

VARIABLES

VAR_GLOBAL

```

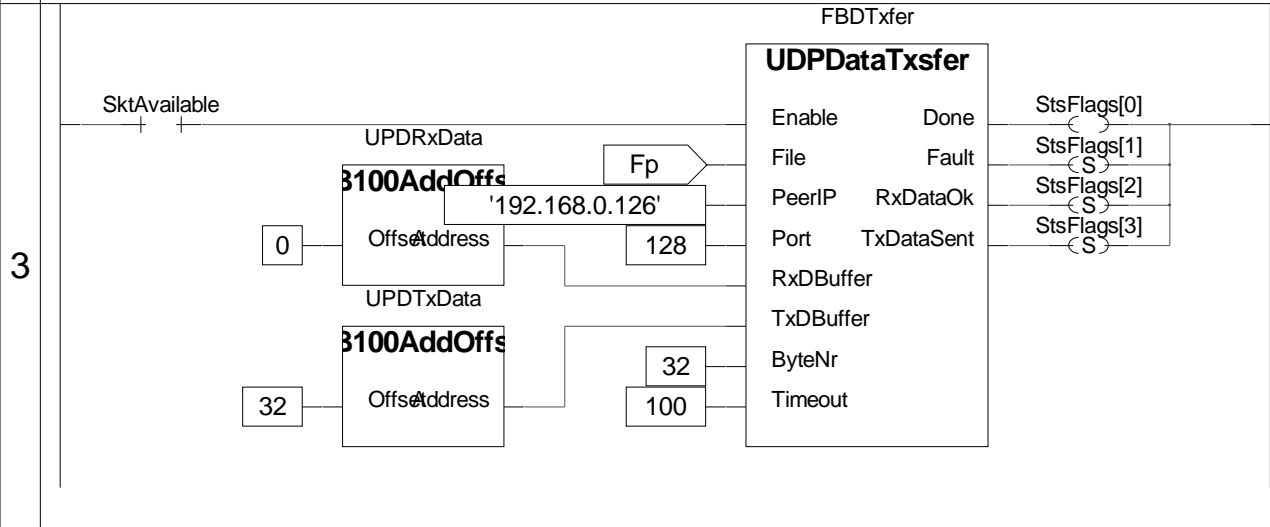
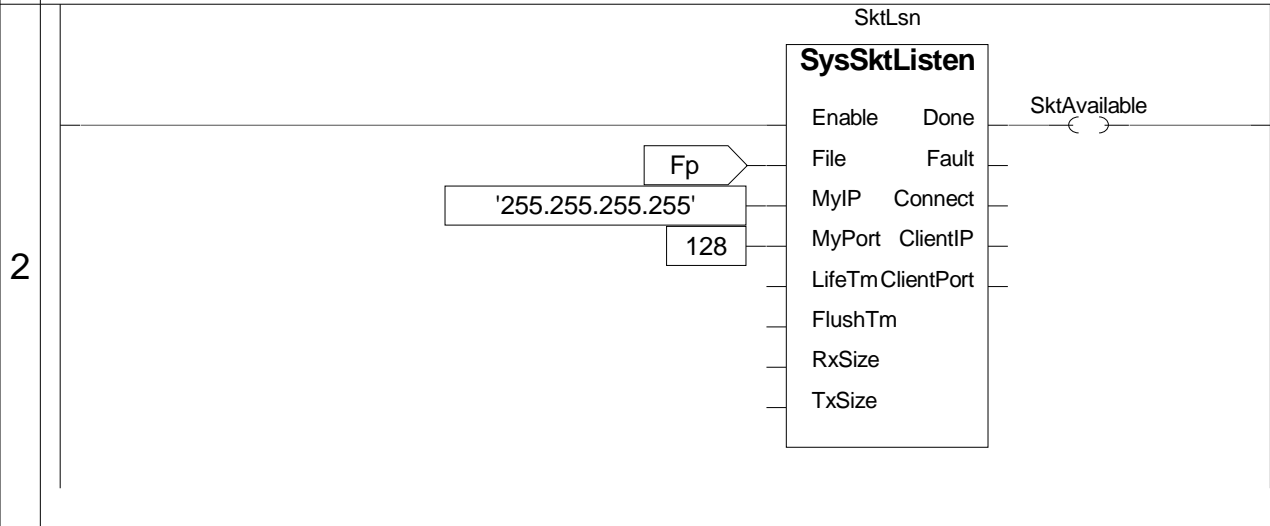
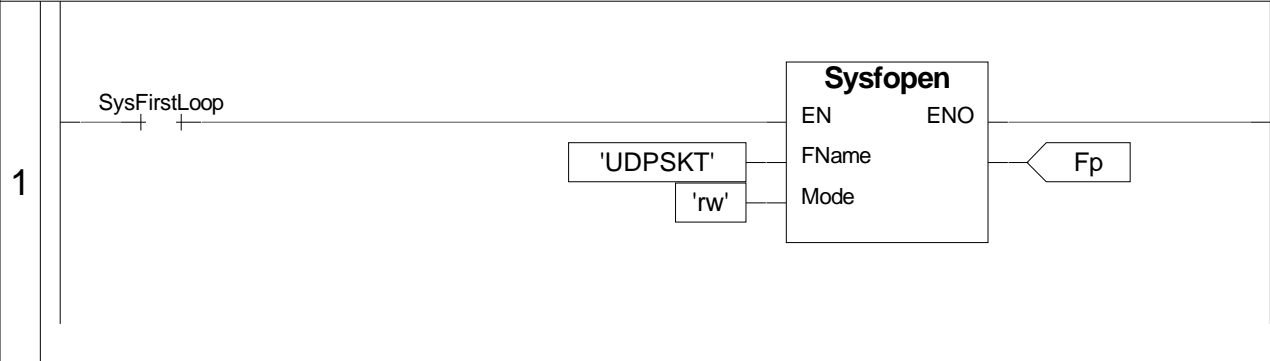
Di00M00 AT %IX0.0 : BOOL; (* Input 00, Module address 0 *)
Di01M00 AT %IX0.1 : BOOL; (* Input 01, Module address 0 *)
Di02M00 AT %IX0.2 : BOOL; (* Input 02, Module address 0 *)
Di03M00 AT %IX0.3 : BOOL; (* Input 03, Module address 0 *)
Di04M00 AT %IX0.4 : BOOL; (* Input 04, Module address 0 *)
Di05M00 AT %IX0.5 : BOOL; (* Input 05, Module address 0 *)
Di06M00 AT %IX0.6 : BOOL; (* Input 06, Module address 0 *)
Di07M00 AT %IX0.7 : BOOL; (* Input 07, Module address 0 *)
Di08M00 AT %IX0.8 : BOOL; (* Input 08, Module address 0 *)
Di09M00 AT %IX0.9 : BOOL; (* Input 09, Module address 0 *)
Di10M00 AT %IX0.10 : BOOL; (* Input 10, Module address 0 *)
Di11M00 AT %IX0.11 : BOOL; (* Input 11, Module address 0 *)
Do00M00 AT %QX0.0 : BOOL; (* Output 00, Module address 0 *)
Do01M00 AT %QX0.1 : BOOL; (* Output 01, Module address 0 *)
Do02M00 AT %QX0.2 : BOOL; (* Output 02, Module address 0 *)
Do03M00 AT %QX0.3 : BOOL; (* Output 03, Module address 0 *)
Do04M00 AT %QX0.4 : BOOL; (* Output 04, Module address 0 *)
Do05M00 AT %QX0.5 : BOOL; (* Output 05, Module address 0 *)
Do06M00 AT %QX0.6 : BOOL; (* Output 06, Module address 0 *)
Do07M00 AT %QX0.7 : BOOL; (* Output 07, Module address 0 *)
DiToRemote AT %MW100.32 : WORD; (* Digital input to remote *)
DoFromRemote AT %MW100.0 : WORD; (* Digital output from remote *)
AiToRemote AT %MD100.36 : ARRAY[ 0..3 ] OF REAL; (* Analog input to remote *)
AoFromRemote AT %MD100.4 : ARRAY[ 0..3 ] OF REAL; (* Analog output from remote *)
END_VAR

