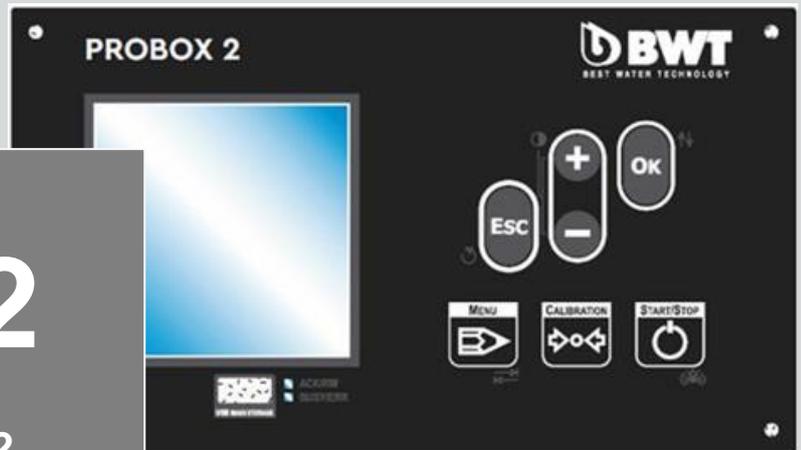


# PROBOX 2

Analyseur multi-paramètres – 2  
voies



FR -NOTICE - COMMUNICATIONS

**TRÈS IMPORTANT** : Avant tout raccordement et utilisation, lire attentivement la présente notice. Le non-respect de ces prescriptions, entraîne la déchéance de la garantie **BWT**.

2 autres notices sont disponibles

- P0012276A Installation, mise en service, maintenance.
- P0012276B Programmation.

## SOMMAIRE

I.	Consignes de sécurité et d'environnement.....	4
1)	Utilisation de l'équipement.....	4
2)	Obligations de l'utilisateur.....	4
3)	Prévention des risques.....	4
II.	Synoptiques fondamentaux de communication.....	5
1)	Connexion locale avec un logiciel de maintenance.....	5
III.	Connexion.....	6
1)	Connexion sur le port RS485 avec adaptateur RS485/USB.....	6
IV.	Programmation de l'appareil.....	7
1)	Communication RS485 sur BWT PROBOX 2.....	7
I.	Registres MODBUS.....	10
1)	Adresse des registres Modbus.....	10
2)	Formatage des données.....	15

## I. Consignes de sécurité et d'environnement

Veillez :

- Lire attentivement ce manuel avant de déballer, de monter ou de mettre en service cet équipement
- Tenir compte de tous les dangers et mesures de précaution préconisées

Le non-respect de ces procédures est susceptible de blesser gravement les intervenants ou d'endommager l'appareil.

### 1) Utilisation de l'équipement

Les équipements **BWT PROBOX 2** ont été conçus pour mesurer et réguler deux paramètres choisis à l'aide de capteurs et de commandes d'actionneurs appropriés dans le cadre des possibilités d'utilisation décrites dans le présent manuel.



Toute utilisation différente sera considérée comme non-conforme et doit être proscrite. BWT France n'assumera en aucun cas la responsabilité et les dommages qui en résultent.

### 2) Obligations de l'utilisateur

L'utilisateur s'engage à ne laisser travailler avec les équipements **BWT PROBOX 2** décrits dans ce manuel que le personnel qui :

- Est sensibilisé avec les consignes fondamentales relatives à la sécurité du travail et de la prévention des accidents
- Est formé à l'utilisation de l'appareil et de son environnement
- A lu et compris la présente notice, les avertissements et les règles de manipulation

### 3) Prévention des risques



L'installation et le raccordement des équipements **BWT PROBOX 2** ne doivent être effectués que par un personnel spécialisé et qualifié pour cette tâche.

L'installation doit respecter les normes et les consignes de sécurité en vigueur !



Avant de mettre l'appareil sous tension ou de manipuler les sorties relais, veuillez toujours couper l'alimentation électrique primaire !

Ne jamais ouvrir l'appareil sous tension !

Les opérations d'entretien et les réparations doivent être uniquement effectuées par un personnel habilité et spécialisé !



Veillez à bien choisir le lieu d'installation des équipements en fonction de l'environnement !

Le boîtier électronique **BWT PROBOX 2** ne doit pas être installé dans un environnement à risque et doit être mis à l'abri des projections d'eau et des produits chimiques. Il doit être installé dans un endroit sec et ventilé, isolé des vapeurs corrosives.



S'assurer que les capteurs chimiques utilisés avec cet appareil correspondent bien aux produits chimiques utilisés. Reportez-vous à la notice technique individuelle de chaque

capteur. La chimie de l'eau est très complexe, en cas de doute, contacter immédiatement notre service technique ou votre installateur agréé.

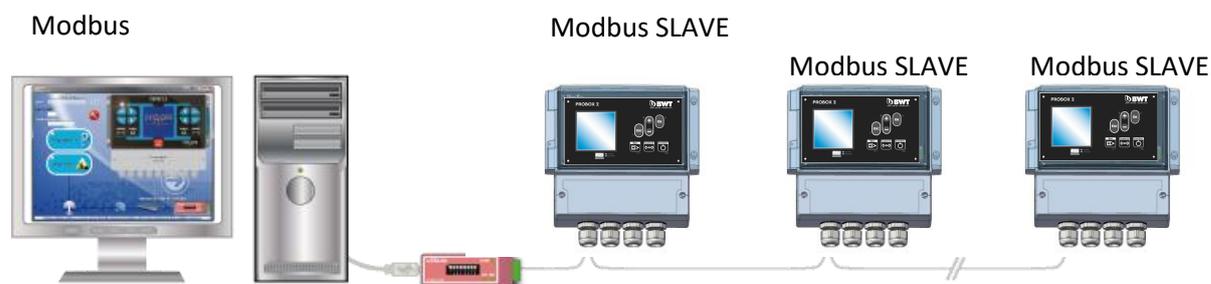


Les capteurs chimiques sont des éléments sensibles et dotés de parties consommables. Ils doivent être surveillés, entretenus et étalonnés régulièrement à l'aide de trousse d'analyses spécifiques non-fournies avec cet équipement. En cas de défaut, un risque potentiel d'injection excédentaire de produit chimique peut être constaté. Dans le doute, un contrat d'entretien doit être contracté auprès de votre installateur ou à défaut auprès de nos services techniques. Contacter votre installateur agréé ou notre service commercial pour plus d'informations.

