

# **Bilaga B**

## **Funktionspezifikation**

## Table of contents

<b>1</b>	<b>INTRODUCTION .....</b>	<b>3</b>
1.1	SUMMARY .....	3
1.2	REFERENCES .....	3
<b>2</b>	<b>COMPLIANCE WITH SPECIFICATION .....</b>	<b>4</b>
2.1	POSSIBLE FUTURE EXTENSIONS .....	4
<b>3</b>	<b>SPECIFICATION .....</b>	<b>4</b>
3.1	OPERATION DESCRIPTION .....	4
3.1.1	<i>Limitations in application</i> .....	5
3.2	C SUBFUNCTIONS .....	5
3.2.1	<i>lua_functions</i> .....	5
3.2.1.1	lua_functions_write_pdbuff .....	5
3.2.1.2	lua_functions_read_pdbuff .....	6
3.2.1.3	lua_functions_light_diode .....	6
3.2.1.4	lua_functions_tp1_high .....	6
3.2.1.5	lua_functions_tp1_low .....	6
3.2.2	<i>lua_port</i> .....	6
3.2.2.1	LUA_Init() .....	7
3.2.2.2	lua_Task() .....	7
3.2.2.3	lua_port_openlualibs() .....	7
3.3	LUA SUBFUNCTIONS .....	7
3.3.1	<i>modbus.lua</i> .....	7
3.3.1.1	modbusInit .....	7
3.3.1.2	modbusReadMultiple .....	8
3.3.1.3	modbusWriteMultiple .....	8

Utfärdare / Issued by <b>Jesper Håkansson &amp; Jimmie Hansson</b>	Datum / Date <b>2011-12-06</b>	Beteckning / Ref. <b>Error! Unknown</b>	Sida / Page <b>3 (8)</b>
Ärende / Subject <b>Functional specification</b>		Dokumenttyp / Document type	
		Godkänd av / Approved by	

# 1 Introduction

## 1.1 Summary

The system creates a Lua state and execute a Lua file from the built in FTP-server. The Lua file “modbus.lua” connects to a Modbus-slave and sends function 3 (read multiple registers) and 16 (write multiple registers) with 32 byte data in an endless loop. The response from the Modbus slave is written back to the ABCC module that sends it back to the TCP-IP client.

## 1.2 References

[www.lua.org](http://www.lua.org)

Utfärdare / Issued by <b>Jesper Håkansson &amp; Jimmie Hansson</b>	Datum / Date <b>2011-12-06</b>	Beteckning / Ref. <b>Error! Unknown</b>	Sida / Page <b>4 (8)</b>
Ärende / Subject <b>Functional specification</b>			Dokumenttyp / Document type
			Godkänd av / Approved by

## **2 Compliance with specification**

### **2.1 Possible future extensions**

To implement all the other Modbus commands.

## **3 Specification**

### **3.1 Operation description**

Utfärdare / Issued by <b>Jesper Håkansson &amp; Jimmie Hansson</b>	Datum / Date <b>2011-12-06</b>	Beteckning / Ref. <b>Error! Unknown</b>	Sida / Page <b>5 (8)</b>
Ärende / Subject <b>Functional specification</b>		Dokumenttyp / Document type	
		Godkänd av / Approved by	

The purpose of the X-gateway is to make it possible for devices on Modbus network and TCP-IP networks to be able to communicate and understand each other.

When the system starts a Lua task is created and this task makes it possible for the X-gateway to read and understand Lua scripts. The Lua scripts are stored on the built in FTP-server and the Lua task will load and execute the Lua file named "modbus.lua" that will give instructions to the system what to do next.

The Lua script will tell the system to connect to a Modbus slave and send the Modbus command "read multiple registers". The Modbus slave will respond to the call and return 32 values of the Modbus slaves register. The program then writes the received values to the ABCC write buffer that periodically sends the write buffer to any TCP-IP client connected to the module.

