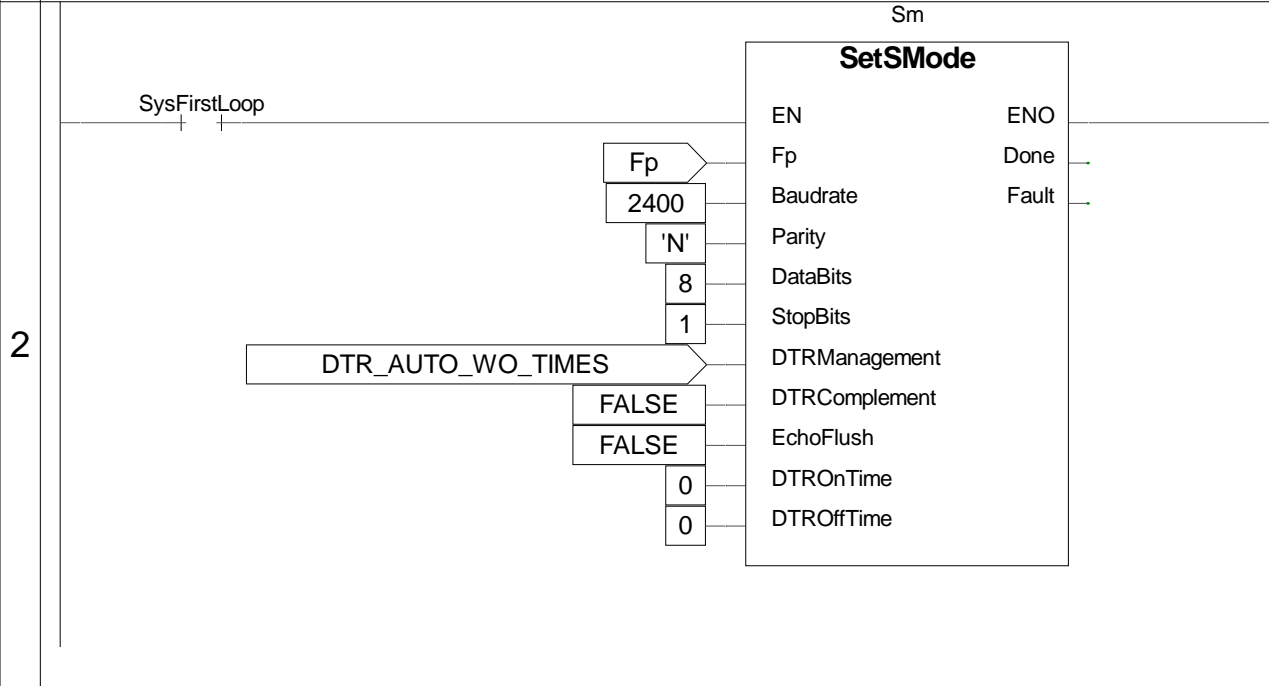
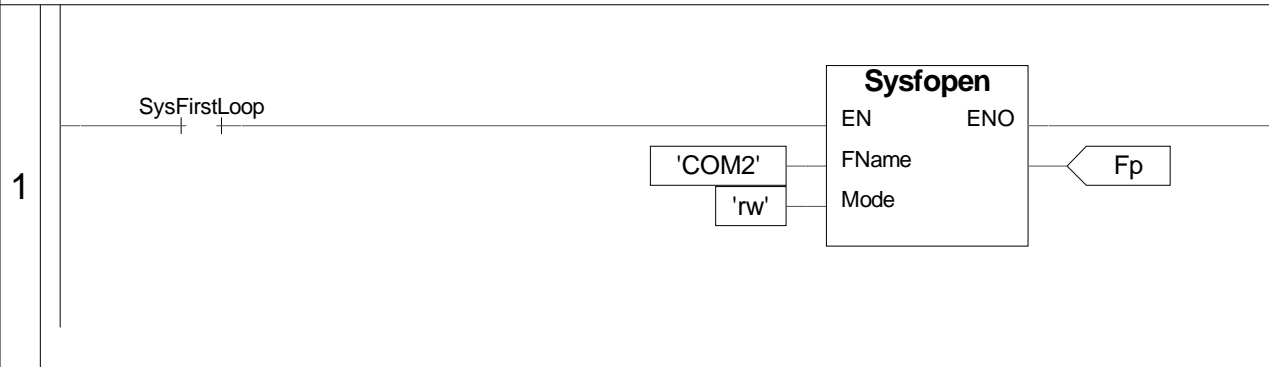
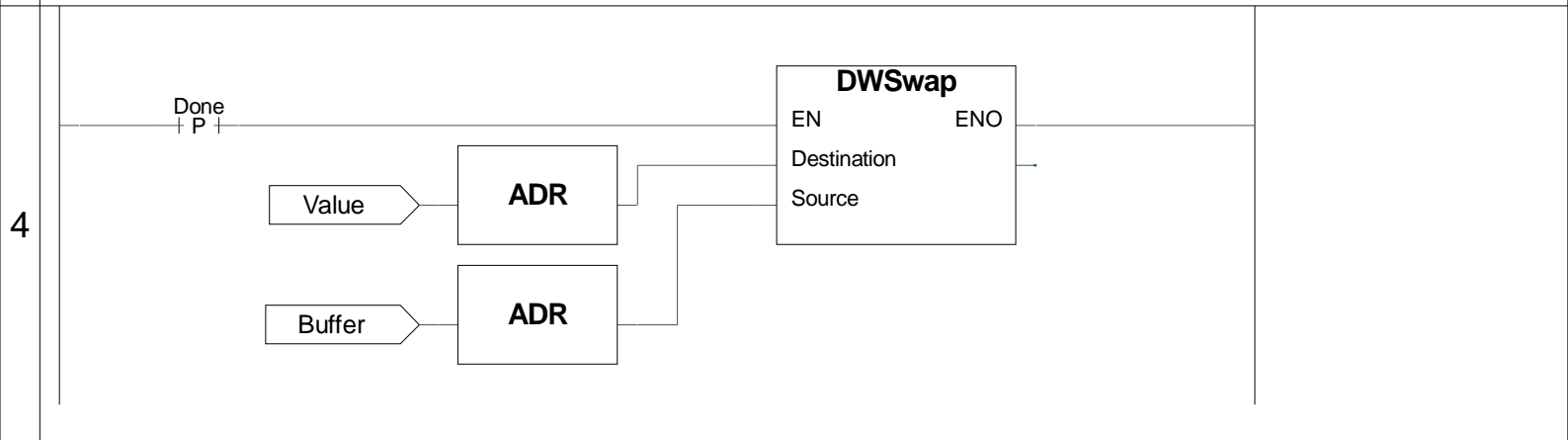
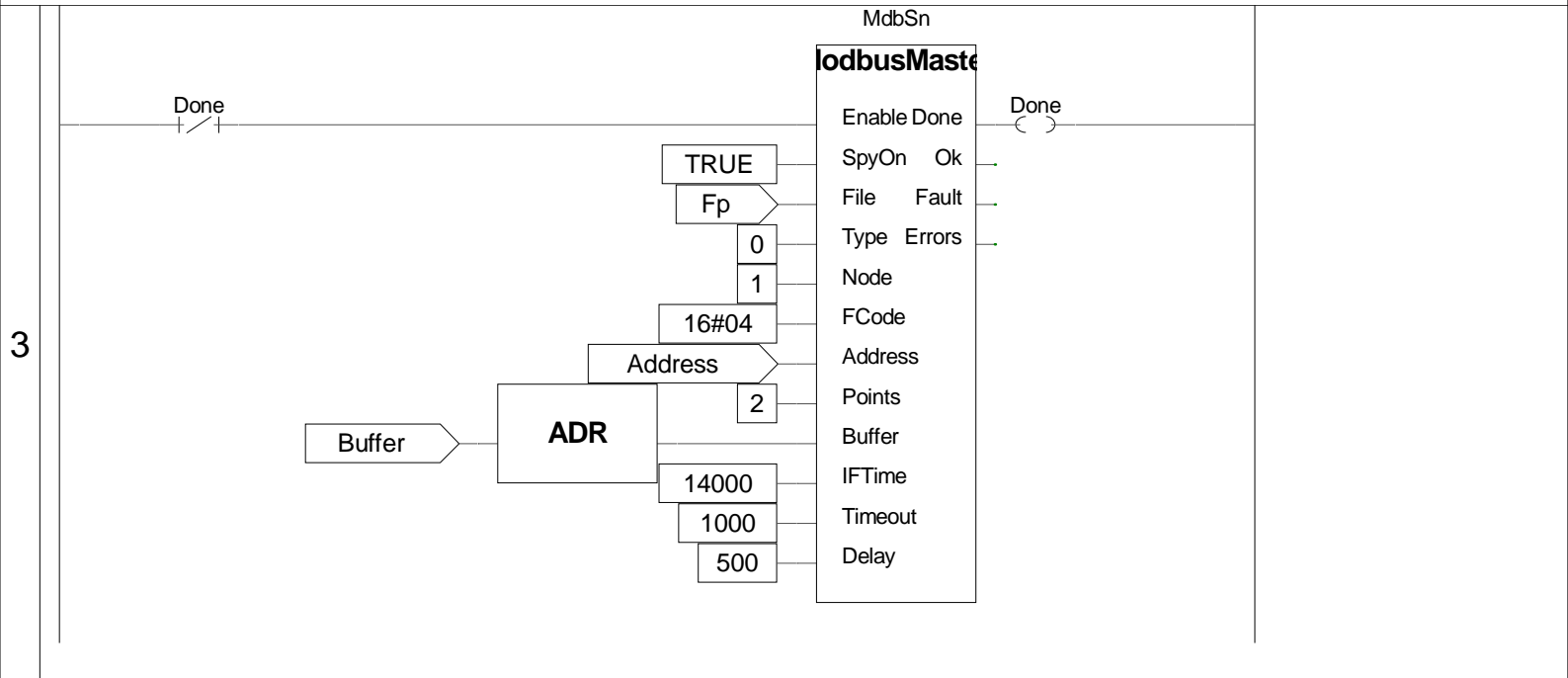


```

VAR
RxCoils : ARRAY[ 0..15 ] OF BOOL; (* Rx coils data buffer *)
Fp : FILEP; (* File pointer *)
MdbSn : ModbusMaster; (* Modbus master FB *)
Sm : SetSMode; (* Set communication mode *)
Done : BOOL; (* Modbus done *)
Address : UINT := 1; (* Register address *)
Value : REAL; (* Register value *)
Buffer : ARRAY[ 0..1 ] OF WORD; (* Read buffer *)
END_VAR
    
```