```

	Project : ASystem	
	VARIABLES :	
	Release :	Ver :1.00
	Author :	Date:04/02/2011
	Note :	Page:1 of 1

```

VAR
SktAvailable : BOOL; (* Socket available *)
UPDRxData : DB100AddOffset; (* Address of Rx data *)
UPDTxData : DB100AddOffset; (* Address of Tx data *)
Fp : FILEP; (* UDP socket *)
FBDTxfer : UDPDataTxfer; (* UDP data transfer *)
SktLsn : SysSktListen; (* Socket listen *)
StsFlags : ARRAY[ 0..3 ] OF BOOL; (* Status flags *)
CPUiOMng : CPUModuleIO; (* CPU I/O management *)
Dummy : BOOL; (* Dummy variable *)
END_VAR
    
```



Project : ASystem

PROGRAM : DataTransfer

Release : Ver :1.00

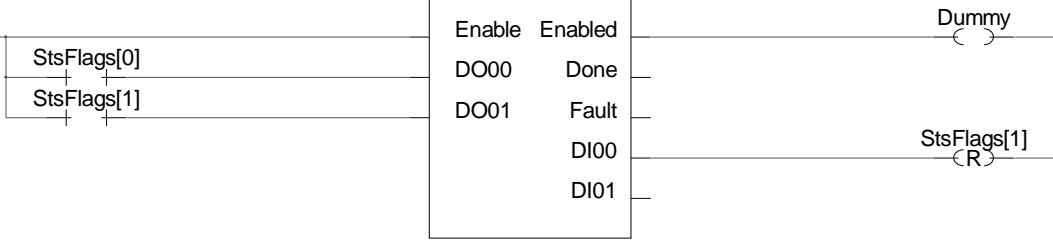
Author : Date:04/02/2011

Note : Page:1 of 2

4

CPUiOMng

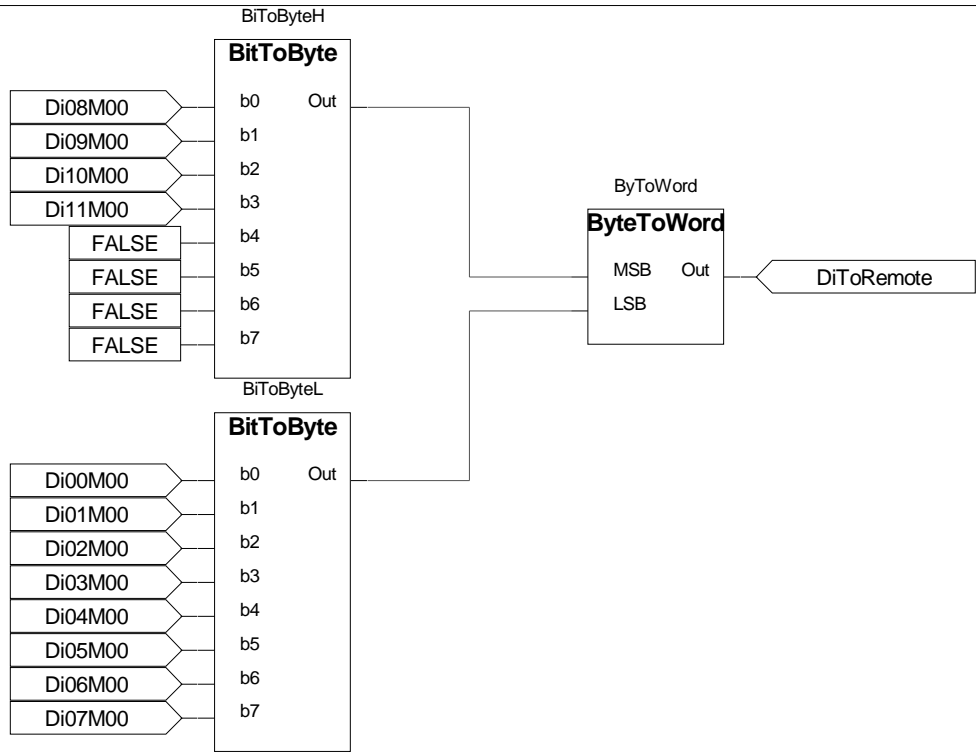
CPUModuleIO



	Project : ASystem	
	PROGRAM : DataTransfer	
	Release :	Ver :1.00
	Author :	Date:04/02/2011
	Note :	Page:2 of 2

```

VAR
BiToByteH : BitToByte; (* Bits to byte *)
BiToByteL : BitToByte; (* Bits to byte *)
ByToWorld : ByteToWorld; (* Byte to word *)
END_VAR
    
```



