STRICTLY CONFIDENTIAL

PROJECT

Gas Lift

TITLE

SBC Requirements

ISSUE STATUS

Issue	Author	DATE	Release status	Change request	AMENDMENTS
A	P.Hockley	18/5/2010	Superseded	N/A	RE-ISSUED WITH NEW NUMBER
B	DM Reaves	19/7/2010	In-Review	N/A	Implement comments from 18Jun10 & 07Jul10 meetings

TABLE OF CONTENTS

1 PURPOSE

The purpose of this document is to define both the hardware and software requirements of the Single Board Computer (SBC) in the SCS Enclosure. It will also define all interface Protocols where these have not already been defined elsewhere.

This document should be used by internal or third party software developers to write appropriate software to drive the SBC.

2 HARDWARE REQUIREMENTS

The following hardware is required as a minimum.

- RS485, One Channel is required for the MODBUS Interface
- RS232, Three Channels required, two for SCS Board Interfaces and one for a configuration interface to set up communications.
- USB 2.0, Two channels required, one for Memory Stick data storage and one for possible future expansion eg for a USB/RS232 converter if a third SCS board interface is added.
- LAN (10/100 Base – T), one channel for the Diagnostics Port.
- +5V or +12V DC Input supply, power consumption less than 3 Watts.
- Dimensions no larger than 100mm per side.

The SBC processor is the Data sound Lab ICOP-6154 Ultra Low Power Embedded PC104 Module with 300MHz Vortex86SX CPU module, running WindowsCE 6.0BSP, or Linux.

System Memory: 128MB DDR2

IDE External Memory: to be defined by software implementer, if required, such that at least 100% spare capacity will exist after successful handover of the software implementation. It is preferable that both the operating system and application will be booted from a memory stick, initial estimate 20Mb (software implementer: please update).

3 SOFTWARE REQUIREMENTS

This section describes all of the software functions required.

Where the word “parameter” is used, a mechanism shall be provided for the value to be changed within reasonable limits to be defined by the implementer, without rebuilding the

software, eg via hyperterminal or a text-based parameter file. It is acceptable that a reboot would be required after any change.

Wingpath are requested to identify any mathematical steps that this specification and its references imply which could involve excessive processing time, so that Camcon Technology can advise on the minimum level of accuracy that is needed in each case.

3.1 SCS BOARD INTERFACES

These are simple RS232 based interfaces, to connect the SBC to the PIC in each of up to three Surface Control Systems (SCS).

The interface settings are parameters: 9,600 bits per second, 1 start bit, 8-bit data, 1 stop bit, no parity. Bytes are sent in the order described below and multi-byte words are sent most significant byte first. No ASCII ACKs or NAKs are sent in either direction.

Normally, the SBC forwards commands received from the MODBUS to the SCS which returns a status response message. The SBC shall not send any message to the SCS whilst CTS from the SCS is false. This occurs in local mode (see below) and whenever the SCS is in dialogue with the DHCS due to a previous SBC command.

In local mode, CTS from the SCS will be false and the SCS sends unsolicited status messages to the SBC which in this mode neither initiates commands nor acknowledges the status messages.

3.1.1 SBC Command Messages

The format of an SBC command to the SCS is as follows.

Byte 1: Mandrel Number
Byte 2: Command Byte 1
Byte 3: Command Byte 2
Byte 4: Checksum

Mandrels 1 to 3 are on the first SCS, 4 to 6 on the second and 7 to 9 on the third. The initial hardware design will only implement the first SCS (3 mandrels).

3.1.2 Command Data Structures

The table below gives the various commands

COMMAND	Command Byte 1	Command Byte 2 (Range)
OPEN ACTUATOR	0x01	0x01 to 0x06
CLOSE ACTUATOR	0x02	0x01 to 0x06
GET STATUS	0x03	0x00
SET ACTUATOR PULSE TIME	0x04	0x00 to 0x03

COMMAND	Command Byte 1	Command Byte 2 (Range)
GET DATA	0x05	0x00
MANDREL CONFIGURATION	0x06	0xnn where 'nn' is the number of active mandrels.

