

CP200X

PROTOCOLO DE COMUNICACIONES



1.- PROTOCOLO DE COMUNICACIONES.

El convertidor CP200x está provisto de una línea de comunicaciones para transmitir todos los valores medidos a un equipo central. Se dispone de dos versiones distintas, RS232 y RS485. La primera se usa en comunicaciones punto a punto, y la segunda en multipunto. Con la opción RS485 se pueden conectar hasta 32 medidores en la misma línea, con una longitud máxima de 1200m.

Las salidas de comunicaciones están aisladas de los circuitos de medida mediante los transformadores de acoplo, de forma que el bus se encuentra flotante, lo que evita la formación de bucles de tierra.

Aunque los niveles de señal son diferentes, se usa el mismo protocolo en ambas. Este protocolo cumple la norma de MODBUS/JBUS tipo RTU. Se trata de un protocolo de fácil implementación, muy extendido en autómatas y PLCs, del tipo maestro/esclavo, sin mensajes espontáneos. La transmisión se realiza a 9600 Bps, con 8 bits, sin paridad, y con un bit de stop. El equipo permite la selección de otras velocidades de transmisión, de 300 a 19200 Bps.

Cada transacción consiste en el envío de una trama de petición generada por el maestro, y una trama de respuesta, generada por el esclavo. En el caso de una escritura, el esclavo devuelve un acknowledge. Si el mensaje no llega correctamente, el esclavo no devuelve nada.

Los comandos aceptados son:

03H	Read Holding Registers (3xxxx).
04H	Read Input Registers (4xxxx).
06H	Preset Single Register (6xxxx)
10H	Preset Multiple Registers



1.1. – DESCRIPCIÓN DE LAS TRAMAS.

1.1.1. - LECTURA (Comandos 03h ó 04H)

Consta de:

- | | |
|------------------------------|-----------------|
| - Identidad | Un byte |
| - Comando | 04H o 03H |
| - Dirección de datos | Dos bytes: H, L |
| - Número de registros a leer | Dos bytes: H, L |
| - CRC | Dos bytes: L, H |

1.1.2. – ESCRITURA.

Preset Single Register.

- | | |
|------------------------|-----------------|
| - Identidad | Un byte |
| - Comando | 06H |
| - Dirección de datos | Dos bytes: H, L |
| - Valor de la variable | Dos bytes |
| - CRC | Dos bytes: L, H |

Preset Multiple Registers

- | | |
|---------------------------------|-----------------|
| - Identidad | Un byte |
| - Comando | 10H |
| - Dirección de datos | Dos bytes: H, L |
| - Número de palabras a escribir | Dos bytes: H, L |
| - Número de bytes a escribir | Un byte |
| - Valor de la variable | Cuatro bytes |
| . | . |
| . | . |
| - Valor de la variable | Cuatro bytes |
| - CRC | Dos bytes: L, H |

1.1.3. – RESPUESTA.

- | | |
|-------------------------------|-----------------|
| - Consta de: | |
| - Identidad | Un byte |
| - Comando (el mismo recibido) | 04H, o 03H |
| - Número de bytes enviados | Un byte |
| - Valor de la variable | Cuatro bytes |
| . | . |
| . | . |
| . | . |
| - Valor de la variable | Cuatro bytes |
| - CRC | Dos bytes: L, H |

1.1.4. – TRAMA DE ACKNOWLEDGE.

- | | |
|---|-----------------|
| - Consta de: | |
| - Identidad | Un byte |
| - Comando (el mismo recibido) | 10H |
| - Dirección (la misma recibida) | Dos bytes: H, L |
| - Número de palabras escritas (las mismas) | Dos bytes: H, L |
| - CRC | Dos bytes: L, H |



2. – MAPA DE DIRECCIONES.

El direccionamiento de los registros es reubicable, mediante un registro base situado en la posición absoluta 0000H, llamado BASE_ADD. La dirección real se forma sumando el valor contenido en el registro base y el valor de offset señalado en las tablas. Por ejemplo, la dirección del registro de Identidad, (offset = 200H), será [BASE_ADD] + 200H. Esto se aplica igualmente al propio registro base, al que se puede acceder en la posición absoluta 0H, o a la relativa [BAS_ADD] + 0H. El registro base puede tener cualquier valor comprendido entre 0 y 0F000H , ambos inclusive. De esta forma el mapa puede ser colocado en cualquier zona de memoria, adaptándose así a los requisitos de algunos autómatas, que necesitan una determinada posición para una determinada clase de variables.