Project : ASystem

PROGRAM : LogicInput

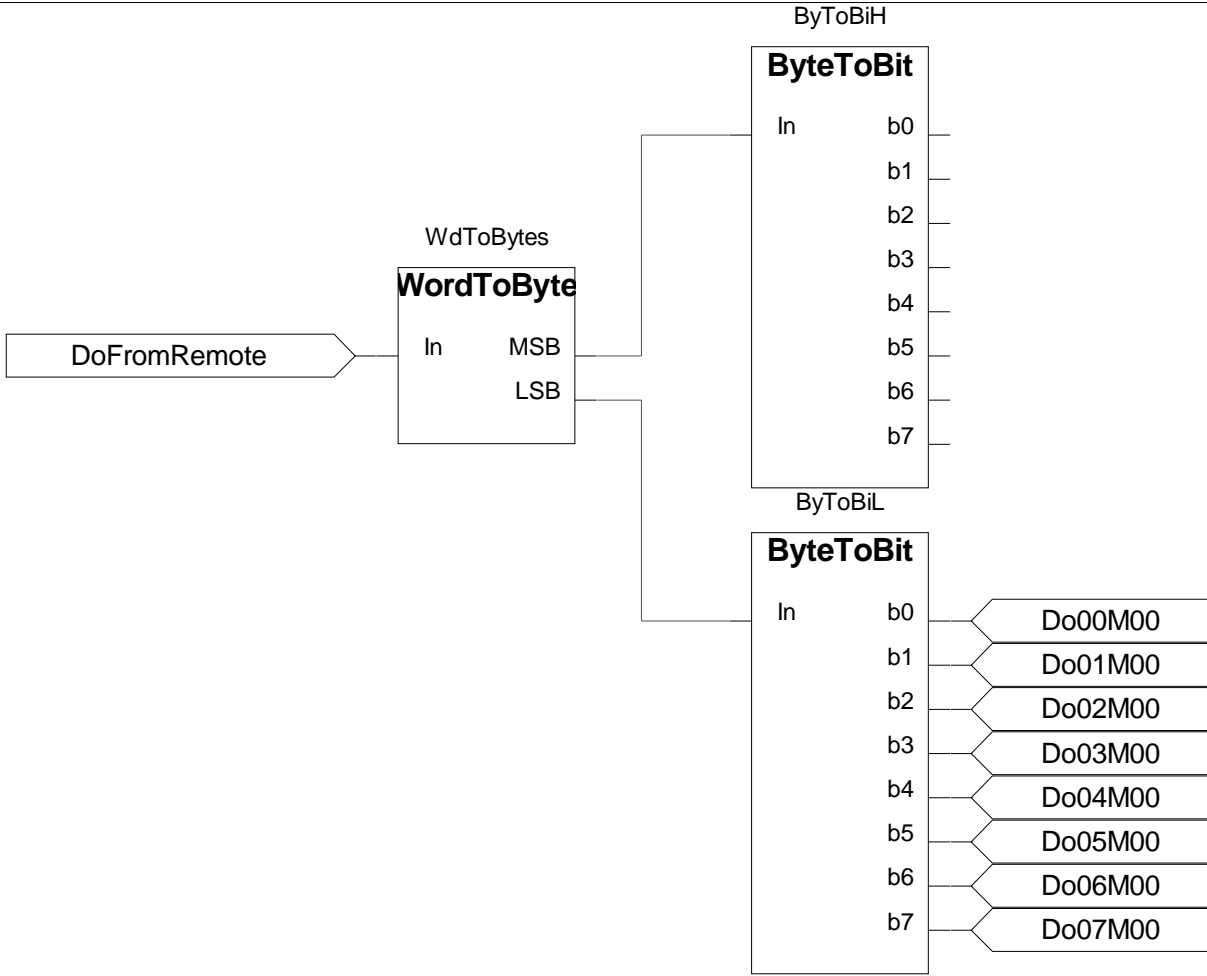
Release : Ver :1.00

Author : Date:04/02/2011

Note : Page:1 of 1

```

VAR
ByToBiH : ByteToBit; (* Byte to bits *)
ByToBiL : ByteToBit; (* Byte to bits *)
WdToBytes : WordToByte; (* Word to bytes *)
END_VAR
    
```



