

Bloc Fonction



OMRON ELECTRONICS S.A.S.
14 Rue de Lisbonne
93561 Rosny-sous-Bois cedex

Référence	MRTU_CPU_Master
Révision	1.3
Auteur	JP Viskovic
Date	09/04/15
+ Support	http://support-omron.fr/

Bloc fonction Modbus RTU Master port série Hostlink (CPU)

Fonction	Modbus RTU maître destiné au port série Hostlink																																																
Symboles	<table border="0" style="width: 100%;"> <tr> <td style="text-align: center;"> <p>Write_Register</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p style="text-align: center;">MRTU_CPU_Fn06</p> <table border="0" style="width: 100%;"> <tr> <td>(BOOL) EN</td> <td>(BOOL) ENO</td> </tr> <tr> <td>(UINT) Slave_No</td> <td>(BOOL) Busy</td> </tr> <tr> <td>(UINT) Register_Address</td> <td>(BOOL) Error</td> </tr> <tr> <td>(INT) Value</td> <td>(BOOL) Done</td> </tr> <tr> <td>(BOOL) Cmd_Write</td> <td></td> </tr> </table> </div> </td> <td style="text-align: center;"> <p>Write_Coil</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p style="text-align: center;">MRTU_CPU_Fn05</p> <table border="0" style="width: 100%;"> <tr> <td>(BOOL) EN</td> <td>(BOOL) ENO</td> </tr> <tr> <td>(UINT) Slave_No</td> <td>(BOOL) Busy</td> </tr> <tr> <td>(UINT) Coil_Address</td> <td>(BOOL) Error</td> </tr> <tr> <td>(BOOL) Value</td> <td>(BOOL) Done</td> </tr> <tr> <td>(BOOL) Cmd_Write</td> <td></td> </tr> </table> </div> </td> </tr> <tr> <td style="text-align: center;"> <p>Read_Registers</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p style="text-align: center;">MRTU_CPU_Fn03</p> <table border="0" style="width: 100%;"> <tr> <td>(BOOL) EN</td> <td>(BOOL) ENO</td> </tr> <tr> <td>(UINT) Slave_No</td> <td>(BOOL) Busy</td> </tr> <tr> <td>(UINT) Register_Address</td> <td>(BOOL) Error</td> </tr> <tr> <td>(UINT) Register_Qty</td> <td>(BOOL) Done</td> </tr> <tr> <td>(UINT) RespData_DM</td> <td></td> </tr> <tr> <td>(BOOL) Cmd_Read</td> <td></td> </tr> </table> </div> </td> <td style="text-align: center;"> <p>Write_Registers</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p style="text-align: center;">MRTU_CPU_Fn10</p> <table border="0" style="width: 100%;"> <tr> <td>(BOOL) EN</td> <td>(BOOL) ENO</td> </tr> <tr> <td>(UINT) Slave_No</td> <td>(BOOL) Busy</td> </tr> <tr> <td>(UINT) Register_Address</td> <td>(BOOL) Error</td> </tr> <tr> <td>(UINT) Register_Qty</td> <td>(BOOL) Done</td> </tr> <tr> <td>(UINT) DataAddress_DM</td> <td></td> </tr> <tr> <td>(BOOL) Cmd_Write</td> <td></td> </tr> </table> </div> </td> </tr> </table>	<p>Write_Register</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p style="text-align: center;">MRTU_CPU_Fn06</p> <table border="0" style="width: 100%;"> <tr> <td>(BOOL) EN</td> <td>(BOOL) ENO</td> </tr> <tr> <td>(UINT) Slave_No</td> <td>(BOOL) Busy</td> </tr> <tr> <td>(UINT) Register_Address</td> <td>(BOOL) Error</td> </tr> <tr> <td>(INT) Value</td> <td>(BOOL) Done</td> </tr> <tr> <td>(BOOL) Cmd_Write</td> <td></td> </tr> </table> </div>	(BOOL) EN	(BOOL) ENO	(UINT) Slave_No	(BOOL) Busy	(UINT) Register_Address	(BOOL) Error	(INT) Value	(BOOL) Done	(BOOL) Cmd_Write		<p>Write_Coil</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p style="text-align: center;">MRTU_CPU_Fn05</p> <table border="0" style="width: 100%;"> <tr> <td>(BOOL) EN</td> <td>(BOOL) ENO</td> </tr> <tr> <td>(UINT) Slave_No</td> <td>(BOOL) Busy</td> </tr> <tr> <td>(UINT) Coil_Address</td> <td>(BOOL) Error</td> </tr> <tr> <td>(BOOL) Value</td> <td>(BOOL) Done</td> </tr> <tr> <td>(BOOL) Cmd_Write</td> <td></td> </tr> </table> </div>	(BOOL) EN	(BOOL) ENO	(UINT) Slave_No	(BOOL) Busy	(UINT) Coil_Address	(BOOL) Error	(BOOL) Value	(BOOL) Done	(BOOL) Cmd_Write		<p>Read_Registers</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p style="text-align: