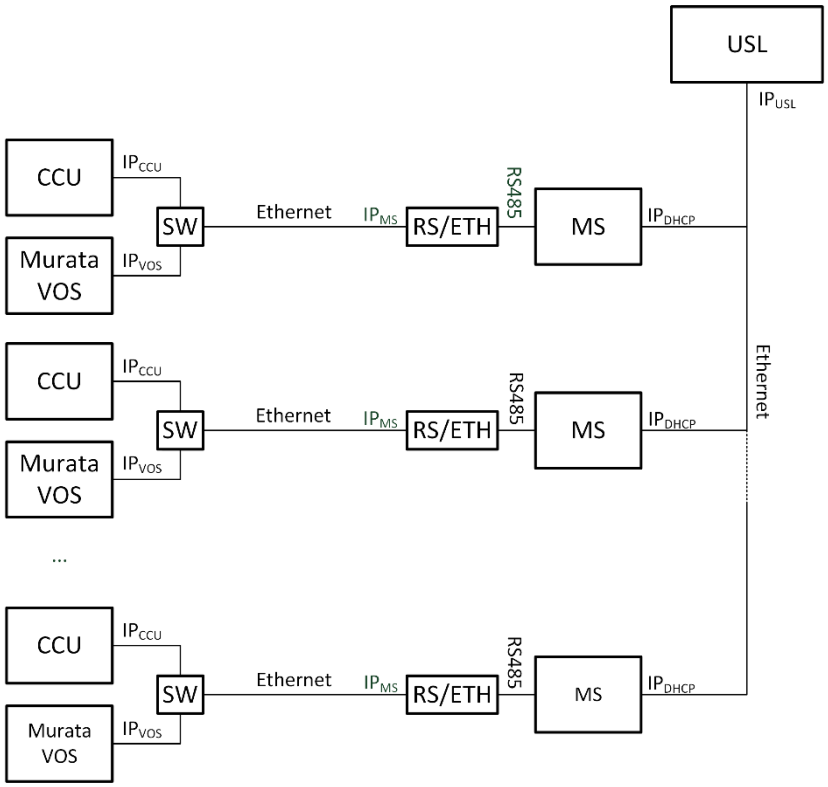


Figure 1: Network topology

USL	USTER® *SENTINEL Expert*
SW	Switch to allow Murata VOS and MS to connect to the CCU
RS/ETH	RS485-to-Ethernet converter (may be replaced by a second Ethernet port at the MS in a future version)
IP_{CCU}	Static IP address of the CCU (is identical for all CCUs)
IP_{VOS}	Static IP address of the Murata VOS (is identical for all VOS)
IP_{MS}	Static IP address of the MS resp. RS485-to-Ethernet converter (is identical for all MS)
IP_{DHCP}	Dynamic IP address of the MS (is different for each MS)
IP_{USL}	IP address of the USL PC

2.2 Network connector at CCU

The CCU connects via Ethernet (max. 100 Mbit/s). The connector is on the back side of the CCU and is called “Machine Ethernet”.

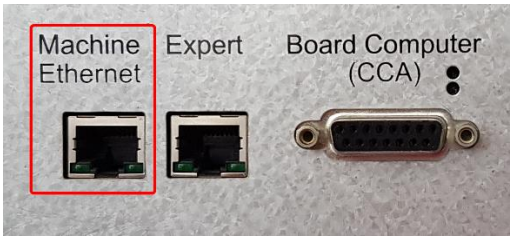


Figure 2: Machine Ethernet connector at CCU

The network configuration of the CCU (IP address, subnet mask, port number) and the target address of the MS are application specific and will be defined during CCU configuration. The addresses are fixed and not changeable at runtime.
When the CCU is also connected to the winding machine control unit, an external switch or hub must be used.

2.3 Network connector at MS

On MS side, the only Ethernet port is used for USL communication and can't be used for CCU connection. The SENTINEL Expert network works with DHCP. The CCU has always the same fixed IP address and therefor it is not possible to connect it to the SENTINEL Expert network.
To create a second Ethernet port, an RS485-to-Ethernet converter will be used.
In a future, redesigned MS version, the converter will be replaced by a second Ethernet port at the MS. The protocol will remain the same on this change.

2.4 IP Addresses

CCU IP address:	192.168.188.1
MS IP address:	192.168.188.100
Subnet mask:	255.255.255.0
Port number:	2876

When using the RS484 to Ethernet adapter, it has to be configured with the appropriate information.