Project : ASystem

PROGRAM : LogicOutput

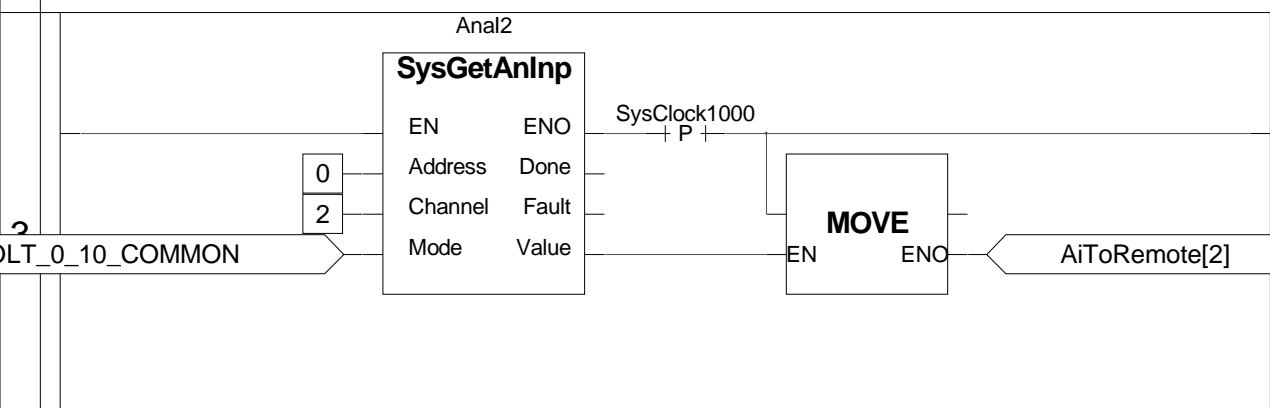