The mandrel number byte is 0x01 to 0x0n for mandrels 1 to n respectively. The pulse times corresponding to the 4 possible command values are defined in DL 100007 (DHCS firmware description - check when issuing).

Notes:

The numbers used for the mandrel number byte are internal to the SCS box and are related to the mandrel MODBUS slave addresses via the SBC configuration interface.

Open Actuator and Close Actuator each cause the SCS to send an appropriate command to the DHCS which will respond with status that updates the data held by the SCS which sends a status message to the SBC.

Get Status causes the SCS to send a command to the DHCS to respond with status without changing any actuator position or pulse time. The response updates the data held by the SCS which sends a status message to the SBC.

Get Data is a command to the SCS Board to return the DHCS status data currently held by it without requesting an update from the DHCS. Note: this command may be superfluous

Set Actuator Pulse Time causes the SCS to set the pulse time within its own non-volatile memory for the mandrel, ie for all actuators together. Its default value is TBD (index value corresponding to 10mS) and is variable via the SBC configuration interface, separately for each mandrel. It is not user-variable. The SCS may use different pulse times under special circumstances (eg minimum for any command to move to the current position and maximum for a sticking actuator).

3.1.3 SCS Status Messages

The format of an SCS status message to the SBC is as follows:

DATA RECEIVED	FORMAT
Mandrel Number (1 byte)	The mandrel to which this message refers.
Command (1 byte)	The most recent command sent to DHCS in the format defined in DL100008 (SCS - DHCS comms description), or 0x00. On start-up, the PIC will send no status messages before attempting to interrogate the DHCS.
SCS Local Mode Status (1 byte)	0x01 if the SCS box is in Local Mode, otherwise 0x00
DHCS Communications Status (1 byte)	Communication Status of SCS - DHCS link (see DL 100001 MODBUS Protocol)
DHCS Actuator Status (1 byte)	Bit 1 (lsb) represents actuator 1 and is set to "1" if the actuator is "open". Similarly for bits 2 to 6. Bits 7 & 8 are always set to "0"
CBV Start (1 byte)	Capacitor Bank Voltage Start 0-255

DATA RECEIVED	FORMAT
CBV Finish (1 byte)	Capacitor Bank Voltage Finish 0-255
Pressure 1 MSB	Annulus Pressure 0-10000 PSI (MSB)
Pressure 1 LSB	Annulus Pressure 0-10000 PSI (LSB)
Pressure 2 MSB	Production Pipe Pressure 0-10000 PSI (MSB)
Pressure 2 LSB	Production Pipe Pressure 0-10000 PSI (MSB)
RTD Reading MSB (Pressure sensor 2)	Production Pipe Sensor RTD Value MSB 0-60000
RTD Reading LSB (Pressure sensor 2)	Production Pipe Sensor RTD Value LSB 0-60000
PCB Temperature (1 byte)	DHCS PCB Temperature -40 to 150, transmitted with an offset of 40 as 0 to 190 (°C)
Checksum (1 byte)	

The SBC will hold the most recent status data received from the SCS for transmission on request from the MODBUS interface. This will not reflect recent DHCS data unless there has been a recent command sent to the DHCS.

Notes:

The RTD Readings are of temperature measured at the pressure sensor.

The SCS status message is sent in response to an SBC Get Data command when in remote mode and is sent unsolicited upon any change to the stored status when in local mode.

In local mode, for each active mandrel, on expiry of 5 minutes without receipt of status from the DHCS and at each subsequent period of 5 minutes without status (or at regular intervals), the SCS PIC will request DHCS status such that the status held by the SCS is never more than 5 minutes stale. The control room PC software is expected to do likewise in remote mode.

3.1.4 Checksum Calculation and Error Handling

3.1.4.1 SBC Commands to SCS

The checksum shall be calculated by adding the mandrel number and the two command bytes together into a single 8 bit byte, ignoring overflow.