3 Protocol

3.1 General

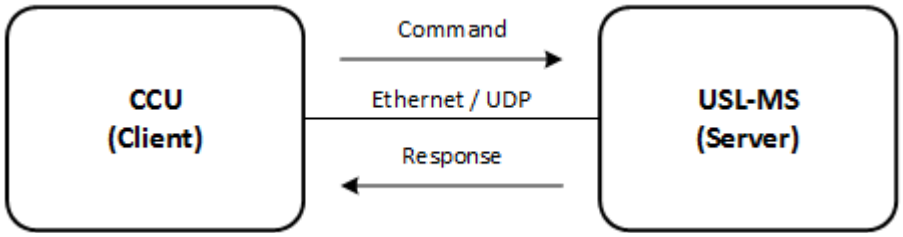


Figure 3: Client / Server definition

The CCU works as Client and requests data sending commands to the MS which is the Server. It reacts to the request with a response message.

Due to poor CPU and memory resources, a UDP based command / response protocol will be used. The MS is the Server and the CCU the Client.
For the same reason, the packet size is limited to 255 bytes. This is the maximum length of a message (command resp. response) including the header and CRC.

3.2 Definition of data frames

3.2.1 Byte order

Values that consist of more than 8 bits (16 or 32 bits) are transmitted high byte first (big-endian).

3.2.2 CRC calculation

The CRC is calculated over the header and the data (if data length is > 0). We use the CRC16 algorithm that is already available in the MS. This is the same algorithm as in the Modbus. The polynomial is $1 + x_2 + x_{15} + x_{16}$. The seed value is 0xFFFF.
For more information see also

- www.modbus.org
- http://www.modbus.org/docs/Modbus_over_serial_line_V1_02.pdf
- https://stackoverflow.com/questions/35495547/crc16-checksum-calculation-which-one-is-correct/35495797#35495797

3.2.3 Command format

Data going from CCU to MS are called “command” and have the following data structure:

Field	Bytes	Description	
0xAA	1	Every command starts with this signature. It is used for synchronization of the parser.	Command Header
0x55	1		
Command ID	1	Unique ID of the command	
Command Version	1	Protocol version	
Data length	2	Number of bytes of the following data	
Data	0-xxx	Application Data. Depends on the command.	
CRC	2	CRC includes all bytes except the CRC itself.	

Table 1: Command format

3.2.4 Response format

The response is always sent from the MS to the CCU as an answer to a command. The Command ID of the response is always the same as of the corresponding command.

Field	Bytes	Description	
0xAA	1	Every command starts with this signature. It is used for synchronization of the parser.	Command Header
0x55	1		
Command ID	1	Unique ID of the command	
Error Code	1	Result of the command	
Data length	2	Number of bytes of the following data	
Data	0-xxx	Application Data. Depends on the command. On error, no data is sent.	
CRC	2	CRC includes all bytes except the CRC itself.	

Table 2: Response format

3.3 Timing

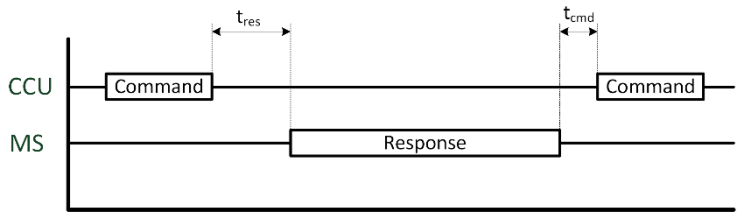


Figure 4: Protocol timing

Name	Time	Description
t_{res}	< 1s	This is the maximum time the CCU waits for a response. If the MS doesn't answer within this time, the communication timeout is invoked.
t_{cmd}	> 200ms	This is the minimum time the MS can accept a new command after the response has been sent.

Table 3: Protocol time definition

4 Connection mechanism

4.1 Connection procedure

- The connection is always invoked by the CCU with the command USL_CONNECT. In the command data, the highest supported protocol version of the CCU is provided. The MS should always accept the USL_CONNECT command, independent of the protocol version in the command header.
- The MS responds with the highest protocol version it supports.
- After that, the CCU operates with the highest common protocol version. All commands are used with this protocol version.
- The CCU can close the connection with the command USL_DISCONNECT.