center;">MRTU_CPU_Fn03</p> <table border="0" style="width: 100%;"> <tr> <td>(BOOL) EN</td> <td>(BOOL) ENO</td> </tr> <tr> <td>(UINT) Slave_No</td> <td>(BOOL) Busy</td> </tr> <tr> <td>(UINT) Register_Address</td> <td>(BOOL) Error</td> </tr> <tr> <td>(UINT) Register_Qty</td> <td>(BOOL) Done</td> </tr> <tr> <td>(UINT) RespData_DM</td> <td></td> </tr> <tr> <td>(BOOL) Cmd_Read</td> <td></td> </tr> </table> </div>	(BOOL) EN	(BOOL) ENO	(UINT) Slave_No	(BOOL) Busy	(UINT) Register_Address	(BOOL) Error	(UINT) Register_Qty	(BOOL) Done	(UINT) RespData_DM		(BOOL) Cmd_Read		<p>Write_Registers</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p style="text-align: center;">MRTU_CPU_Fn10</p> <table border="0" style="width: 100%;"> <tr> <td>(BOOL) EN</td> <td>(BOOL) ENO</td> </tr> <tr> <td>(UINT) Slave_No</td> <td>(BOOL) Busy</td> </tr> <tr> <td>(UINT) Register_Address</td> <td>(BOOL) Error</td> </tr> <tr> <td>(UINT) Register_Qty</td> <td>(BOOL) Done</td> </tr> <tr> <td>(UINT) DataAddress_DM</td> <td></td> </tr> <tr> <td>(BOOL) Cmd_Write</td> <td></td> </tr> </table> </div>	(BOOL) EN	(BOOL) ENO	(UINT) Slave_No	(BOOL) Busy	(UINT) Register_Address	(BOOL) Error	(UINT) Register_Qty	(BOOL) Done	(UINT) DataAddress_DM		(BOOL) Cmd_Write	
<p>Write_Register</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p style="text-align: center;">MRTU_CPU_Fn06</p> <table border="0" style="width: 100%;"> <tr> <td>(BOOL) EN</td> <td>(BOOL) ENO</td> </tr> <tr> <td>(UINT) Slave_No</td> <td>(BOOL) Busy</td> </tr> <tr> <td>(UINT) Register_Address</td> <td>(BOOL) Error</td> </tr> <tr> <td>(INT) Value</td> <td>(BOOL) Done</td> </tr> <tr> <td>(BOOL) Cmd_Write</td> <td></td> </tr> </table> </div>	(BOOL) EN	(BOOL) ENO	(UINT) Slave_No	(BOOL) Busy	(UINT) Register_Address	(BOOL) Error	(INT) Value	(BOOL) Done	(BOOL) Cmd_Write		<p>Write_Coil</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p style="text-align: center;">MRTU_CPU_Fn05</p> <table border="0" style="width: 100%;"> <tr> <td>(BOOL) EN</td> <td>(BOOL) ENO</td> </tr> <tr> <td>(UINT) Slave_No</td> <td>(BOOL) Busy</td> </tr> <tr> <td>(UINT) Coil_Address</td> <td>(BOOL) Error</td> </tr> <tr> <td>(BOOL) Value</td> <td>(BOOL) Done</td> </tr> <tr> <td>(BOOL) Cmd_Write</td> <td></td> </tr> </table> </div>	(BOOL) EN	(BOOL) ENO	(UINT) Slave_No	(BOOL) Busy	(UINT) Coil_Address	(BOOL) Error	(BOOL) Value	(BOOL) Done	(BOOL) Cmd_Write																													
(BOOL) EN	(BOOL) ENO																																																
(UINT) Slave_No	(BOOL) Busy																																																
(UINT) Register_Address	(BOOL) Error																																																
(INT) Value	(BOOL) Done																																																
(BOOL) Cmd_Write																																																	
(BOOL) EN	(BOOL) ENO																																																
(UINT) Slave_No	(BOOL) Busy																																																
(UINT) Coil_Address	(BOOL) Error																																																
(BOOL) Value	(BOOL) Done																																																
(BOOL) Cmd_Write																																																	
<p>Read_Registers</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p style="text-align: center;">MRTU_CPU_Fn03</p> <table border="0" style="width: 100%;"> <tr> <td>(BOOL) EN</td> <td>(BOOL) ENO</td> </tr> <tr> <td>(UINT) Slave_No</td> <td>(BOOL) Busy</td> </tr> <tr> <td>(UINT) Register_Address</td> <td>(BOOL) Error</td> </tr> <tr> <td>(UINT) Register_Qty</td> <td>(BOOL) Done</td> </tr> <tr> <td>(UINT) RespData_DM</td> <td></td> </tr> <tr> <td>(BOOL) Cmd_Read</td> <td></td> </tr> </table> </div>	(BOOL) EN	(BOOL) ENO	(UINT) Slave_No	(BOOL) Busy	(UINT) Register_Address	(BOOL) Error	(UINT) Register_Qty	(BOOL) Done	