The SCS on receipt of a command with a valid format and checksum, will respond with the SCS Status message, either immediately or for a command involving the DHCS, within 2 seconds on receipt of the DHCS response. Otherwise, including the case of no response from the DHCS, the SCS will not respond to the SBC.

If the SBC receives no valid response to a command within a parameter timeout of 3 seconds (for an open or close actuator, or a get status command) or a parameter time of 1 second (for any other command), the failure will be written as an SCS communications failure to the SBC log file and indicated as a DHCS No Response Communications failure in responses to status requests from the MODBUS so as to indicate that the status data sent is stale. The log entry will indicate whether CTS became false or not and whether it had become true before expiry of the timeout.

The SBC will not attempt any recovery of failed communications with the SCS, this being initiated by the CRS. Recovery failure would require the user to re-boot the SCS box and on continued failure to call for service.

Note that the SCS will reset its communications processor if it believes that there is a problem with its communications with the SBC.

3.1.4.2 SCS Messages to SBC

When sending status data to the SBC the checksum shall be calculated by adding all the data bytes together (as single bytes), in the same manner as above.

If the SBC detects a checksum error or the message format is invalid, the SBC will disregard the data other than to log its receipt and the error type.

3.1.5 Communications Protocol

Important Note: The SCS board when communicating with the Down Hole Control System (DHCS) must not be interrupted by the SBC. This is achieved by the SBC only ever initiating a command when CTS is true.

In local mode, the SCS board sends status messages to the SBC whenever received from the DHCS but the SBC sends no messages to the SCS. On entering local mode, the SCS will set CTS to false and then send an unsolicited status message to the SBC, indicating the new switch position. On leaving local mode, the SCS will simply set CTS to true.

In remote mode, a MODBUS Fn5 command will cause the SBC to send the appropriate command to the SCS. The SCS responds by sending back a Status message to the SBC after a delay of approximately 2 seconds for DHCS communications. If a further MODBUS Fn5 message is received for a mandrel that has not responded with a status message since a previous such message and the parameter 3-second timeout has not expired, the SBC will ignore the MODBUS message.

If the SBC detects a false CTS line that persists for more than a 3 seconds parameter without receiving a status message indicating local mode, the SBC will indicate a DHCS No Response communications failure in response to any Fn3 status request command received from the MODBUS.

3.2 SBC CONFIGURATION INTERFACE

This is an RS232 based port that allows set up of the SBC. The configuration is an RS232 interface that accepts simple commands to set up the various Modbus parameters, and (in future) the Ethernet (LAN) parameters. It is designed to be run from standard terminal programs (e.g. Hyper Terminal)

The port is COM4 of the CPU which is normally reserved for a third SCS board interface, hence the SBC will examine messages received on this interface to determine from their format whether the port is currently in use as a configuration port or as an SCS port. [Note: alternatively, the second USB port specified at section 2 could be used for one of these interfaces vial a USB/RS232 converter]

The interface settings are 9,600 bits per second, 1 start bit, 8-bit data, 1 stop bit, no parity. Bytes are sent in the order described below and multi-byte words are sent most significant byte first. No ASCII ACKs or NAKs are sent in either direction.

There is no checksum error checking. Commands sent will be echoed by the SBC with a Pass/Fail if the parameter was accepted and programmed. Errors will be corrected manually by the user.

Modbus Programmable parameters are given below.

- Communications: Bit rate, parity setting and stop bits
- Device Identification: Vendor name, product code and major minor revision
- Slave Address: This can be set separately for each mandrel within the range 1-247.

Ethernet Parameters are TBD (future implementation)

Further commands interrogate the SBC for the DHCS capacitor bank values and memory stick log data.

3.2.1 Communications Protocol

The protocol is based on simple string instructions, with parameters separated by commas.

The following commands have been defined.