Tanto el comando 03H, como el 04H son válidos para la lectura, así como el 06H, y el 10H para la escritura.

Este tipo de direccionamiento cumple con los requisitos del protocolo MODBUS totalmente, que considera cada registro como una palabra de 16 bits. La diferencia entre dos registros consecutivos es por tanto dos bytes, y la diferencia entre dos variables de tipo flotante es cuatro bytes.

Las variables usadas en el CP200x se indican en las tablas siguientes.

2.1. – VARIABLES DE CALIBRACIÓN.

OFFSET (DEC)	VARIABLE (NOMBRE)	TIPO	R/W	US.	COD.	BLOCK R/W
400	VR100	LONG	R	NO	SI	NO
402	VS100	LONG	R	NO	SI	NO
404	VT100	LONG	R	NO	SI	NO
406	PR100	LONG	R	NO	SI	NO
408	PS100	LONG	R	NO	SI	NO
410	PT100	LONG	R	NO	SI	NO
412	QR100	LONG	R	NO	SI	NO
414	QS100	LONG	R	NO	SI	NO
416	QT100	LONG	R	NO	SI	NO
418	IR100	LONG	R	NO	SI	NO
420	IS100	LONG	R	NO	SI	NO
422	IT100	LONG	R	NO	SI	NO
424	IR0	WORD	R	NO	SI	NO
425	VR0	WORD	R	NO	SI	NO
426	IS0	WORD	R	NO	SI	NO
427	VS0	WORD	R	NO	SI	NO
428	IT0	WORD	R	NO	SI	NO
429	VT0	WORD	R	NO	SI	NO

Estas variables se usan solamente para la calibración del equipo. No deben ser modificadas por el usuario, aunque pueden ser leídas.



2.2. – VARIABLES DE CONFIGURACIÓN.

OFFSET (DEC)	VARIABLE (NOMBRE)	TIPO	R/W	US.	COD.	BLOCK R/W
0	BASE_ADD	WORD	R/W	SI	NO	NO
1	ESCALAV	IEEE	R/W	SI	NO	NO
5	ESCALAI	IEEE	R/W	SI	NO	NO
9	REF_ENER	IEEE	R/W	SI	NO	NO
19	OUT_LOW0	IEEE	R/W	SI	NO	NO
21	OUT_HIGH0	IEEE	R/W	SI	NO	NO
23	VAR_LOW0	IEEE	R/W	SI	NO	NO
25	VAR_HIGH0	IEEE	R/W	SI	NO	NO
31	OUT_LOW1	IEEE	R/W	SI	NO	NO
33	OUT_HIGH1	IEEE	R/W	SI	NO	NO
35	VAR_LOW1	IEEE	R/W	SI	NO	NO
37	VAR_HIGH1	IEEE	R/W	SI	NO	NO
75	FREC_NOM	IEEE	R/W	SI	NO	NO
27	OUT_MED0	IEEE	R/W	SI	NO	NO
29	VAR_MED0	IEEE	R/W	SI	NO	NO
39	OUT_MED1	IEEE	R/W	SI	NO	NO
41	VAR_MED1	IEEE	R/W	SI	NO	NO
212	PROG_VELOC	IEEE	R/W	SI	NO	NO

Estas variables se usan para la configuración del equipo.

BASE_ADD: Es la dirección base para formar la dirección absoluta de cada variable.

ESCALAV: Valor de la tensión de primario. Si no se emplean transformadores de tensión, este valor debe coincidir con el valor de ajuste de secundario.

ESCALAI: Valor de la corriente nominal de primario. Si no se usan transformadores de intensidad, este valor debe coincidir con el valor de secundario (1 ó 5 A).

REF_ENER: Es el valor del impulso de energía, que proporcionan las salidas digitales. Afecta igualmente a la energía activa y a la reactiva. Se puede programar cualquier valor, pero es recomendable usar un múltiplo de diez que sea coherente con la programación general del equipo, de forma que no se produzcan demasiados impulsos o demasiado pocos en la salida digital. Por ejemplo, para un equipo de 200 kW de potencia nominal, un valor de 1 kWh por impulso daría 200 impulsos por hora, que es adecuado. Un valor de 10 kWh sería asimismo adecuado, pero un valor de 0,1 kWh produciría demasiados impulsos por hora.

FREC_NOM: Aunque el equipo es capaz de funcionar a cualquier frecuencia comprendida entre 45 y 65 Hz, se necesita definir la frecuencia a la cual debe actuar la alarma si se selecciona, y la salida analógica asociada.