## II. Synoptiques fondamentaux de communication

Les équipements **BWT PROBOX 2** ont été conçus pour être connectés sur un bus RS485 avec un protocole ModBus RTU ou ASCII. Plusieurs appareils peuvent être connectés les uns aux autres.

### 1) Connexion locale avec un logiciel de maintenance



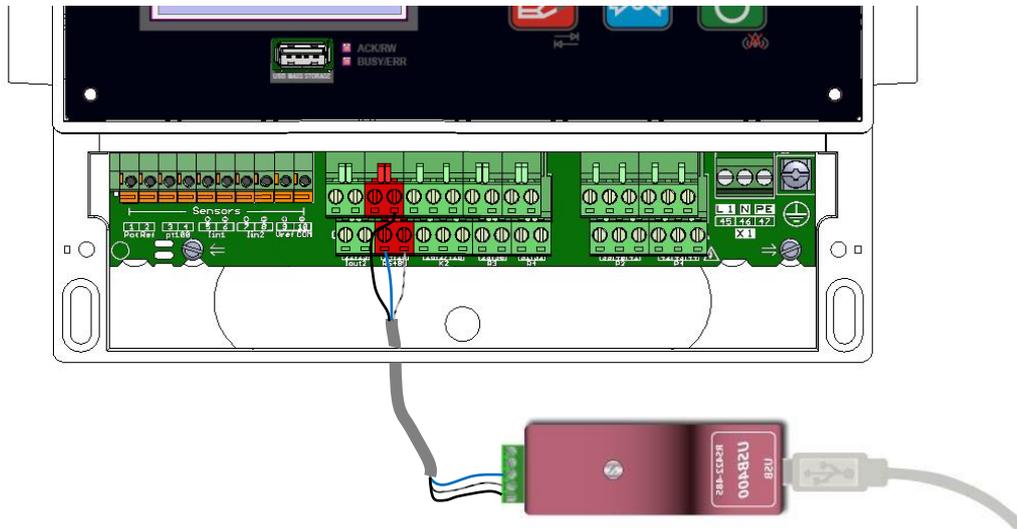
Connexion d'un ou plusieurs régulateurs **BWT PROBOX 2** via le BUS RS485.

Afin de connecter votre **BWT PROBOX 2** à votre ordinateur, nous vous proposons un module d'interface USB/RS485.

Référence	Désignation
P0943349	Convertisseur USB 485

### III. Connexion

#### 1) Connexion sur le port RS485 avec adaptateur RS485/USB



Les systèmes peuvent être chaînés en respectant l'ordre, et en repartant d'un système vers l'autre.



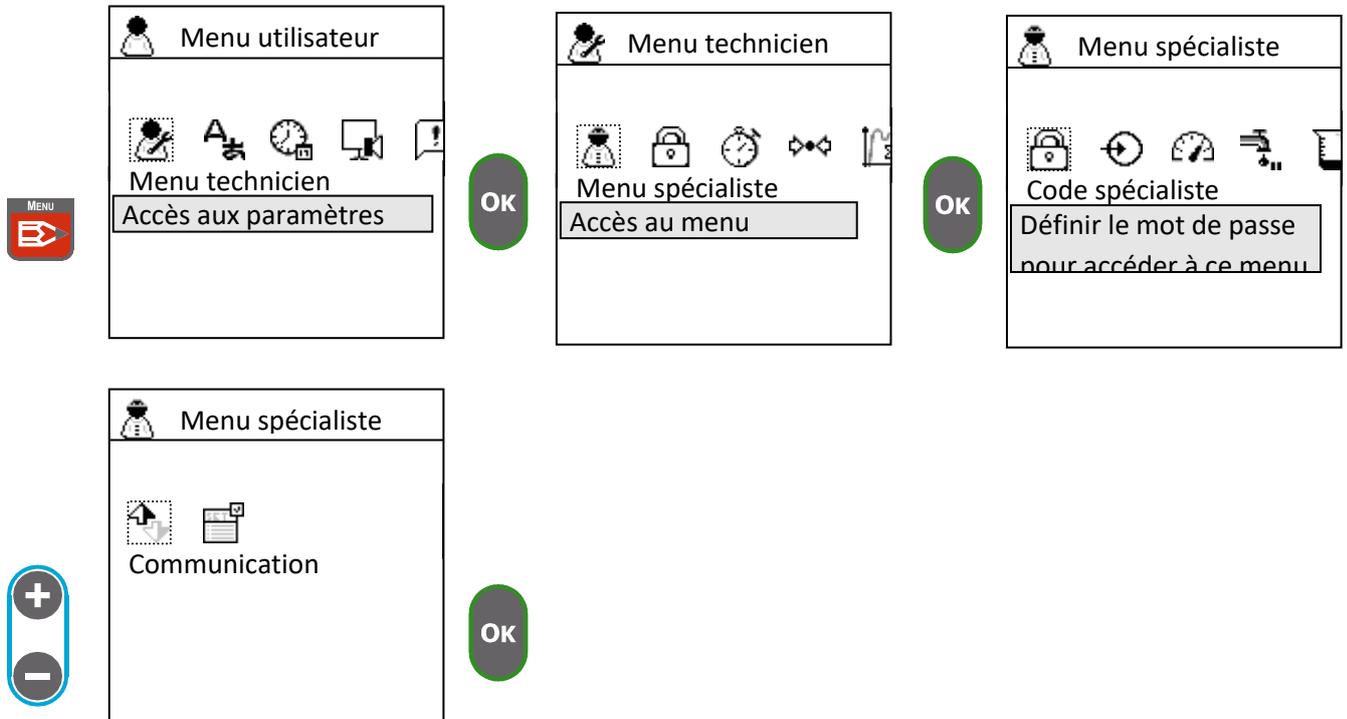
- Bleu (Bornier n°3) : AA' RS485
- Blanc (Bornier n°4) : BB' RS485
- Noir (Bornier n°5) : Masse RS485

Configuration : tous les interrupteurs sur "ON"

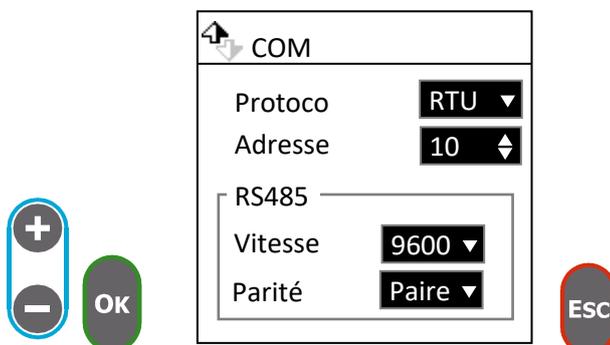
#### IV. Programmation de l'appareil

##### 1) Communication RS485 sur BWT PROBOX 2

Pour connecter un appareil sur un réseau, vous devez configurer tous les appareils avec les mêmes paramètres.