4.2 CCU connection states

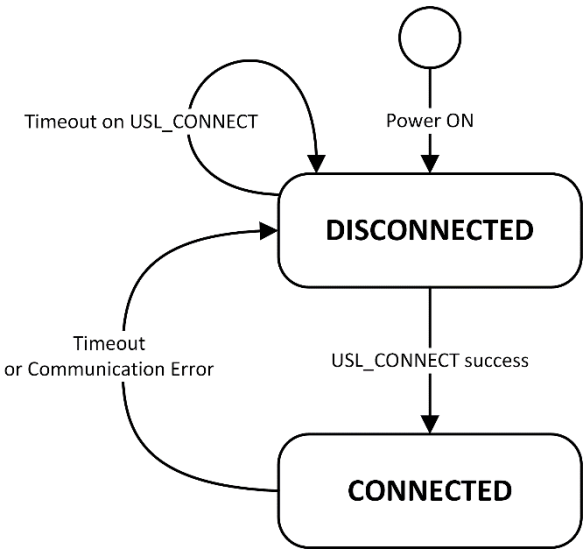


Figure 5: CCU connection states

Chapter 4.1 ”Connection procedure” describes the detailed connection procedure.

State	Description
DISCONNECTED	This is the state after power up or broken communication due to a communication error. *Actions:* - Periodically send command USL_CONNECT until MS answers. - If MS answers, evaluate the highest common protocol version.
CONNECTED	The connection is established and OK. *Actions:* - Synchronize all data between CCU and MS. - Normal operation using the common protocol version.

Table 4: CCU connection states

4.3 MS connection states

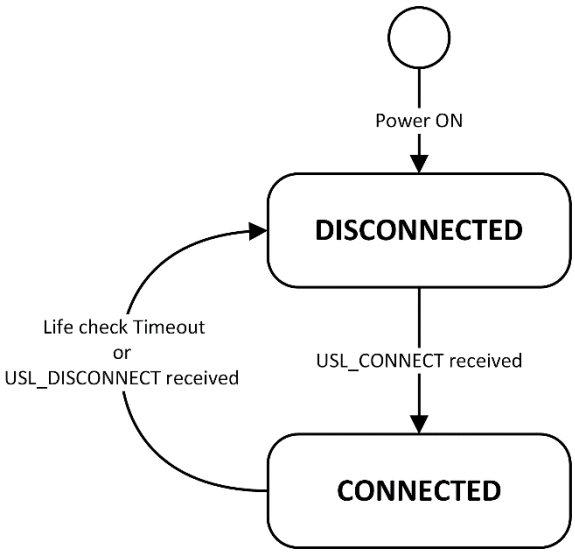


Figure 6: MS connection states

Chapter 4.1 ”Connection procedure” describes the detailed connection procedure.

State	Description
DISCONNECTED	This is the state after power up, broken communication due to a communication error or a disconnection by the CCU. *Actions:* - Wait for the USL_CONNECT command. - Don’t accept other commands. - Respond with the highest supported protocol version.
CONNECTED	The connection is established and OK. *Actions:* - Normal operation. The client uses the highest common protocol version.

Table 5: MS connection states

4.4 Version and compatibility handling

As mentioned in the previous chapter, both systems can support more than one protocol version. To achieve a stable and compatible communication, each system provides his highest supported protocol version during the connection procedure.
The CCU takes the highest common protocol version for communication with the MS.
With each command, the protocol version is given as a parameter in the command header.

Note: Both systems must support all protocol versions that have ever been defined.

4.5 Life check

Each side can detect if the connection is broken. The life check works as follows:

MS: Expects at least every t_{cmd} a command from the CCU.
Conditions for broken or terminated connection:
- No command received during t_{cmd}
- Command USL_DISCONNECT received

CCU: Calls at least every t_{trig} the status request or any other command to trigger the MS life check timer.
Conditions for broken or terminated connection:
- No reply within t_{res} (timeout)
- CCU terminates with USL_DISCONNECT

Name	Time	Description
t_{cmd}	30s	Interval the MS expects at least one command.
t_{trig}	15s	Interval the CCU sends at least one command ($t_{MS} > t_{trig}$)
t_{res}	see chapter 3.3	Maximum time the CCU is waiting for a response to a command. After that, a timeout is detected and the connection is closed.

Table 6: Life check times

4.6 Error handling

4.6.1 CRC Error

A CRC error can occur by disturbances on the communication line. They probably are detected by the UART and may be handled there. In any case, the CRC check will find such errors in the data stream.
If the MS detects a CRC fault in the received command, it will respond with the error code USL_ERR_CRC.
If the CCU receives error code USL_ERR_CRC or its CRC calculation fails, it will repeat the command.
After three subsequent trials, the CCU will terminate the connection and then tries to re-establish it.
For diagnostics, the CCU has a counter for CRC errors.

4.6.2 Receive timeout

If the MS doesn't answer within the define time (t_{res}, see chapter 3.3), this is a receive timeout. The connection will be terminated and the CCU tries to reconnect to the MS.
For diagnostics, the CCU has a counter for receive timeouts.