Modbus Bit Rate

MBBR,n	Where n is a standard bit rate.
e.g. MBBR,9600	Sets The Modbus bit rate to 9,600 (decimal), the default value.

Modbus Parity

MBPY,n	Where n = 0 for no parity, 1 for odd parity or 2 for even parity.
e.g. MBPY,0	Sets The Modbus parity to off, the default value.

Modbus Stop Bits

MBSB,n Where n is the number of stop bits 1 - 2.

e.g. MBSB,1 Sets The Modbus stop bits to 1, the default value.

Vendor Name (System)

MVEN,a Where a is free text (1 - 50 alphanumeric and space characters only).

e.g. MBSA,Camcon Technology Ltd.

Product Code (Mandrel)

MPRO,n,a Where n is the mandrel number (1 - 9) and a is free text (1 - 50 alphanumeric and space characters only).

e.g. MBSA,2,DL100000

MajorMinorRevision (Mandrel)

MREV,n,a Where n is the mandrel number (1 - 9) and a is free text (1 - 20 alphanumeric and space characters only).

e.g. MBSA,2,B-2

[Wingpath: please check whether these field lengths will all fit into one MODBUS response message]

Mandrel Slave Address

MBSA,n,m Where n is the mandrel number (1 - 9) and m is the slave address 1-247 which will be rejected if not unique within this system

e.g. MBSA,2,20 Sets Mandrel 2 slave address to 20 (decimal)

Delete Mandrel

MBSA,n,0 Where n is the mandrel number (1 - 9)

e.g. MBSA,2,0 Removes mandrel 2 from the system configuration and purges all capacitor health data for the mandrel other than log data, such that it cannot be used again. This command will fail unless the selected mandrel is the highest numbered mandrel connected to the SCS. [Note: If an intermediate mandrel is to be deleted, the dealer will need to delete the highest numbered mandrel and then change the configuration details of the intermediate mandrel to those of the deleted one].

Actuator Pulse Time

MAPT,n Where n is an index to a pulse time (0 - 3).

e.g. MAPT,2

Capacitance

MCAP,n Where n is an actuator number (1 - 9).

e.g. MCAP,2

Any other commands have yet to be defined, but will follow a similar pattern.

Once the command sequence has been accepted (or fails) the SBC will reply with the given command, and a Pass/Fail. Therefore a complete command sequence will look similar to the example below where ">" is the cursor of the controlling PC application window and is not transmitted.

>MBSA,2,20	(sent to the SBC)
MBSA,2,20 OK	(reply from SBC)
>	(waiting for next command)

For the capacitance and resistance commands, the pass response will be the given command and the value as in the example below. The fail response will be as for other commands.

>MCAP,2	(sent to the SBC)
MCAP,2,500	(reply from SBC)
>	(waiting for next command)

Commands and responses for memory stick log access (see section 3.5) are to be proposed by Wingpath

3.3 SERIAL MODBUS INTERFACE

This is a RS485 based Modbus Protocol interface.

Please refer to the following documents for the full requirements

- MODBUS Application Protocol Specification V1.1b.
- MODBUS over Serial Line Specification and Implementation Guide V1.02
- Camcon Document DI100001 Modbus Protocol.

These documents fully describe the requirements for this interface.

3.4 ETHERNET MODBUS INTERFACE (FUTURE REQUIREMENT)

This is a 10/100 Base-T MODBUS TCP based interface. It shall conform to the MODBUS TCP/IP Specification.

Refer to the documents

- MODBUS Application Protocol Specification V1.1b.
- MODBUS Messaging On TCP/IP Implementation Guide V1.0b
- Camcon Document DL100001 Modbus Protocol.

It shall perform the same function as the Serial MODBUS Interface, and from the user point of view should be indistinguishable.

3.5 DATA COLLECTION INTERFACE

All data transfers are to be logged to a USB Memory stick. This will conform to the USB 2.0 specification.