(UINT) RespData_DM		(BOOL) Cmd_Read		<p>Write_Registers</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p style="text-align: center;">MRTU_CPU_Fn10</p> <table border="0" style="width: 100%;"> <tr> <td>(BOOL) EN</td> <td>(BOOL) ENO</td> </tr> <tr> <td>(UINT) Slave_No</td> <td>(BOOL) Busy</td> </tr> <tr> <td>(UINT) Register_Address</td> <td>(BOOL) Error</td> </tr> <tr> <td>(UINT) Register_Qty</td> <td>(BOOL) Done</td> </tr> <tr> <td>(UINT) DataAddress_DM</td> <td></td> </tr> <tr> <td>(BOOL) Cmd_Write</td> <td></td> </tr> </table> </div>	(BOOL) EN	(BOOL) ENO	(UINT) Slave_No	(BOOL) Busy	(UINT) Register_Address	(BOOL) Error	(UINT) Register_Qty	(BOOL) Done	(UINT) DataAddress_DM		(BOOL) Cmd_Write																									
(BOOL) EN	(BOOL) ENO																																																
(UINT) Slave_No	(BOOL) Busy																																																
(UINT) Register_Address	(BOOL) Error																																																
(UINT) Register_Qty	(BOOL) Done																																																
(UINT) RespData_DM																																																	
(BOOL) Cmd_Read																																																	
(BOOL) EN	(BOOL) ENO																																																
(UINT) Slave_No	(BOOL) Busy																																																
(UINT) Register_Address	(BOOL) Error																																																
(UINT) Register_Qty	(BOOL) Done																																																
(UINT) DataAddress_DM																																																	
(BOOL) Cmd_Write																																																	
Fichier	MRTU_CPU_Master.zip																																																
API	<ul style="list-style-type: none"> - port série de CJ1/CS1, CJ2H/M - port série de CP1L-J14/J20/L14/L20 - port série droit de CP1H-X/XA/Y - port série droit de CP1L/M20/M40/M60 - port série droit de CP1L-EL/EM 																																																
Condition d'utilisation	<p>Les blocs fonction MRTU_CPU_Master sont proposés 'tel que' et peuvent servir de base de développement. Les utilisateurs doivent, au préalable, tester leur adéquation avec l'application finale.</p> <p>Omron France ne pourra en aucun cas être tenu pour responsable en cas de dysfonctionnement de l'application finale.</p>																																																
Principe	<p>Configuration du port série en mode RS-232C et 8 bits de données aucun délimiteur.</p> <p>Le switch en façade relatif au port série doit être positionné sur « Setup » configuration utilisateur (voir switch)</p> <p>Liste des commandes de lecture/écriture implémentées</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Code</th> <th>Fonction Modbus</th> <th>Bloc fonction</th> </tr> </thead> <tbody> <tr> <td>0x03</td> <td>Read Holding Registers</td> <td>MRTU_CPU_Fn03</td> </tr> <tr> <td>0x05</td> <td>Write Single Coil</td> <td>MRTU_CPU_Fn05</td> </tr> <tr> <td>0x06</td> <td>Write Single Register</td> <td>MRTU_CPU_Fn06</td> </tr> <tr> <td>0x10</td> <td>Write Multiple Registers</td> <td>MRTU_CPU_Fn10</td> </tr> </tbody> </table>	Code	Fonction Modbus	Bloc fonction	0x03	Read Holding Registers	MRTU_CPU_Fn03	0x05	Write Single Coil	MRTU_CPU_Fn05	0x06	Write Single Register	MRTU_CPU_Fn06	0x10	Write Multiple Registers	MRTU_CPU_Fn10																																	
Code	Fonction Modbus	Bloc fonction																																															
0x03	Read Holding Registers	MRTU_CPU_Fn03																																															
0x05	Write Single Coil	MRTU_CPU_Fn05																																															
0x06	Write Single Register	MRTU_CPU_Fn06																																															
0x10	Write Multiple Registers	MRTU_CPU_Fn10																																															