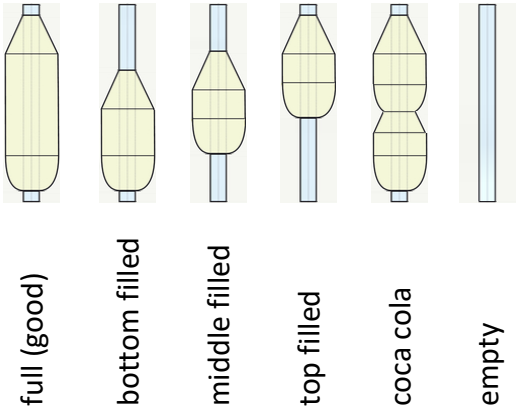
5 Data definition

This data list describes all data going between CCU and MS. The column “Scope” shows if the data item exists once per machine (MA) or per ring spindle (SP). See chapter 6.2 “Commands” for exact definition of data types.

Item	Scope	Description	Direction
Machine type	MA	The type of the ring spinning machine as a string signature.	CCU←MS
Machine ID	MA	The identification of the ring spinning machine in the mill. It’s the number or name as a string.	CCU←MS
Number of spindles	MA	The physical number of ring spinning positions on the ring spinning machine.	CCU←MS
Spindle number mapping	MA	Information about the mapping between physical to the logical spindle number which is shown to the user. See chapter 6.1 “Spindle numbering” for more details.	CCU←MS
Machine status	MA	Information about the current machine status. Possible states are: - unknown (this state is transient e.g. after power loss) - standing - startup - running - doffing	CCU←MS

Spindle data valid	MA	Indicates spindle data availability. For each type of spindle data a flag exists that tells the CCU that this data is available and ready to be transferred. It can also happen, that the data of a doff could not be calculated (e.g. due to power loss) or the feature is not yet available. The flags will be cleared at the start of a doff and will be set after the doff has been finished and the data have been calculated. The following flags exist: - Cop shape available - Spindle quality available Note: The end break data is always available during production. See chapter 7.3 “Spindle data valid”	CCU←MS
Yarn count	MA	[Nm] Yarn count of the currently producing yarn. The value is 0 if not yet known in the MS.	CCU←MS
Time to doff	MA	Remaining time until the next doff [s]. There are different sources of the doff time: - MS gets this time from the ring spinning machine. - MS gets it from the last doff and probably adjusts it during several doffs. - The USL calculates it from the spinning parameters. - The USL estimates it from the same article already produced in the past. - For a new article USL takes a similar article that has been produced before. The value is 0xFFFF if not known in the MS (e.g. impossible to evaluated or power loss).	CCU←MS
		If a valid doff time has been received before, the CCU can restore it in the MS if 0xFFFF has been received. This is only done when the machine status is “running”. See chapter 7.1.4 “Behavior after power up” for more information.	CCU→MS

Doff number	MA	The ID number of the doff. For each doff it is incremented when the doff starts to produce. On overflow, it starts at 1. The value is 0 if not known in the MS (e.g. due to power loss).	CCU←MS
		If a valid doff number has been received before, the CCU can restore it in the MS if 0xFFFF has been received. See chapter 7.1 “Doff number” for more information	CCU→MS
Temperature	MA	Temperature on the ring spinning machine [°C]. The resolution is 0.1 °C. The value is 0 if not known in the MS (e.g. the temperature sensor is not connected).	CCU←MS
Humidity	MA	Humidity on the ring spinning machine [%rh]. The resolution is 0.1 %rh. The value is 0 if not known in the MS (e.g. the humidity sensor is not connected).	CCU←MS

Cop shape	SP	Information about the buildup of a spinning cop.  On USL/MS side, the shape has to be classified according to its own rules. The value “unknown” is set if the cop shape can’t be evaluated by the MS. This data is available at the end of the doff for <u>each</u> spindle.	CCU←MS
Slip spindle	SP	The spindle was not always rotating with the nominal speed. This results in bad twist and therefore bad quality of the yarn. This data is available at the end of the doff for <u>each</u> spindle.	CCU←MS
Rogue spindle	SP	There are a lot of end breaks on the spin cops. This data is available at the end of the doff for <u>each</u> spindle.	CCU←MS
Off quality spindle	SP	Bad quality of the cop. The number of end breaks per time has exceeded a given limit. This data is available at the end of the doff for <u>each</u> spindle.	CCU←MS
End breaks	SP	Counter of the end breaks. If the end breaks of a spindle could not be counted, the counter remains 0. This data is available any time during production for <u>each</u> spindle.	CCU←MS