- ▶ Sélectionnez le protocole de communication.
- ▶ Choisissez l'adresse de l'appareil.
- ▶ Sélectionnez la vitesse de transfert.
- ▶ Sélectionnez la parité de la liaison série.



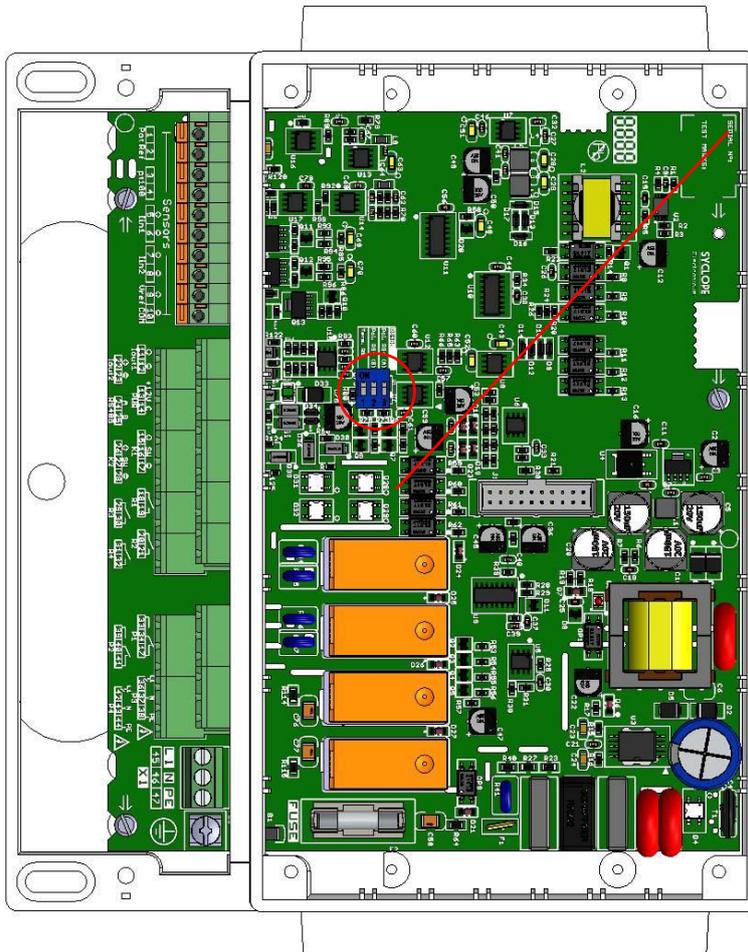
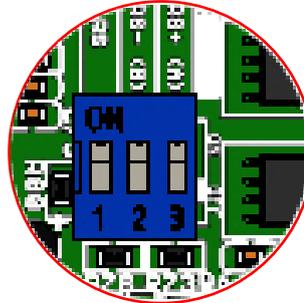
Tous les appareils connectés sur la même liaison RS485 doivent être à la même vitesse, avoir la même parité et disposer d'adresses différentes.

Nom	Signification	Réglage	Valeurs par défaut
Protocole	Protocole de communication	RTU/ASCII	RTU
Adresse	Adresse de l'appareil sur le réseau «slave ID»	1...247	10
Vitesse	Vitesse de communication de la liaison RS485	300, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200	9600
Parité	Parité de communication de la liaison RS485	Aucune, Paire, Impaire	Paire



Trois interrupteurs se trouvent sur la carte électronique de l'appareil pour gérer la terminaison et la polarisation des lignes de communication RS485. Par défaut votre appareil est livré avec tous les interrupteurs sur la position OFF.

- 1 : Terminaison
- 2 : Polarisation B
- 3 : Polarisation A



## V. Registres MODBUS

### 1) Adresse des registres Modbus

Les registres sont numérotés conformément du standard MODBUS. Ce sont des "HOLDING REGISTER" sur la plage de registres de 40001 à 49999.

Certain logiciels Modbus et automates utilisent un adressage de 0 à 65535.

Le registre ModBus 40001 correspond donc à l'adresse Modbus 0, le registre 40002 correspond à l'adresse 1 et ainsi de suite.

Registre ModBus	Nombre de registre	Nom	Accès	Format	Description
<b>Configuration</b>					
40001	788	eprom	rw	STRUCT	Mémoire
<b>Interfaces</b>					
41001	2	signal_POT	r	REAL	Signal sur l'entrée POT [mV]
41003	2	signal_RTD	r	REAL	Valeur de résistance sur l'entrée RTD [ohm]
41005	2	signal_IIN1	r	REAL	Courant sur IN1 [mA]
41007	2	signal_IIN2	r	REAL	Courant sur IN2 [mA]
41009	2	signal_K1	r	REAL	Fréquence sur K1 [Hz]
41011	2	signal_K2	r	REAL	Fréquence sur K2 [Hz]
41013	1	supply_IN	r	BOOL	0=12V / 1=24V
41014	1	state_IN1	r	BOOL	0=OUVERT / 1=FERME
41015	1	state_IN2	r	BOOL	0=OUVERT / 1=FERME
41016	1	state_K1	r	BOOL	0=OUVERT / 1=FERME
41017	1	state_K2	r	BOOL	0=OUVERT / 1=FERME
41018	1	state_P1	r	BOOL	0=OUVERT / 1=FERME
41019	1	state_P2	r	BOOL	0=OUVERT / 1=FERME
41020	1	state_P3	r	BOOL	0=OUVERT / 1=FERME
41021	1	state_P4	r	BOOL	0=OUVERT / 1=FERME