All traffic and faults shall be logged, this will include,

- MODBUS commands (both Serial and Ethernet based traffic)
- Data sent to the SCS board
- Data received from the SCS Board
- SBC Configuration interface commands and responses
- Communications errors

In addition to the actual data, the source of the data shall also be noted (MODBUS Serial, MODBUS Ethernet, SCS status etc), and all data transfers shall be individually time and date stamped.

The file format shall be plain ASCII text with binary data converted to hexadecimal notation. It shall be readable by any text file reader and shall use commas or tab characters exclusively to define fields such as date, time, source, event etc. However, the file shall be encrypted prior to saving and shall be given the maximum protection against deletion that is available within the operating system. .

A mechanism shall be provided for the file to be retrieved and decrypted on command (TBD) via the SBC Configuration interface with search controls appropriate to the envisaged file size. No mechanism shall be provided to delete or amend the file on the memory stick other than to add sequential log data to the end of the file as specified herein.

It is not expected that the memory stick would ever become full (as a 32GB stick would last 87 years, if each data block was 1000 bytes long and there were 1000 transfers a day).

3.6 DHCS DATA TRANSLATION

Most of the data received from the DHCS is in a basic form, and needs to be translated for the Modbus interface. This section deals with the required conversions.

3.6.1 Pressure Measurement

The data from the pressure sensors is in raw ADC counts. This must be converted in to PSI by using the following conversion.

First convert the ADC counts into mV

Pressure Reading (mV) = ADC Counts * 78.125 / 65535

Then convert in to PSI

Pressure Reading PSI = (Reading mV * 10,000) / 13.0

3.6.2 RTD Temperature Measurement

The data from the Pressure sensor RTD is in raw ADC Counts. This must be converted in to Degrees C. This calculation is first to convert ADC counts in to resistance, by the formula below.

RTD Resistance (Ohms) = ADC Counts * 2000 / 65535

From the resistance the temperature can be determined by solving for the following equation,

RT = 1000 * (1+ (0.00381 * T) – (0.0000005775 * T^2))

or by using a look up table with a 0.1°C step size (accuracy to 1°C is required) based on the above equation.

3.7 ACTUATOR TEST SCHEME

A test scheme needs to be implemented so that each actuator is moved at least once per week and so that the capacitor bank health is checked once per week. The action of moving the actuator helps to prevent it becoming stuck in one position and also detects any electrical failure. The scheme is described in the following document which also specifies the associated calculations and diagnoses:

- DL100005 Heath Monitoring Report, Section 4.0.

When the SCS is in local mode, the scheme will be the responsibility of the user and the capacitor bank health check will not be performed. Otherwise, the scheme will be run from the MODBUS, under manual control by the control room user or in future via a script option.

Whether in local or remote mode, the SBC will check each status message received for an actuator that has not moved when commanded or has moved when not commanded.

If an additional actuator moves as well as the commanded one, this is likely to be due to a short circuit to actuator casing and is left to the user to diagnose (In local mode, an actuator LED will start to flash at the same time as a commanded change to another actuator occurs; in remote mode, the CRS can display a message that suggests the possibility).

If an actuator moves in the absence of any contemporaneous command, it is likely to be due to pressure imbalance rather than a fault.

The SBC will perform the diagnosis of any actuator failing to move when commanded or being commanded to move to its current position (as reported to it in an SCS status message), in readiness to respond to a MODBUS Fn3 command. Note that a command to move to its current position is not necessarily an indication of a fault condition.

The user can command a faulty actuator to move to its current position so that the capacitor drop calculation can be performed more accurately. The SBC will detect this command to a faulty actuator and perform the calculation to make the diagnosis available to the MODBUS. Note that the parameters involved in the checks may differ from those necessary when checking against an actuator movement.

The user will need to be aware that any communications error since the previous command to the actuator may invalidate the diagnosis.

Data will be logged to memory stick and reported over the MODBUS to the control PC in the normal manner.

3.8 CAPACITOR BANK HEALTH

Refer to the following document for ~~all~~ information regarding the method and calculations required to determine the Capacitor lifetime left:

- DL100005 Health Monitoring Report, Section 2.0.