<table>
<tr><td></td><td></td><td>During startup the startup end breaks are collected. The end of the startup phase is indicated by the machine status that changes from “startup” to “running”.</td><td></td></tr>
<tr><td>Alarm</td><td>SP</td><td>If the CCU or the connected winding machine detects a cop with bad quality, the alarm is invoked in order to switch on the signal lamp at the ring spindle and to stop it if “Rowing stop” is installed.
Additional, the reason of the alarm is sent as a string code which may be mapped to a localized text (in the USL). If unknown, the string code can be displayed instead.
Definition of the alarm codes:
<table>
<tr><th>Code</th><th>Description</th></tr>
<tr><td>NSA</td><td>Too many neps and short thick places</td></tr>
<tr><td>LA</td><td>Too many long faults</td></tr>
<tr><td>TA</td><td>Too many thin places</td></tr>
<tr><td>CA</td><td>Too many yarn count deviations at startup</td></tr>
<tr><td>CCA</td><td>Too many yarn count deviations during winding</td></tr>
<tr><td>PFA</td><td>Too many periodic faults</td></tr>
<tr><td>OCA</td><td>Too many off center core faults</td></tr>
<tr><td>MCA</td><td>Too many missing core faults</td></tr>
<tr><td>SVA</td><td>Too many shade variation faults at startup</td></tr>
<tr><td>CSVA</td><td>Too many shade variation faults during winding</td></tr>
<tr><td>CVA</td><td>Too high CV compared with group average</td></tr>
<tr><td>HA</td><td>Too high hairiness compared with group average</td></tr>
<tr><td>YA1</td><td>Too many N,S,L,T,C,CC faults</td></tr>
</table></td><td>CCU→MS</td></tr>
</table>

		…	[tbd] Faults detected by the winder	
		This data is sent to the MS as soon as it is detected.		

Table 7: Data definitions

6 Command definition

6.1 Spindle numbering

Here, the spindle number naming of the USL system is used.

6.1.1 Side naming

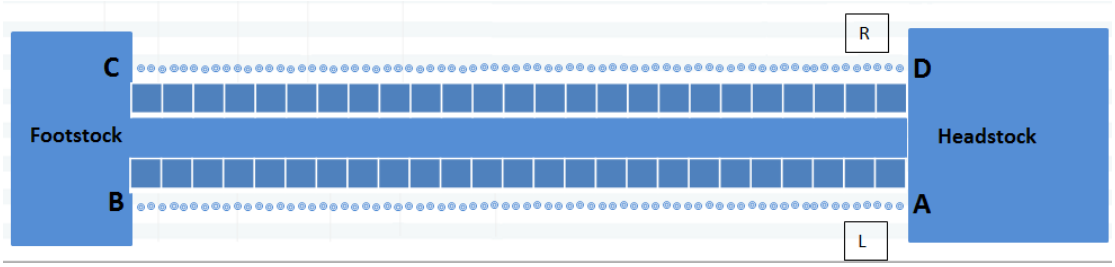


Figure 7: Machine side naming

Left (L) = AB, as seen from the machine headstock.
Right (R) = DC, as seen from the machine headstock.

6.1.2 Number naming

We use the letters A, B, C and D to define the numbering pattern (e.g. “ABDC”). There are 12 possible combinations. The following figures shows the numbering combinations for a machine with 1200 spindles. If the name of the numbering pattern has a “/”, this indicates independent spindle numbering on each side (e.g. “AB/DC”).