41021	1	state_R1	r	BOOL	0=OUVERT / 1=FERME
41023	1	state_R2	r	BOOL	0=OUVERT / 1=FERME
41024	1	state_R3	r	BOOL	0=OUVERT / 1=FERME
41025	1	state_R4	r	BOOL	0=OUVERT / 1=FERME
41026	2	current_IOUT1	r	REAL	Courant sur OUT1 [mA]
41028	2	current_IOUT2	r	REAL	Courant sur OUT2 [mA]
41030	2	Timestamp Local	r	DWORD	Temps depuis le 1 <sup>er</sup> janvier 1970 0h00 [s]
41032	2	Run time	r	DWORD	Temps depuis le démarrage de la machine
<b>Valeurs et états</b>					
41101	1	device_state	rw	WORD	Bit 0: appareil en marche Bit 1: timer en marche Bit 2: appareil en cours de démarrage Bit 3: appareil à l'arrêt à cause d'un timer
41201	2	param_E1_state	rw	DWORD	Bit 0: régulation et alarmes en marche Bit 1: capteur en cours de démarrage Bit 2: pause temporaire Bit 3: contact de circulation et débitmètre (1 == circulation) Bit 4: maintenance nécessaire Bit 5: en cours de dosage Bit 6: alarme(s) en cours Bit 7: régulation et alarme en pause dû à un timer Bit 8: capteurs hors échelle ou déconnectés Bit 9: capteurs hors échelle de mesure Bit 10: valeur du capteur instable Bit 11: seuil d'alarme basse franchi Bit 12: seuil d'alarme haut franchi Bit 13: temps de dosage max ou fond de

					cuve Bit14: utilisation d'un timer Bit15: commande à distance en cours Bit16: erreur de configuration de la voie
41203	2	param_E1_measure_value	r	REAL	Valeur de mesure [unité de mesure]
41205	2	param_E1_control_w	rw	REAL	Consigne de régulation [unité de mesure]
41207	2	param_E1_dosage_u	r	REAL	Commande de dosage [1/1]
41209	2	param_E1_alarm_high	rw	REAL	Valeur basse d'alarme [unité de mesure]
41211	2	param_E1_alarm_low	rw	REAL	Valeur haute d'alarme [unité de mesure]
				DWORD	Bit 0: régulation et alarmes en marche Bit 1: capteur en cours de démarrage Bit 2: pause temporaire Bit 3: contact de circulation et débitmètre (1 == circulation) Bit 4: maintenance nécessaire Bit 5: en cours de dosage Bit 6: alarme(s) en cours Bit 7: régulation et alarme en pause dû à un timer Bit 8: capteurs hors échelle ou déconnectés Bit 9: capteurs hors échelle de mesure Bit 10: valeur du capteur instable Bit 11: seuil d'alarme basse franchi Bit 12: seuil d'alarme haut franchi Bit 13: temps de dosage max ou fond de cuve bit14: utilisation d'un timer bit15: commande à distance en cours bit16: erreur de configuration de la voie
41301	2	param_E2_state	rw		
41303	2	param_E2_measure_value	r	REAL	Valeur de mesure [unité de mesure]

41305	2	param_E2_control_w	rw	REAL	Consigne de régulation [unité de mesure]
41307	2	param_E2_dosage_u	r	REAL	Commande de dosage [1/1]
41309	2	param_E2_alarm_high	rw	REAL	Valeur basse d'alarme [unité de mesure]
41311	2	param_E2_alarm_low	rw	REAL	Valeur haute d'alarme [unité de mesure]
41401	2	sensor_POT_value	r	REAL	Valeur de mesure du capteur POT [unité du capteur]
41403	2	sensorl_RTD_value	r	REAL	Valeur de mesure du capteur RTD [unité du capteur]
41405	2	sensor_IN1_value	r	REAL	Valeur de mesure du capteur IN1 [unité du capteur]
41407	2	sensor_IN2_value	r	REAL	Valeur de mesure du capteur IN2 [unité du capteur]
41409	2	sensor_K1_value	r	REAL	Valeur de mesure du capteur K1 [unité du capteur]
41411	2	sensor_K2_value	r	REAL	Valeur de mesure du capteur K2 [unité du capteur]
<b>Appareil</b>					
42001	22	device	r	STRUCT	Etats et valeur de l'appareil
<b>Voies</b>					
42101	124	param_E1	r	STRUCT	Etats et valeur de la voie E1
42301	124	param_E2	r	STRUCT	Etats et valeur de la voie E2
<b>Capteur</b>					
42501	34	sensor_POT	r	STRUCT	Etats et valeur du capteur POT
42551	34	sensor_RTD	r	STRUCT	Etats et valeur du capteur RTD
42601	34	sensor_IIN1	r	STRUCT	Etats et valeur du capteur IN1
42651	34	sensor_IIN2	r	STRUCT	Etats et valeur du capteur IN2
42701	34	sensor_K1	r	STRUCT	Etats et valeur du capteur K1
42751	34	sensor_K2	r	STRUCT	Etats et valeur du capteur K2
<b>Contacts</b>					
42801	10	switch_IIN1	r	STRUCT	Etat du contact IN1
42821	10	switch_IIN2	r	STRUCT	Etat du contact IN2

42841	10	switch_K1	r	STRUCT	Etat du contact K1
42861	10	switch_K2	r	STRUCT	Etat du contact K2
<b>Sorties 0/4-20mA</b>					
42901	16	iout_1	r	STRUCT	Etats et valeur de la sortie OUT1
42921	16	iout_2	r	STRUCT	Etats et valeur de la sortie OUT2
<b>Relais</b>					
43001	56	relay_P1	r	STRUCT	Etats et valeur du relais P1
43101	56	relay_P2	r	STRUCT	Etats et valeur du relais P2
43201	56	relay_P3	r	STRUCT	Etats et valeur du relais P3
43301	56	relay_P4	r	STRUCT	Etats et valeur du relais P4
43401	56	relay_R1	r	STRUCT	Etats et valeur du relais R1
43501	56	relay_R2	r	STRUCT	Etats et valeur du relais R2
43601	56	relay_R3	r	STRUCT	Etats et valeur du relais R3
43701	56	relay_R4	r	STRUCT	Etats et valeur du relais R4
<b>Evènements</b>					
44001	36	device_1_event	r	STRUCT	Etat du timer de l'appareil
44051	36	param_E1_event	r	STRUCT	Etat du timer de la voie E1
44101	36	param_E2_event	r	STRUCT	Etat du timer de la voie E2
44151	36	relay_event_P1	r	STRUCT	Etat du timer du relais P1
44201	36	relay_event_P2	r	STRUCT	Etat du timer du relais P2
44251	36	relay_event_P3	r	STRUCT	Etat du timer du relais P3
44301	36	relay_event_P4	r	STRUCT	Etat du timer du relais P4
44351	36	relay_event_R1	r	STRUCT	Etat du timer du relais R1
44401	36	relay_event_R2	r	STRUCT	Etat du timer du relais R2
44451	36	relay_event_R3	r	STRUCT	Etat du timer du relais R3
44501	36	relay_event_R4	r	STRUCT	Etat du timer du relais R4
<b>Affichage</b>					
45000	2048	screen	r	STRUCT	Buffer de l'écran 2bpp 128x128pix

## 2) Formatage des données

### **BOOL**

“bool” utilise 1 registre et peut avoir deux valeurs 0 ou 1.