PROG_VELOC define la velocidad de la línea serie de comunicaciones, según la tabla.



PROG_VELOC	BAUD RATE
0	9600
1	300
2	600
3	1200
4	2400
5	4800
6	19200

OUT_LOWx, OUT_HIGHx : Corresponden al los valores inferior y superior de la salida, respectivamente. Deben ser programados de acuerdo con la salida elegida, en formato IEEE, bien en voltios para salida de tensión, bien en mA para la corriente, y pueden ir desde el margen inferior de la salida asociada, hasta el superior. Por ejemplo, si la salida es 5 mA, el valor inferior puede ser desde – 5 mA hasta 5 mA. Estos valores son los denominados Y0 e Y2 en los diagramas de salida.

VAR_LOWx, VAR_HIGHx : Son los valores inferior y superior de la variable medida asociados a la salida. Corresponden a los valores X0 y X2 de los diagramas de salida. Admiten cualquier valor comprendido entre los márgenes nominales de la variable, pero hay que tener en cuenta que un margen muy estrecho disminuye la precisión de la lectura. Se deben programar en porcentaje del valor nominal de la variable. Por ejemplo, para una entrada de tensión, se puede aceptar un margen de 40 a 120 %. Para una variable bidireccional sería igualmente aceptable de –40 a 100%.

OUT_MEDx, son los puntos intermedios de la característica quebrada. Se programan de igual manera que OUT_LOWx. Su valor debe estar comprendido entre ambos extremos.

VAR_MEDx son los valores de la variable en la característica quebrada. Se programan de igual forma que VAR_LOWx, y deben asimismo estar comprendidos entre ambos extremos.

La selección del modo de salida lineal –quebrada se hace mediante el registro CONF_OUT.

2.3.- VARIABLES DE CONFIGURACIÓN DE SALIDA.

OFFSET (DEC)	VARIABLE (NOMBRE)	TIPO	R/W	US.	COD.	BLOCK R/W
217	CONF_OUT	WORD	R/W	SI	NO	NO
224						
225						
218	CONF_DAC0	BYTE	R/W	SI	NO	NO
219	CONF_DAC1	BYTE	R/W	SI	NO	NO
220						
213						
214						
222	CONF_HARD	BYTE	R/W	SI	NO	NO
223	CONF_MANUAL	BYTE	R/W	SI	NO	NO

Estas variables se usan para configurar los modos de funcionamiento de las salidas.



CONF_OUT: Es una variable de dos bytes, con el significado siguiente.

Bit 7	Volt/mA salida 2	1 : tensión
Bit 6	Ajuste DAC 2	1 : Permite ajuste
Bit 5	Disable salida 2	1: Pone la salida a 0
Bit 4	Salida quebrada 2	1: Quebrada
Bit 3	Volt/ mA salida 1	1: tensión
Bit 2	Ajuste DAC 1	1 : Permite ajuste
Bit 1	Disable salida 1	1: Pone la salida a 0
Bit 0	Salida quebrada 1	1: Quebrada

Los bits Volt/mA comunican al microprocesador el tipo de salida, si es en tensión o corriente. Es necesario para poder calcular los márgenes de escala, dentro del alcance seleccionado.

Los bits de ajuste del DAC controlan el proceso de calibración. No se deben modificar.

Los bits Disable pasan la salida a cero, independientemente de su valor.

Los bits Salida quebrada determinan si la salida va a ser en forma lineal, de dos puntos, o quebrada, de tres.

Byte CONF_DACx:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	-	X	X	X	X	X

Cada byte define la variable que va a ser tomada para la salida, de acuerdo con la tabla:

Valor en CONF_DAC (XX)	Variable
0H	Vfr
1H	Vfs
2H	Vft
3H	Vrs
4H	Vst
5H	Vtr
6H	Pfr
7H	Pfs
8H	Pft
9H	Qfr
AH	Qfs
BH	Qft
CH	Ifr
DH	Ifs
EH	Ift
FH	Sfr
10H	Sfs
11H	Sft
12H	Cosr
13H	Coss



14H	Cost
15H	Prst
16H	Qrst
17H	Srst
18H	Cos
19H	Sinr
1AH	Sins
1BH	Sint
1CH	Sin
1DH	Freq

Byte CONF_MANUAL:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	-	-	-	-	X	X

Estos bits definen el modo de salida. Si están a 0, la salida corresponde a la lectura de la variable definida en CONF_DAC. Si están a 1, la salida corresponde al valor definido en la palabra AN_OVERx. Esto permite simular cualquier valor de la lectura con el propósito de comprobar el bucle.