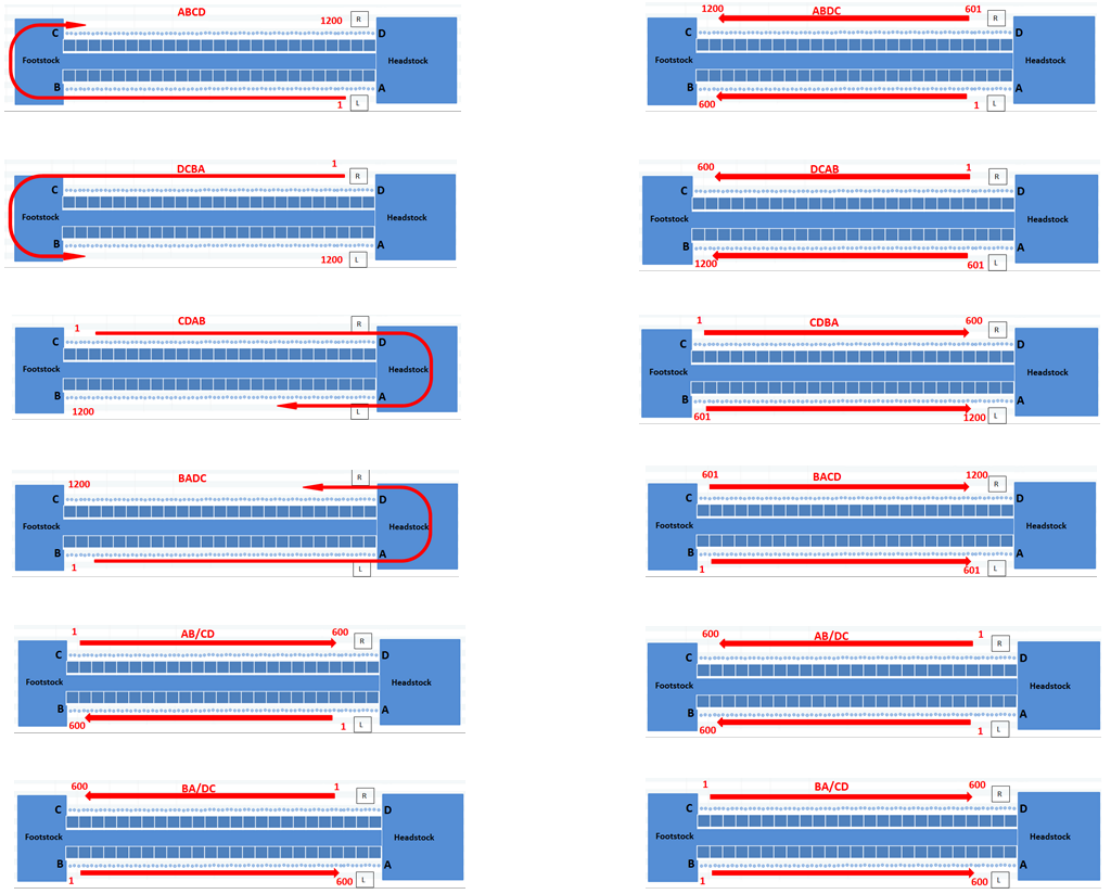


Figure 8: Spindle numbering combinations

6.1.3 Internal numbering

Internal in the CCU and in the communication to the MS, the spindle numbering “ABDC” is used. The naming pattern is only used to display the spindle number on the CCU screen (e.g. in the alarm screen).

6.2 Commands

In this chapter, the exact data definition is given for each protocol version.
The command header and the CRC are not shown here.

Data frame definition see chapter 3.2 “Definition of data frames”.
Data definition see chapter 5 “Data definition”.

The following conventions are used in the following table:

- uint8 unsigned integer 8 bit (1 byte)
- int8 signed integer 8 bit (1 byte)
- uint16 unsigned integer 16 bit (2 bytes)
- int16 signed integer 16 bit (2 bytes)
- uint32 unsigned integer 32 bit (4 bytes)
- int32 signed integer 32 bit (4 bytes)
- string[x] zero (null) terminated character array of max size x (including null terminator)
- [x] array size

Command Name	Cmd ID	Protocol versions	Command Data			Response Data		
			Name	Type	Remarks / Unit	Name	Type	Remarks / Unit
USL_CONNECT	0x01	0[1)]	max version	uint8	Highest supported protocol version of CCU	max version	uint8	Highest supported protocol version of MS
USL_DISCONNECT	0x02	0[1)]	---			---		
USL_LIFECHECK	0x03	0[1)]	---			---		
USL_CONFIG_GET	0x04	1	---			machine type	string[20]	
						machine ID	string[20]	
						number of spindles	uint16	
						spindle number mapping	string[6]	"ABCD", "ABDC", "AB/CD" … see chapter 6.1 "Spindle numbering"
USL_STATUS_GET	0x05	1	---			machine status	uint8	0: unknown 1: standing 2: startup 3: running 4: doffing
						spindle data available	uint8	Bit field (1 = available) Bit 0 (LSB): cop shape Bit 1: spindle quality data Bit 2-7: spare
						yarn count	uint32	[0.0001Nm] 0 = invalid
						time to doff	uint16	[s] 0xFFFF = invalid
						doff number	uint16	0 = invalid
						temperature	uint16	[0.1°C] 0 = invalid