Exemple :

Le registre 41018 est l'état du relais P1.

REG(41018) = 0: relais ouvert

REG(41018) = 1: relais fermé

### **REAL**

“real” utilise 2 registres et permet de coder des valeurs à virgule flottante sur 32bits.

Exemple :

Le registre 41102 est la valeur de mesure de la voie E1, l'unité de cette valeur est l'unité sélectionnée dans le menu mesure de l'appareil.

Pour une valeur de mesure de 1.94ppm, l'encodage hexadécimale est 0x3FF851EC.

REG(41103) = 0x51EC

REG(41104) = 0x3FF8

### **WORD**

“word” utilise 1 registre pour coder un entier 16bits ou un champ de bits.

Exemple (bits) :

Le registre 41101 contient les indicateurs d'état de l'appareil.

REG(4101) = b0000000000000101

REG(41101)(bit00) = 1: l'appareil est en marche

REG(41101)(bit01) = 0: le timer n'est pas en marche

REG(41101)(bit02) = 1: la régulation et les alarmes d'au moins une voie de mesure est en cours de démarrage

REG(41101)(bit03) = 0: il n'y a aucun de timer actif

REG(41101)(bit04) = 0: non utilisé

REG(41101)(bit05) = 0: non utilisé

REG(41101)(bit06) = 0: non utilisé

REG(41101)(bit07) = 0: non utilisé

REG(41101)(bit08) = 0: non utilisé

REG(41101)(bit09) = 0: non utilisé

REG(41101)(bit10) = 0: non utilisé

REG(41101)(bit11) = 0: non utilisé

REG(41101)(bit12) = 0: non utilisé

REG(41101)(bit13) = 0: non utilisé

REG(41101)(bit14) = 0: non utilisé

REG(41101)(bit15) = 0: non utilisé

**DWORD**

“dword” utilise 2 registres et permet de coder un entier 32bits ou un champ de bits.

Exemple :

Le registre 41030 contient l’heure locale de l’appareil, cette heure correspond au nombre de secondes écoulées depuis le 1<sup>er</sup> janvier 1970.

Le 27 avril 2015 à 3h35min19sec correspond à 1430141719 secondes depuis la date de référence, la valeur hexadécimale est 0x553E3B17.

REG(41032) = 0x3B17

REG(41032) = 0x553E

**STRUCT (device)**

Ce block de données contient les états les valeurs et les configurations de l'appareil.

Nom	Taille [octets]	Décalage [octets]	Type	Description
dev	1	0	byte	Interne
name	12	1	string	Nom de l'appareil
align	3	13	--	Interne
fd	4	16	integer	Interne
flag	1	20	bits	bit0: régulation et alarme en marche bit1: timer en marche bit2: appareil en cours de démarrage bit3: appareil à l'arrêt à cause d'un timer
align	3	21	--	Interne
device.calendar.flag	1	24	bits	bit0: traitement du timer en marche bit1: évènement du timer en cours bit2: traitement du timer en pause temporaire
align	3	25	--	Interne
device.calendar.event_list	4	28	--	Interne
device.calendar.next	4	32	--	Interne
device.param.flag	3	36	bits	Bit 0: régulation et alarme en marche Bit 1: capteurs en cours de démarrage Bit 2: pause temporaire Bit 3: contact de circulation ou débitmètre (1 == circulation) Bit 4: maintenance nécessaire Bit 5: dosage en cours Bit 6: alarme active Bit 7: traitement de la régulation et des alarmes en pause à cause d'un

				timer Bit 8: capteurs hors échelle ou déconnecté Bit 9: capteurs hors échelle de mesure Bit 10: valeur de mesure instable Bit 11: seuil d'alarme bas franchi Bit 12: seuil d'alarme haut franchi Bit 13: tems max de dosage ou fond de cuve bit14: utilisation d'un timer bit15: commande à distance active bit16: erreur de configuration
align	1	39	--	Interne
next	4	40	--	Interne

Exemple :

Pour lire l'état marche/arrêt de l'appareil la base de registre est REG(42001).

Le décalage de "flag" est de  $(1+12+3+4) = 20$  octets

Le décalage en registre est donc de  $20/2 = 10$

Le registre de "flag" est REG(42001+20) = REG42021)

REG(42021) = 0x0100

Les données sont codées en "little endian" donc l'ordre des octets est inversé.

Flag = 0x01 l'appareil est en marche.

**STRUCT (param)**

Ce block de données contient les états, les valeurs et les configurations des voies de mesure.

Nom	Taille [octets]	Décalage [octets]	Type	Description
par	1	0	byte	Interne
align	3	1	--	Interne
fd	4	4	integer	internal
flag	3	8	bits	Bit 0: régulation et alarme en marche Bit 1: capteurs en cours de démarrage Bit 2: pause temporaire Bit 3: contact de circulation ou débitmètre (1 == circulation) Bit 4: maintenance nécessaire Bit 5: dosage en cours Bit 6: alarme active Bit 7: traitement de la régulation et des alarmes en pause à cause d'un timer Bit 8: capteurs hors échelle ou déconnecté Bit 9: capteurs hors échelle de mesure Bit 10: valeur de mesure instable Bit 11: seuil d'alarme bas franchi Bit 12: seuil d'alarme haut franchi Bit 13: tems max de dosage ou fond de cuve bit14: utilisation d'un timer bit15: commande à distance active bit16: erreur de configuration
align	1	11	--	interne
decice	4	12	--	interne

measure_sensor	16	16	--	interne
measure_kind	1	32	byte	Type de voie de mesure: 0: Inactif 1: Chlore libre 2: Chlore actif 3: Chlore total 4: Chloramines 5: Chlorite 6: Dioxyde de chlore 7: H2O2 8: BCDMH 9: DBDMH 10: Brome libre 11: Brome actif 12: Brome total 13: PAA 14: Ozone 15: Oxygène dissous 16: Nitrate 17: PHMB 18: Salinité 19: TDS 20: Turbidité 21: Conductivité 22: Température 23: Débit 24: pH 25: RedOx 26: Chloride