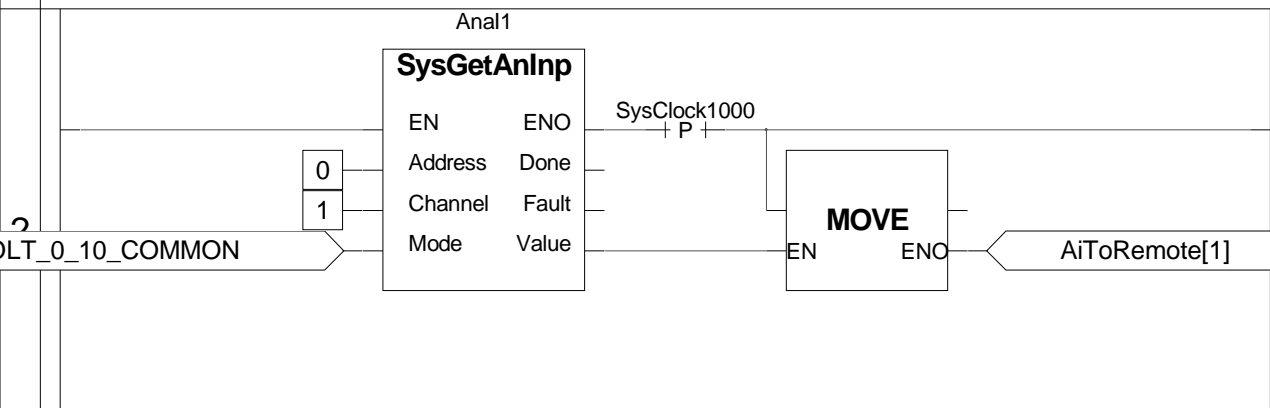
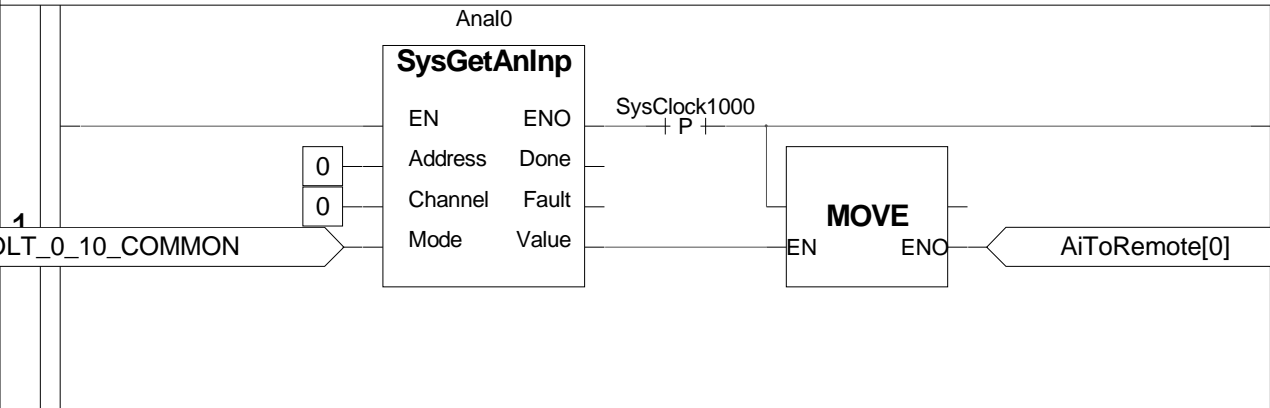
Release : Ver :1.00