						humidity	uint16	[0.1%rh] 0 = invalid

USL_COP_SHAPE_GET[3)]	0x10	1	spindle ID first	uint16	First spindle ID of the requested range	spindle ID first	uint16	Spindle ID of the first data item
						length[2)]	uint16	Number of data items x
						cop shape [0..x-1]	uint8[x]	0: unknown 1: full 2: bottom filled 3: middle filled 4: top filled 5: coca cola 6: empty
USL_COP_QUALITY_GET[3)]	0x11	1	spindle ID first	uint16	First spindle ID of the requested range	spindle ID first	uint16	Spindle ID of the first data item
						length[2)]	uint16	Number of data items x
						cop quality [0..x-1]	uint8[x]	Bit field (1 = set) Bit 0 (LSB): Off quality spindle Bit 1: Rogue spindle Bit 2: Slip spindle Bit 3-7: spare
USL_END_BREAKS_GET[3)]	0x12	1	spindle ID first	uint16	First spindle ID of the requested range	spindle ID first	uint16	Spindle ID of the first data item
						length[2)]	uint16	Number of data items x
						counter [0..x-1]	uint8[x]	Counter
USL_ALARM_SET	0x80	1	spindle ID	uint16	Spindle number that has an alarm	---		
			alarm code	string[10]				

USL_STATUS_RESTORE	0x81	1	time to doff	uint16	[s] 0xFFFF = invalid Restore the time to doff in case the MS sends 0xFFFF.	---
			doff number	uint16	0 = invalid Restore the doff number in case the MS sends 0.	

Table 8: Command definitions

Remarks:

1) Protocol version 0 is supported in all protocol versions of MS and CCU
2) The maximum number is limited by the packet size
3) For all spindle data, the same command has to be called several times (block wise, with different values of “spindle ID first” and dependent of the received length). The MS responds always with the maximum size of spindles the packet size allows.

6.3 Error codes

Error codes returned by the MS in the command header.

Name	Code	Description	Returned by
USL_ERR_OK	0x00	No error, the command call was successful.	All
USL_ERR_DISCONNECTED	0x02	The CCU tries to call a function in disconnected state.	All except USL_CONNECT
USL_ERR_CRC	0x03	The MS has detected a CRC error in the command.	All
USL_ERR_COMMAND	0x04	Invalid command. The MS has no command for the given protocol version.	All, except the following: - USL_CONNECT - USL_DISCONNECT - USL_LIFECHECK
USL_ERR_BUSY	0x05	The MS is busy. Try again.	All
USL_ERR_SPINDLE	0x10	The spindle number given in the command is out of range.	USL_COP_SHAPE_GET USL_QUALITY_GET USL_END_BREAKS_GET USL_ALARM_SET
USL_ERR_WRONG_ARG	0x11	An argument was wrong or out of range. For spindle related commands, the error USL_ERR_SPINDLE is returned if the spindle number is out of range.	not yet defined
USL_ERR_NOT_AVAIL	0x12	The requested data is not available because of data loss or not yet implemented feature.	USL_COP_SHAPE_GET USL_COP_QUALITY_GET USL_END_BREAKS_GET
USL_ERR_GENERAL	0xFF	An unspecified error occurred in the MS. The CCU will retry the command if this happens.	All

Table 9: Error codes

7 Behavior and special conditions

7.1 Doff number

See also chapters 5 and 6.2.

7.1.1 Definition

The doff number identifies the cops produced and allows the winder and the CCU to track different doffs. The doff numbers should be unique for all cops that are in the winding process. This can include more than one doff due to storage capacity in the winding machine.
The following numbers are defined:

0: indicates an invalid doff number
1…65535: valid doff number

7.1.2 Initial value

The first doff has the number 1. It is set when the doff starts to produce.

7.1.3 Incrementing

The doff number is incremented when a new doff starts to produce. Incrementing 65535 will result in 0 for a 16 bit value. In this case, the doff number must be set to 1.

7.1.4 Behavior after power up

There are two different scenarios, dependent on the version of the MS.

- If the MS has a persistent memory (future version), the doff number is restored and remains the same as before the power loss.

- If the MS has no persistent memory (current version), the doff number is set to 0.
  If the CCU gets doff number 0, it will restore the one it has stored with the command USL_STATUS_RESTORE.
  If the CCU has also doff number 0, it reacts as follows:
  - If the machine is standing: The CCU will not restore the number (doff number remains 0 in the MS until the machine is running).
  - If the machine is running: The CCU restores the doff number 1 in the MS.

7.2 Time to doff

7.2.1 Definition

The time to doff is the remaining time until the cops are full and the doff is finished. The unit is seconds and is stored in a 16 bit variable. This allows a range from 0 to 65535 (0x00 to 0xFFFF) seconds.
The value of 0xFFFF has a special function. It designates an invalid time to doff.

7.2.2 Calculation

The time to doff is calculated in the MS. The initial value at doff start can come from different sources:

- Directly form the ring spinning machine.
- From the last doff of the same article.
- From the USL at article start.
  - For a known article the doff time has been stored
  - For a new article, the doff time can be evaluated from the spinning parameters or from a similar known article.