				27: Amonia 28: Fluoride 29: ISE 30: Volume
measure_unit	1	33	byte	Unité de mesure 0: Inactif 1: aucune 2: décade 3: pH 4: ppb 5: ppm 6: µg/l 7: mg/l 8: g/l 9: % 10: µS/cm <sup>2</sup> 11: mS/cm <sup>2</sup> 12: NTU 13: FNU 14: °K 15: °C 16: °F 17: °R 18: mA 19: mV 20: Hz 21: str/min 22: ms 23: sec

				24: min 25: h 26: l 27: m3 28: l/min 29: l/h 30: m3/h 31: imp/l 32: imp/m3 33: Ohm 34: mOhm 35: impulsion
align	2	34	--	Interne
measure_min_value	4	36	float	Valeur basse de mesure
measure_max_value	4	40	float	Valeur haute de mesure
measure_value	4	44	float	Valeur de mesure
measure_m_factor	4	48	float	Facteur de correction de la mesure [1/1]
measure_t_factor	4	52	float	Correction en température [%/°C]
alarm_flag	1	56	bits	Indicateurs d'alarme  bit0: valeur de mesure supérieure à l'alarme haute  bit1: valeur de mesure inférieure à l'alarme basse  bit2: temps de dosage max dépassé ou fond de cuve  bit3: capteurs déconnectés ou hors échelle
alarm_threshold_delay	1	57	byte	Temps de retard des alarmes[s]
alarm_threshold_tick	1	58	byte	Interne
align	1	59	--	Interne

alarm_threshold_hyst	4	60	float	Hystérésis des seuils d'alarme
alarm_threshold_low	4	64	float	Seuil bas
alarm_threshold_high	4	68	float	Seuil haut
flow_sensor	4	72	--	Interne
flow_switch	16	76	--	Interne
flow.op	1	92	byte	Condition de circulation 0: au moins 1 1 = tous
flow_unit	1	93	byte	Unité du débit: 28 : l/min 29 : l/h 30 : m3/h
align	2	94	--	Interne
flow_threshold	4	96	float	Seuil de débit pour indiquer l'arrêt de la circulation
flow_q_min	4	100	float	Valeur de débit pour la compensation de la régulation x0%
flow_q_max	4	104	float	Valeur de débit pour la compensation de la régulation x100%
flow_q	4	108	float	Valeur de débit en cours
control_flag	1	112	bits	bit0: 0 = 1DOF 1 = 2DOF bit[1~3]: mode de régulation 0 = inactif 1 = hystérésis 2 = seuils 3 = PID Bit4: fonction "hold" active
align	3	113	--	Interne
control_w	4	116	float	Consigne de régulation

control_x_dead	4	120	float	Bande morte ou hystérésis en fonction du mode de régulation.
control_xp	4	124	float	Valeur proportionnelle réciproque
control_ki	2	128	float	Coefficient d'intégrale
control_kd	2	130	float	Coefficient de dérivé
control_kb	4	132	float	Coefficient de retour de boucle
control.threshold_low	4	136	float	Seuil de régulation bas
control.threshold_high	4	140	float	Seuil de régulation haut
control_z_y	4	144	float	Valeur réglante
control_z_ex	4	148	float	Erreur de boucle ou entrée en fonction du nombre de degrés de liberté de la régulation
control_z_dex	4	152	float	Erreur de la dérivé
control_sum_e	4	156	float	Erreur de l'intégrale
dosage_flag	1	160	bits	bit[0~1]: sens de régulation 0 = montant 1 = descendant 2 = les deux bit2: compensation au débit bit3: dosage en pause
align	1	161	--	Interne
dosage.tick	2	162	short	Temps de surdosage
dosage.control_time	2	164	short	Limite de temps de dosage
align	2	166	--	Interne
dosage.control_threshold	4	168	float	Seuil de régulation pour le surdosage
dosage_u_bias	4	172	float	Charge de base
dosage_u	4	176	float	Commande de dosage
tank.switch_direct	4	180	--	Interne
tank.switch_invert	4	184	--	Interne
tank.sensor_direct	4	188	--	Interne

tank.sensor_invert	4	192	--	Interne
tank.threshold_direct	4	196	float	Seuil de fond de cuve du produit montant
tank.threshold_invert	4	200	float	Seuil de fond de cuve du produit descendant
remolte.calendar.flag	1	204	bits	bit0: traitement du timer en marche bit1: évènement du timer en cours bit2: traitement du timer en pause temporaire
align	3	205	--	Interne
remote.calendar.event_list	4	208	--	Interne
remote.calendar.next	4	212	--	Interne
remotre.flag	1	216	bits	bit[0~1]: source de commande à distance  0 = inactif 1 = timer 2 = capteur 3 = contact
align	3	217	--	Interne
remote.sensor	4	220	--	Interne
remote.control.w	4	224	float	Consigne de la commande à distance en cours
remote.control.w_target	4	228	float	Consigne de commande à distance à atteindre
remote.control.kv	1	232	byte	Vitesse de variation de la consigne
align	3	233	--	Interne
next	4	236	--	Interne

Exemple :

Pour lire la valeur de débit de la voie E1 la base des registres est REG(42101).

Le décalage de "flow\_q" est de 108 octets

Donc le décalage en registre est de  $108/2 = 54$

La valeur est stockée sur 4 octets donc les registres où lire "flow\_q" sont REG(42101+54) = REG(42055) et REG(42056).

REG(42155) = 0xA470

REG(42156) = 0x4541

Les données sont codées en "little endian" donc la valeur est 0x414570A4, soit 12.34 en virgule flottante 32bits

**STRUCT (sensor)**

Ce block de données contient les états les valeurs et les configurations des capteurs.