After that the program reads 32 values from the ABCC read buffer and sends that data together with the Modbus command "write multiple registers" to the Modbus slave. The program then checks for a response message from the Modbus slave to check if there were any errors executing the command. If an error was returned the program will print "write error". If the response message was ok the system will continue calling the two Modbus commands in an endless loop.

### 3.1.1 Limitations in application

Only two Modbus commands are implemented and they only read and write 32 bytes of data. If the same code would be written in C it would run faster because script languages always run slower.

## 3.2 C Subfunctions

### 3.2.1 lua\_functions

#### 3.2.1.1 lua\_functions\_write\_pdbuff

Writes 32 bytes to the ABCC write buffer

Syntax:  
int lua\_functions\_write\_pdbuff( lua\_State \*l )

Input:  
Lua state

---

Utfärdare / Issued by <b>Jesper Håkansson &amp; Jimmie Hansson</b>	Datum / Date <b>2011-12-06</b>	Beteckning / Ref. <b>Error! Unknown</b>	Sida / Page <b>6 (8)</b>
Ärende / Subject <b>Functional specification</b>		Dokumenttyp / Document type	
		Godkänd av / Approved by	

### **3.2.1.2 lua\_functions\_read\_pdbuff**

Reads 32 byte from the ABCC read buffer and pushes it on the Lua stack as a Lua table

Syntax:

```
int lua_functions_read_pdbuff( lua_State *l )
```

Input:

Lua state

---

### **3.2.1.3 lua\_functions\_light\_diode**

Sets the MBTCP\_STATUS\_LED and SD\_STATUS\_LED to blinking red

Syntax:

```
int lua_functions_light_diode( lua_State *l )
```

Input:

Lua state

---

### **3.2.1.4 lua\_functions\_tp1\_high**

Sets test pin 1 to a high state. Can be used to measure how long time certain Lua scripts take to execute

Syntax:

```
int lua_functions_tp1_high( lua_State *l )
```

Input:

Lua state

---

### **3.2.1.5 lua\_functions\_tp1\_low**

Sets test pin 1 to a low state. Can be used to measure how long time certain Lua scripts take to execute

Syntax:

```
int lua_functions_tp1_low( lua_State *l )
```

Input:

Lua state

---

## **3.2.2 lua\_port**

Utfärdare /Issued by <b>Jesper Håkansson &amp; Jimmie Hansson</b>	Datum /Date <b>2011-12-06</b>	Beteckning / Ref. <b>Error! Unknown</b>	Sida / Page <b>7 (8)</b>
Ärende / Subject <b>Functional specification</b>		Dokumenttyp / Document type	
		Godkänd av / Approved by	

### 3.2.2.1 LUA\_Init()

Initiates the Lua task

Syntax:  
 StatusType LUA\_Init( UINT16 iOptions )

Input:  
 iOptions - Initialization flags

---

### 3.2.2.2 lua\_Task()

Lua task main function

Syntax:  
 void lua\_Task( void )

---

### 3.2.2.3 lua\_port\_openlualibs()

A function to open up all the Lua libraries declared in luaL\_reg lualibs[]

Syntax:  
 static void lua\_port\_openlualibs( lua\_State \*l )

Input:  
 Lua state

---

## 3.3 Lua Subfunctions

### 3.3.1 modbus.lua

#### 3.3.1.1 modbusInit

Initiate variables that is used by other functions in Modbus.lua

Syntax:  
 function modbusInit()

---

Utfärdare / Issued by <b>Jesper Håkansson &amp; Jimmie Hansson</b>	Datum / Date <b>2011-12-06</b>	Beteckning / Ref. <b>Error! Unknown</b>	Sida / Page <b>8 (8)</b>
Ärende / Subject <b>Functional specification</b>			Dokumenttyp / Document type
			Godkänd av / Approved by

### **3.3.1.2 modbusReadMultiple**

sends Modbus command 3 (read multiple registers) to Modbus slave

Syntax:

function modbusReadMultiple()

### **3.3.1.3 modbusWriteMultiple**

Gets 32 bytes from ADCC read buffer and sends it together with Modbus command 16 (write multiple registers) to Modbus slave

Syntax:

function modbusWriteMultiple()