7.2.3 Behavior after power up

There are two different scenarios, dependent on the version of the MS.

- If the MS has a persistent memory (future version), the time to doff is restored and remains the same as before the power loss.

- If the MS has no persistent memory (current version), the time to doff is set to 0xFFFF.
  If the CCU gets doff number 0xFFFF, it will restore the one it has stored with the command USL_STATUS_RESTORE.
  If the CCU has no valid time to doff, it will not restore it. This means that no time do doff is available for this doff. This will also be communicated to the winding machine.

7.3 Spindle data valid

See also chapter 5 “Data definition”.
In general, the “Spindle data valid” flags indicate that the spindle data is ready for reading by the CCU.
The data valid flags will be cleared when the doff begins to produce.
When the machine has finished the doff the flags will be set as soon as the MS has calculated the values of each spindle.

8 Setup and debugging information

8.1 Log filter

For all log output of the MS-CCU communication, there is a special log filter defined: LOG_USL. This filter can be set with the QC-Terminal in the “Log filter” menu or the tool button filter ().

8.2 Set IP Address of the CCU

At the CCU, the IP address of the machine Ethernet connection is defined by the environment variables `MA_ETHERNET_IP` and `MA_ETHERNET_MASK`.

8.3 Check IP Addresses of the CCU

Show the IP addresses of the network interfaces.

`ksPci0`: EXPERT connection

`ksPci1`: Machine Ethernet connection (used for winding machine integration and USL-MS communication)

Command: `-> ifconfig`
Output:

```
…
ksPci0    Link type:Ethernet  HWaddr 40:ec:f8:00:05:3c  Queue:none
    inet 10.128.30.145  mask 255.255.240.0  broadcast 10.128.31.255
    inet 224.0.0.1  mask 240.0.0.0
    UP RUNNING SIMPLEX BROADCAST MULTICAST
    MTU:1500  metric:1  VR:0  ifindex:2
    RX packets:61633 mcast:40035 errors:0 dropped:5555
    TX packets:1629 mcast:0 errors:0
    collisions:0 unsupported proto:0
    RX bytes:6690k  TX bytes:112k

ksPci1    Link type:Ethernet  HWaddr 40:ec:f8:00:05:3d  Queue:none
    inet 192.168.188.1  mask 255.255.255.0  broadcast 192.168.188.255
    inet 224.0.0.1  mask 240.0.0.0
    UP RUNNING SIMPLEX BROADCAST MULTICAST
    MTU:1500  metric:1  VR:0  ifindex:3
    RX packets:1169 mcast:32 errors:0 dropped:0
    TX packets:1234 mcast:0 errors:0
    collisions:0 unsupported proto:0
    RX bytes:91k  TX bytes:208k
```

8.4 Set IP Address and port number of the MS in the CCU

The IP address and port number of the MS are defined by environment variables in the CCU.

- `USL_PEERADDR` defines the IP address
- `USL_PORT` defines the port number

9 Testing

9.1 QC-Terminal commands on CCU

uslShow()	Shows the current state of: - Connection - Used protocol version - Machine ID - Number of spindles - Spindle number mapping - Machine status - Spindle data available - Yarn count - Time to doff - Doff number - Temperature - Humidity
uslSpDataShow(first, last)	Shows a list of all spindle data in the CCU from spindle “first” to spindle “last”.
uslLogLevelSet(x)	Set the log level of the USL module. 0: default (lowest level) 1: show detailed data of the communication (send and receive)
uslAlarmSet(sp, “code”)	Send an alarm to the MS for spindle “sp” and the alarm code (must be in “”, e.g. uslAlarmSet(123,”NSA”)).

9.2 MS side

On the MS, different operation conditions must be invoked.

- Machine status changes
- Machine data simulated (Machine type, Machine ID, Number of spindles, Spindle number mapping, Spindle data valid, Yarn count, Time to doff, Doff number, Temperature, Humidity)
- Spindle data simulated (Cop shape, Slip spindle, Rogue spindle, Off quality spindle, End break counter)
- Alarm from CCU

9.3 Connection tests

Test behavior on startup and when the cable is disconnected.

9.4 Power loss

Test the correct handling of the doff number and the time to doff when:

- The MS is switched off and on
- The CCU is switched off and on
- Both systems are switched off and on simultaneous

Test this for machine standing and machine running.

9.5 Calculation of time to doff

This test should verify if the time to doff is calculated correctly for an existing article and for a new article.

9.6 Calculation of the cop shape

[tbd] Define the tests together with USL

9.7 Calculation of the spindle quality data

[tbd] Define the tests together with USL