Nom	Taille [octets]	Décalage [octets]	Type	Description
sen	1	0	byte	Interne
ch	1	1	byte	Interne
def	1	2	byte	Interne
align	1	3	--	interne
fd	4	4	integer	Interne
flag	2	8	bits	bit0: erreur bit1: déconnecté bit2: la valeur de l'entrée a atteint sa limite haute bit3: la valeur de l'entrée a atteint sa limite basse bit4: valeur de mesure haute bit5: valeur de mesure basse bit6: mesure instable bit7: maintenance/calibrage nécessaire bit8: en cours de démarrage
kind	1	10	byte	Type de mesure: 0: Inactif 1: Chlore libre 2: Chlore actif 3: Chlore total 4: Chloramines 5: Chlorite 6: Dioxyde de chlore 7: H2O2 8: BCDMH 9: DBDMH

				10: Brome libre 11: Brome actif 12: Brome total 13: PAA 14: Ozone 15: Oxygène dissous 16: Nitrate 17: PHMB 18: Salinité 19: TDS 20: Turbidité 21: Conductivité 22: Température 23: Débit 24: pH 25: RedOx 26: Chloride 27: Amonia 28: Fluoride 29: ISE 30: Volume
unit	1	11	byte	Unité de mesure 0: Inactif 1: Aucune 2: Décade 3: pH 4: ppb 5: ppm 6: µg/l

				7: mg/l
				8: g/l
				9: %
				10: $\mu\text{S}/\text{cm}^2$
				11: $\text{mS}/\text{cm}^2$
				12: NTU
				13: FNU
				14: °K
				15: °C
				16: °F
				17: °R
				18: mA
				19: mV
				20: Hz
				21: str/min
				22: ms
				23: sec
				24: min
				25: h
				26: l
				27: m <sup>3</sup>
				28: l/min
				29: l/h
				30: m <sup>3</sup> /h
				31: imp/l
				32: imp/m <sup>3</sup>
				33: Ohm
				34: mOhm
				35: impulsion

transducer	1	12	byte	Type de convertisseur: 0: Aucun 1: 0...20mA 2: 4...20mA 3: pH -> 4...20mA 4: RedOx -> 4...20mA 5: pt100 -> 4...20mA 6: fluoride -> 4...20mA 7: fluoride (100) -> 4...20mA 8: ISOCAP pH -> 4...20mA 9: ISOCAP RedOk -> 4...20mA 10: UNISO P -> 4...20mA 11: UNISO R1 -> 4...20mA 12: UNISO R -> 4...20mA 13: UNISO B -> 4...20mA 14: 0...2000mV 15: 0...-2000mV 16: (potentiométrique) mV 17: impulsionnel 18: PT100 19: PT1000
align	1	13	--	Interne
delay	2	14	short	Temps de démarrage à réaliser (multiple de 0.5s)
tick	2	16	short	Compteur de temps de démarrage (multiple de 0.5s)
fault_tick	1	18	byte	Temps d'erreur (multiple de 0.5s)
align	1	19	--	Interne
min_value	4	20	float	Valeur basse de l'échelle de mesure
max_value	4	24	float	Valeur haute de l'échelle de mesure

cal_slope	4	28	float	Pente du capteur
cal_offset	4	32	float	Offset
std_slope	4	36	float	Pente avant calibrage
std_offset	4	40	float	Offset avant calibrage
z_dex	4	44	--	Interne
interface	4	48	float	Interface value
signal	4	52	float	Valeur du signal du capteur
std_value	4	56	float	Valeur de mesure avant calibrage
value	4	60	float	Valeur de mesure après calibrage
next	4	64	--	Interne

Exemple :

Pour lire l'unité de mesure du capteur connecté sur IN1, la base de registre est REG(42601).

Le décalage de "unit" est de 11 octets

Le décalage en registre est donc de  $11/2 = 5$

Le registre pour lire "unit" est  $REG(42601+5) = REG(42606)$

$REG(42606) = 0x0205$

Les données sont codées en "little endian" donc "unit" est sur l'octet de poids faible 0x05.

$0x05 = [ppm]$

**STRUCT (switch)**

Ce block de données contient les états, les valeurs et les configurations des contacts.

Nom	Taille [octets]	Décalage [octets]	Type	Description
sw	1	0	byte	Interne
ch	1	1	byte	Interne
align	2	2	--	Interne
fd	4	4	integer	Interne
flag	1	8	bits	bit0: contact opérationnel bit1: état de repos; NO=0; NF=1 bit2: interne bit3: état physique; ouvert=0; fermé=1 bit4: état retardé; ouvert=0; fermé=1 bit5: contact actif, en fonction de l'état de repos et de l'état retardé
align	1	9	--	Interne
delay	2	10	short	Délai d'anti-rebonds, ½ sec
tick	2	12	short	Temps d'anti-rebonds
align	2	14	--	Interne
next	4	16	--	Interne

Exemple :

Pour lire le temps d'anti-rebonds du contact K1 la base des registre est REG(42841).

Le décalage de "tick" est de 12 octets

Le décalage de registre est  $12/2 = 6$

Le registre où lire "tick" est  $REG(42841+6) = REG(42847)$

$REG(42847) = 0x0A00$

Les données sont codées en "little endian" donc l'ordre des octets est inversé.

tick = 0x000A

l'unité de "tick" est la ½ seconde donc le temps d'anti-rebonds est  $0xA \times 0.5 \text{ sec} = 5 \text{ sec}$ .

**STRUCT (relay)**

Ce block de données contient les états les valeurs et les configurations des relais.