Remarque : Les échanges Modbus peuvent être vérifiés à l'aide de l'utilitaire [Multiway](#)

1- Variables d'Entrée du bloc fonction MRTU_CPU_Fn03

Entrées	type	valeur	Description
EN	Bool	OFF, ON	Activation du FB
Slave_No	UINT	0-00FF	Numéro d'esclave
Register_Address	UINT	0-FFFF	Adresse du 1er registre
Register_Qty	UINT	0-00FF	Nombre de registres
RespData_DM	UINT	0-FFFF	Offset de consignation des valeurs lues en zone DM (ex : &100 pour le D100)
Cmd_Read	Bool	OFF, ON	Commande de lecture

2- Variables d'Entrée du bloc fonction MRTU_CPU_Fn05

Entrées	type	valeur	Description
EN	Bool	OFF, ON	Activation du FB
Slave_No	UINT	0-00FF	Numéro d'esclave
Coil_Address	UINT	0-FFFF	Adresse de la bobine
Value	Bool	OFF, ON	Valeur de la bobine
Cmd_Write	Bool	OFF, ON	Commande d'écriture

3- Variables d'Entrée du bloc fonction MRTU_CPU_Fn06

Entrées	type	valeur	Description
EN	Bool	OFF, ON	Activation du FB
Slave_No	UINT	0-00FF	Numéro d'esclave
Register_Address	UINT	0-FFFF	Adresse du registre
Value	UINT	0-FFFF	Valeur du registre
Cmd_Write	Bool	OFF, ON	Commande d'écriture

4- Variables d'Entrée du bloc fonction MRTU_CPU_Fn10

Entrées	type	valeur	Description
EN	Bool	OFF, ON	Activation du FB
Slave_No	UINT	0-00FF	Numéro d'esclave
Register_Address	UINT	0-FFFF	Adresse du 1er registre
Register_Qty	UINT	0-00FF	Nombre de registres
DataAddress_DM	UINT	0-FFFF	Offset du DM à partir duquel sont consignées les données à écrire (ex : &100 pour désigner le D100)
Cmd_Write	Bool	OFF, ON	Commande d'écriture

Variables de sortie des bloc MRTU_CPU_FN03, FN05, Fn06 et Fn10

Sorties	type	Range	Description
ENO	Bool	OFF, ON	ON : port Hostlink disponible
Busy	Bool	OFF, ON	Commande en cours
Error	Bool	OFF, ON	Drapeau d'erreur
Done	Bool	OFF, ON	Commande exécutée (quel que soit le drapeau d'erreur !!)