Note: The life time left will start at 100%, and fall over time to 0% at which point its effective capacity has halved and further use will become unreliable. Once the lifetime has fallen, the displayed value will not be increased, even if the received values indicate this (i.e. the lifetime left can never increase, only fall).

The capacitor health check is only ever performed whilst the SCS is in remote mode and is CRS user-initiated using non-faulty actuators which are commanded to their current position. The SBC detects a command to a non-faulty actuator and performs a capacitor health check from the reported status.

If a received measurement implies a change not exceeding a parameter 2% it will become part of a rolling average of the last five valid measurements for the same actuator. Ignoring any

faulty actuators, an average will be taken of these rolling averages for each actuator and will be the current capacitor health value for the mandrel. The value reported via the MODBUS will be this value or the last reported value, whichever is the lower.

If a measurement is diagnosed as due to an actuator fault as described in section 3.7, the measurement shall be ignored for the purposes of capacitor bank health and the rolling average for that actuator will be eliminated from the overall average.

If the change exceeds 2%, the value reported via the MODBUS will be the out-of-range value 0xFF which will indicate to the CRS that the SBC will check the actuator by toggling and then unless faulty, issue further commands for the same actuator until either a change not exceeding 2% is detected or three consecutive greater changes are measured which are within 2% of each other. In the former case, the last reading will be accepted into the rolling average and previous greater change readings will be discarded. In the latter case, the three readings will be accepted and used to start a new rolling average, building up over time to an average of five. In both cases, the normal rolling average value will be reported via the MODBUS so that the CRS will know that the tests for the actuator are complete. A faulty actuator will be eliminated from the overall average as specified above.

3.9 STARTUP INITIALISATION

On SCS box startup, the SBC will wait for CTS from the SCS to become true and then send a Mandrel Configuration command to the SCS, followed for each mandrel by a Set Actuator Pulse Time command and a Get Status command, waiting for the status response or timeout between each command. Until a status message response has been received from a mandrel, the SBC will ignore any commands for that mandrel from the MODBUS, including status requests.

On SCS box startup, the most recent capacitor rolling averages shall be retrieved from static memory, whether the memory stick (by analysing the log or by including each new calculation as a log entry) or otherwise (eg flash).

Note that this action will not provide up to date status on capacitor health nor actuator faults. In local and remote modes, the user can check actuator health manually. In remote mode, the user can also check capacitor health; in a future implementation, this and actuator health could be checked via a CRS script.

If no mandrels have been configured within the SBC, eg at installation, or on any mandrel being added via the SBC Configuration interface, then the SBC, as each mandrel is added, will send a Mandrel Configuration command to the SCS followed by a Set Actuator Pulse Time command for the mandrel and a Get Status command, waiting for the status response between each command. It is the responsibility of the installer to request a capacitor health check for each new mandrel. The SBC will, on detecting the request and the absence of initial values, obtain

initial mandrel capacitance values as specified in DL100005. Capacitor health will be set to 100%, the initial value will be used as the first for the rolling average for that actuator and the initial temperature will be stored. Once the SBC has an initial capacitor health value for an actuator, any further value will continue to build up the rolling average for that actuator which will be available for reporting as part of the overall average to the MODBUS. As other actuators are addressed, the same process will be followed except that the initial temperature will remain unaltered and the initial value for each actuator will be adjusted for any temperature change since the first.

If a mandrel is removed permanently from the system, it must be deleted via an SBC Configuration interface command before a new mandrel is fitted. This action shall be recorded to the log such as to cause a re-initiation for any new mandrel fitted and shall purge any initial and rolling capacitor data for that mandrel from flash. Deletion of a mandrel will cause the SCS to purge pulse time and status data for that mandrel.

If initial data does not exist for an actuator, detection of failure to move will indicate the current position in the response to a Fn3 command but as capacitor calculations cannot be performed, there will be no faults indicated in the actuator status message.