Nom	Taille [octets]	Décalage [octets]	Type	Description
rel	1	0	byte	Interne
ch	1	1	byte	Interne
align	2	2	--	Interne
fd	4	4	integer	Interne
flag	1	8	bits	bit[0~2]: mode 0 = inactif 1 = régulation 2 = alarmes d'une voie 3 = alarmes de l'appareil 4 = état d'un contact 5 = état d'un relais 6 = timer bit3: état de repos NO=0; NC=1 bit4: état physique, ouvert=0; fermé=1 bit5: relais actif, actif=1 bit6: interne
align	3	9	--	Interne
param/device /switch/relay	4	12	--	Interne
alarm.delay	1	16	integer	Retard d'enclenchement du relais en cas d'alarme
alarm.tick	1	17	integer	Compteur de temps pour retarder l'alarme
alarm.pending	1	18	bits	Indicateurs d'alarmes en cours bit0: arrêt de la circulation bit1: seuil bas de la mesure bit2: seuil haut de la mesure

				bit3: capteur(s) hors échelle ou déconnectés bit4: capteur(s) hors échelle bit5: surdosage (temps max ou fond de cuve) bit6: capteurs en cours de démarrage
alarm.enable	1	19	bits	Indicateurs d'alarmes en fonctionnement bit0: arrêt de la circulation bit1: seuil bas de la mesure bit2: seuil haut de la mesure bit3: capteur(s) hors échelle ou déconnectés bit4: capteur(s) hors échelle bit5: surdosage (temps max ou fond de cuve) bit6: capteurs en cours de démarrage
dosage.u_min	4	20	float	
dosage.u_max	4	24	float	
dosage.u	4	28	float	Commande de dosage [1/1]
dosage.period	4	32	integer	Durée du cycle
dosage.min_width	2	36	integer	Temps mini de changement d'état du relais
align	2	38	--	Interne
dosage.compute_time	4	40	integer	Interne
dosage.tilt_time	4	44	integer	Interne
dosage.ref_time	4	48	integer	Interne
dosage.delay	4	52	integer	Interne
dosage.flag	1	56	bits	bits[0~1]: mode  0 = ON/OFF  1 = Largeur de cycle

				<p>2 = Impulsionnel</p> <p>bit2: sens de dosage, 0=montant; 1=descendant</p> <p>bit3: interne</p> <p>bit[4~5]: action</p> <p>0 = aucune</p> <p>1 = montante</p> <p>2 = descendante</p>
dosage.q_unit	1	57	bits	
align	2	58	--	Interne
timer.calendar.flag	1	60	bits	<p>bit0: traitement du timer en marche</p> <p>bit1: évènement du timer en cours</p> <p>bit2: traitement du timer en pause temporaire</p>
align	3	61	--	Interne
timer.calendar.event_list	4	64	--	Interne
timer.calendar.next	4	68	--	Interne
timer.action	1	72	bits	
align	3	73	--	Interne
delay_on	1	76	integer	
delay_off	1	77	integer	
tick_active	2	78	integer	
timer.handler	4	80	--	Interne
timer.proc	4	84	--	Interne
timer.delay	4	88	integer	
timer.trig_time	4	92	integer	
timer.next	4	96	--	Interne
next	4	100	--	Interne

Exemple:

Pour lire l'état du relais P1 la base de registres est REG(43001).

Le décalage de "flag" est de 8 octets

Le décalage en registre est de  $8/2 = 4$

Le registre pour lire "flag" est  $REG(43001+4) = REG(43005)$

$REG(43005) = 0x3500$

Les données sont codées en "little endian" donc "flag" se trouve dans l'octet de poids fort

$flag = 0x35 = 0b00110110$

Le bit qui indique l'état actif du relais et le bit 5 = 1, donc le relais est actif.

**STRUCT (iout)**

Ce block de données contient les états les valeurs et les configurations des sorties 0/4...20mA.

Nom	Taille [octets]	Décalage [octets]	Type	Description
out	1	0	integer	Interne
ch	1	1	integer	Interne
align	2	2	--	Interne
param / sensor	4	4	--	Interne
fd	4	8	integer	Interne
flag	2	12	bits	<p>bit0: échelle de sortie</p> <p>0: 4...20mA</p> <p>1: 0...20mA</p> <p>bit[1~2]: courant d'erreur</p> <p>0: 0mA</p> <p>1: 0 ou 4mA en fonction de l'échelle</p> <p>2: 2.6mA</p> <p>bit[3~4]: courant de pause</p> <p>0: aucun changement</p> <p>1: 0mA</p> <p>2: 0 ou 4mA en fonction de l'échelle</p> <p>3: 3.4mA</p> <p>bit[5~6]: courant hors échelle</p> <p>0: 21.7mA</p> <p>1: 20mA</p> <p>2: 20.8mA</p> <p>Bit[7~9]: source</p> <p>0: commande de dosage</p> <p>1: valeur de régulation</p> <p>2: valeur de mesure d'une voie</p>

				3: valeur de mesure d'un capteur 4: valeur d'une entrée bit10: pause en cours bit11: arrêt temporaire
align	2	14	--	interne
point_0_4mA	4	16	float	Valeur de mesure ou de régulation correspondant au point bas 0/4mA
point_20mA	4	20	float	Valeur de mesure ou de régulation correspondant au point haut 20mA
current	4	24	float	Courant de sortie [mA]
next	4	28	--	Interne

Exemple :

Pour lire le courant de la sortie IOUT2, la base de registre est REG(42921).

Le décalage de "current" est de 28 octets

Le décalage en registre est  $28/2 = 14$

Le registre pour lire "current" est  $REG(42921+14) = REG(42935)$  and  $REG(42936)$

$REG(42935) = 0xA470$

$REG(42936) = 0x4541$

Les données sont codées en "little endian" donc la valeur est  $0x414570A4$ , 12.34mA en virgule flottante 32bit.

### **STRUCT (calendar)**

Ce block de données contient les états les valeurs et les configurations des calendrier de gestion des évènement timer.

Nom	Taille [octets]	Décalage [octets]	Type	Description
calendar.flag	1	0	bits	bit0: timer actif bit1: évènement(s) en cours bit2: arrêt temporaire
align	3	1	--	Interne
calendar.event_list	4	4	--	Interne
calendar.next	4	8	--	Interne

## STRUCT (calendar\_event)

Nom	Taille [octets]	Décalage [octets]	Type	Description
name	8	0	string	
ev	1	8		
align	3	9	--	Interne
fd	4	12	integer	Interne
flag	2	16	bits	Jours actifs et indicateurs d'état bit0: lundi bit1: mardi bit2: mercredi bit3: jeudi bit4: vendredi bit5: samedi bit6: dimanche bit7: interne bit8: timer actif bit9: évènement en cours
align	2	18	--	Interne
date_start	4	20	integer	Date et heure du début de l'évènement
date_end	4	24	integer	Date et heure de fin l'évènement
interval	1	28	integer	Nombre de semaines entre chaque répétition de l'évènement
align	3	29	--	Interne
next	4	32	--	Interne

Contact fabricant

**BWT**

103, rue Charles Michels F-93206 Saint Denis Cedex

*bwt.com*