Author : Date:04/02/2011

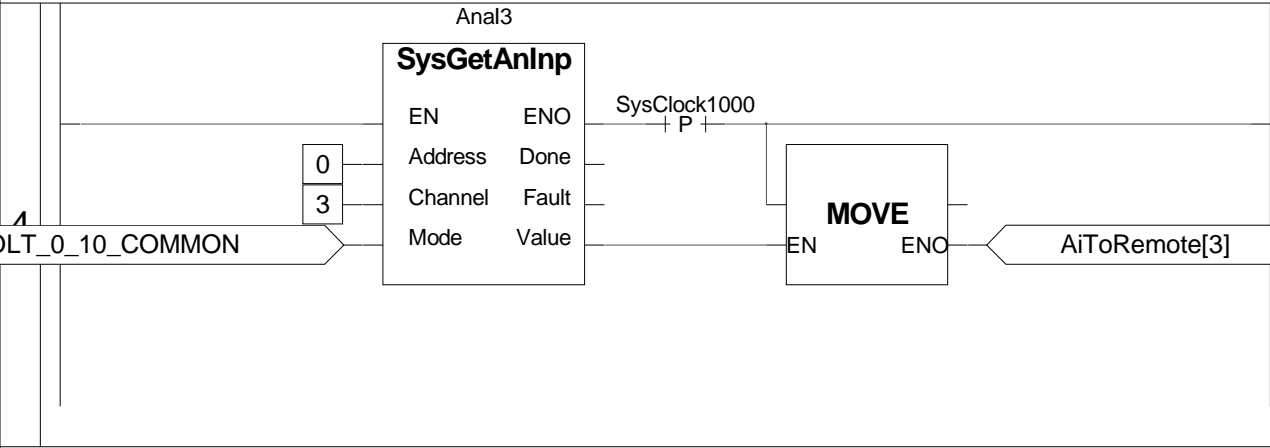
Note : Page:1 of 1

```

VAR
AnaI0 : SysGetAnInp; (* Analog input acquisition *)
AnaI1 : SysGetAnInp; (* Analog input acquisition *)
AnaI2 : SysGetAnInp; (* Analog input acquisition *)
AnaI3 : SysGetAnInp; (* Analog input acquisition *)
END_VAR
    
```



Project : ASystem	
PROGRAM : AnalogInput	
Release :	Ver :1.00
Author :	Date:04/02/2011
Note :	Page:1 of 2

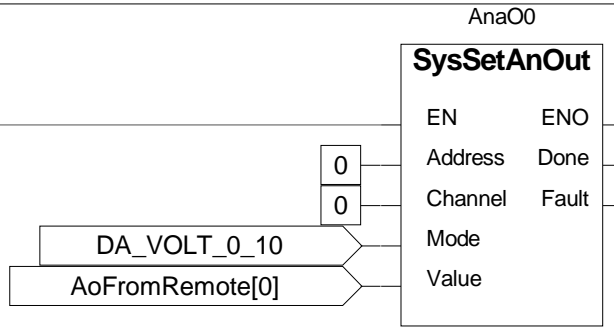


Project : ASystem	
PROGRAM : AnalogInput	
Release :	Ver :1.00
Author :	Date:04/02/2011
Note :	Page:2 of 2

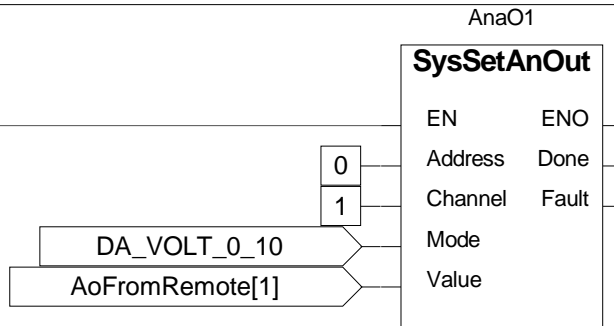
```

VAR
Ana00 : SysSetAnOut; (* Analog output management *)
Ana01 : SysSetAnOut; (* Analog output management *)
END_VAR
    
```

1



2



Project : ASystem	
PROGRAM : AnalogOutput	
Release :	Ver :1.00
Author :	Date:04/02/2011
Note :	Page:1 of 1

Returns the DB100 address offset
 ENCRYPTED CODE

```

VAR_INPUT
Offset : UINT; (* Address offset *)
END_VAR

VAR_OUTPUT
Address : UDINT; (* Address value *)
END_VAR
    
```

1

Project : ASystem	
FUNCTION BLOCK : DB100AddOffset	
Release :	Ver :1.00
Author :	Date:04/02/2011
Note :	Page:1 of 1

FUNCTION_BLOCK BitToByte

Merges 8 BOOL variables into a BYTE
ENCRYPTED CODE

```
VAR_INPUT
b0 : BOOL;
b1 : BOOL;
b2 : BOOL;
b3 : BOOL;
b4 : BOOL;
b5 : BOOL;
b6 : BOOL;
b7 : BOOL;
END_VAR

VAR_OUTPUT
Out : BYTE; (* Function result *)
END_VAR
```