Project : SDM120SingleRegister	
PROGRAM : SDMInterface	
Release : SDM120	Ver :1.00
Author :	Date:24/06/2015
Note :	Page:1 of 2



Project : SDM120SingleRegister	
PROGRAM : SDMInterface	
Release : SDM120	Ver :1.00
Author :	Date:24/06/2015
Note :	Page:2 of 2

(ePLCUtyLib_C000) Set the serial communication mode
 ENCRYPTED CODE

```

VAR_INPUT
Fp : FILEP;
Baudrate : UDINT; (* Baudrate *)
Parity : STRING[ 1 ]; (* Parity type *)
DataBits : USINT; (* Nr of data bits *)
StopBits : USINT; (* Nr of stop bits *)
DTRManagement : USINT; (* DTR management type *)
DTRComplement : BOOL; (* Complement the DTR signal *)
EchoFlush : BOOL; (* Flush the echo *)
DTROnTime : UINT; (* DTR On wait time *)
DTROffTime : UINT; (* DTR Off wait time *)
END_VAR

VAR_OUTPUT
Done : BOOL; (* Execution done *)
Fault : BOOL; (* Execution fault *)
END_VAR
    
```

1

Project : SDM120SingleRegister	
FUNCTION BLOCK : SetSMoDe	
Release : SDM120	Ver :1.00
Author :	Date:24/06/2015
Note :	Page:1 of 1

FUNCTION_BLOCK ModbusMaster

(ePLCutyLib_C000) Manages the modbus master communication
 ENCRYPTED CODE

```

VAR_INPUT
Enable : BOOL; (* FB enable *)
SpyOn : BOOL; (* Spy active *)
File : FILEP; (* Terminal I/O pointer *)
Type : USINT; (* Modbus type *)
Node : USINT; (* Node number *)
FCode : USINT; (* Function code *)
Address : UINT; (* Start address *)
Points : UDINT; (* Number of points *)
Buffer : @USINT; (* Address of data buffer *)
IFTime : UDINT; (* Interframe time (uS) *)
Timeout : UINT; (* Timeout time (mS) *)
Delay : UINT; (* Delay time (mS) *)
END_VAR
  
```

```

VAR_OUTPUT
Done : BOOL; (* Command done *)
Ok : BOOL := FALSE; (* Execution Ok *)
Fault : BOOL; (* Command fault *)
Errors : UDINT; (* Error counter *)
END_VAR
  
```

1

	Project : SDM120SingleRegister	
	FUNCTION BLOCK : ModbusMaster	
	Release : SDM120	Ver :1.00
	Author :	Date:24/06/2015
	Note :	Page:1 of 1

FUNCTION DWSwap

```
VAR_INPUT
Destination : @USINT; (* Destination pointer *)
Source : @USINT; (* Source pointer *)
END_VAR
```

```
VAR
Buffer : ARRAY[ 0..3 ] OF BYTE; (* Swap buffer *)
END_VAR
```

```
1 (* ***** *)
2 (* FUNCTION "BOOL DWSwap(@USINT Source)" *)
3 (* ***** *)
4 (* Questa funzione esegue lo swap di una variabile DW. *)
5 (* ----- *)
6 (* Eseguo swap variabile. *)
7
8     Buffer[2]:=@Source; (* Swap buffer *)
9     Source:=Source+1; (* Source pointer *)
10
11     Buffer[3]:=@Source; (* Swap buffer *)
12     Source:=Source+1; (* Source pointer *)
13
14     Buffer[0]:=@Source; (* Swap buffer *)
15     Source:=Source+1; (* Source pointer *)
16
17     Buffer[1]:=@Source; (* Swap buffer *)
18
19     @Destination:=Buffer[0];
20     Destination:=Destination+1; (* Source pointer *)
21
22     @Destination:=Buffer[1];
23     Destination:=Destination+1; (* Source pointer *)
24
25     @Destination:=Buffer[2];
26     Destination:=Destination+1; (* Source pointer *)
27
28     @Destination:=Buffer[3];
29     DWSwap:=TRUE; (* Function result *)
30
31 (* [End of file] *)
32
33
```

Project : SDM120SingleRegister	
FUNCTION : DWSwap	
Release : SDM120	Ver :1.00
Author :	Date:24/06/2015
Note :	Page:1 of 1