1

	Project : ASystem	
	FUNCTION BLOCK : BitToByte	
	Release :	Ver :1.00
	Author :	Date:04/02/2011
	Note :	Page:1 of 1

FUNCTION_BLOCK ByteToBit

Splits BYTE variable into 8 BOOL
ENCRYPTED CODE

```
VAR_INPUT
In : BYTE; (* Input data *)
END_VAR

VAR_OUTPUT
b0 : BOOL; (* Bit 0 result *)
b1 : BOOL; (* Bit 1 result *)
b2 : BOOL; (* Bit 2 result *)
b3 : BOOL; (* Bit 3 result *)
b4 : BOOL; (* Bit 4 result *)
b5 : BOOL; (* Bit 5 result *)
b6 : BOOL; (* Bit 6 result *)
b7 : BOOL; (* Bit 7 result *)
END_VAR
```

1

Project : ASystem	
FUNCTION BLOCK : ByteToBit	
Release :	Ver :1.00
Author :	Date:04/02/2011
Note :	Page:1 of 1

FUNCTION_BLOCK ByteToWorld

Merges 2 BYTE variables into a WORD
ENCRYPTED CODE

```
VAR_INPUT
MSB : BYTE; (* MSB Value *)
LSB : BYTE; (* LSB Value *)
END_VAR

VAR_OUTPUT
Out : WORD; (* Function result *)
END_VAR
```

1

	Project : ASystem	
	FUNCTION BLOCK : ByteToWorld	
	Release :	Ver :1.00
	Author :	Date:04/02/2011
	Note :	Page:1 of 1

FUNCTION_BLOCK WordToByte

Splits WORD variable into 2 BYTES
ENCRYPTED CODE

VAR_INPUT
In : WORD; (* Input data *)
END_VAR

VAR_OUTPUT
MSB : BYTE; (* MSB Result *)
LSB : BYTE; (* LSB Result *)
END_VAR

1

	Project : ASystem	
	FUNCTION BLOCK : WordToByte	
	Release :	Ver :1.00
	Author :	Date:04/02/2011
	Note :	Page:1 of 1

FUNCTION_BLOCK CPUModuleIO

Manages the logic I/O on the CPU module
ENCRYPTED CODE

```
VAR_INPUT
Enable : BOOL; (* Function enable *)
DO00 : BOOL; (* Digital output 0 *)
DO01 : BOOL; (* Digital output 1 *)
END_VAR

VAR_OUTPUT
Enabled : BOOL; (* Function enabled *)
Done : BOOL; (* Function done *)
Fault : BOOL; (* Function fault *)
DI00 : BOOL; (* Digital input 0 *)
DI01 : BOOL; (* Digital input 0 *)
END_VAR
```

1

	Project : ASystem	
	FUNCTION BLOCK : CPUModuleIO	
	Release :	Ver :1.00
	Author :	Date:04/02/2011
	Note :	Page:1 of 1

ENCRYPTED CODE

```

VAR_INPUT
Enable : BOOL; (* FB enable *)
File : FILEP; (* UDP socket pointer *)
PeerIP : STRING[ 15 ]; (* IP adress *)
Port : UINT; (* Port number *)
RxDBuffer : @USINT; (* Rx data buffer *)
TxDBuffer : @USINT; (* Tx data buffer *)
ByteNr : UINT; (* Byte to exchange *)
Timeout : UINT; (* Timeout time (mS) *)
END_VAR
    
```

```

VAR_OUTPUT
Done : BOOL; (* Function done *)
Fault : BOOL; (* Function fault *)
RxDataOk : BOOL; (* Data received *)
TxDataSent : BOOL; (* Data sent *)
END_VAR
    
```

1

	Project : ASystem	
	FUNCTION BLOCK : UDPDataTxfer	
	Release :	Ver :1.00
	Author :	Date:04/02/2011
	Note :	Page:1 of 1