CONF_HARD:

Este byte permite definir el margen de salida, de acuerdo con la tabla.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	-	-	O21	O20	O11	O10

Ox1	Ox0	Salida
1	1	-
1	0	1V ó 1 mA
0	1	5 V ó 5 mA
0	0	10 V ó 20 mA

2.4. – VARIABLES DE LECTURA.

Estas variables se pueden leer en cualquier momento. La variable SEQUENCE indica la correcta secuencia de las tres tensiones cuando indica cero. Cualquier valor distinto indica fallo de secuencia. La comprobación se realiza solamente después del arranque, y se visualiza además en el led TE, que parpadea durante tres segundos en caso de fallo.

OFFSET (DEC)	VARIABLE (NOMBRE)	TIPO	R/W	US.	COD.	BLOCK R
120	VFR	IEEE	R	SI	NO	SI
122	VFS	IEEE	R	SI	NO	SI
124	VFT	IEEE	R	SI	NO	SI
126	VRS	IEEE	R	SI	NO	SI
128	VST	IEEE	R	SI	NO	SI
130	VTR	IEEE	R	SI	NO	SI
132	PFR	IEEE	R	SI	NO	SI
134	PFS	IEEE	R	SI	NO	SI
136	PFT	IEEE	R	SI	NO	SI



138	QFR	IEEE	R	SI	NO	SI
140	QFS	IEEE	R	SI	NO	SI
142	QFT	IEEE	R	SI	NO	SI
144	IFR	IEEE	R	SI	NO	SI
146	IFS	IEEE	R	SI	NO	SI
148	IFT	IEEE	R	SI	NO	SI
150	SFR	IEEE	R	SI	NO	SI
152	SFS	IEEE	R	SI	NO	SI
154	SFT	IEEE	R	SI	NO	SI
156	COSR	IEEE	R	SI	NO	SI
158	COSS	IEEE	R	SI	NO	SI
160	COST	IEEE	R	SI	NO	SI
162	PRST	IEEE	R	SI	NO	SI
164	QRST	IEEE	R	SI	NO	SI
166	SRST	IEEE	R	SI	NO	SI
168	COSENO	IEEE	R	SI	NO	SI
170	FREC_RED	IEEE	R	SI	NO	SI
172	SENR	IEEE	R	SI	NO	SI
174	SENS	IEEE	R	SI	NO	SI
176	SENT	IEEE	R	SI	NO	SI
178	SENO	IEEE	R	SI	NO	SI
216	SEQUENCE	BYTE	R	SI	NO	NO

2.5. – VARIABLES DE IDENTIFICACIÓN.

OFFSET (DEC)	VARIABLE (NOMBRE)	TIPO	R/W	USER	CODE	BLOCK R/W
200	SER.NUMBER	STRING 10	R	NO	YES	NO
205	ID	BYTE	R/W	SI	NO	NO
206	TIPO	STRING 6	R	SI	NO	NO

SER. NUMBER es una variable de diez caracteres, que define el número de serie del equipo al ser fabricado.

ID representa la identidad del equipo conectado en red. Es un byte, y se puede cambiar a cualquier valor entre 0 y 255. No deben usarse los valores 0, 199, y 255. La identidad 199 es un valor genérico, y cualquier comando dirigido a ella será aceptado por todas las unidades conectadas.

TIPO indica la versión SW del equipo.

2.6. –VARIABLES DE CONTROL.

OFFSET (DEC)	VARIABLE (NOMBRE)	TIPO	R/W	US.	COD.	BLOCK R/W
454	CODE_ACC	BYTE	W	SI	NO	NO
211	TIPO_PROT	BYTE	R/W	SI	NO	NO
212	PROG_VEL	BYTE	R/W	SI	NO	NO
213	ESCALAP	IEEE	R	SI	NO	NO



TIPO_PROT:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	-	-	-	-	-	X

Cuando vale 1, significa protocolo MODBUS. Si =0, significa JBUS.

La diferencia estriba únicamente en el orden en que los datos se envían por la línea serie.

En JBUS, los datos se envían:

S+EXP Mantisa H Mantisa M Mantisa L

En MODBUS:

Mantisa M Mantisa L S+EXP Mantisa H

ESCALAP: Expresa la potencia nominal. Su valor permite comprobar que los valores programados de tensión y corriente son correctos. Corresponde a:

$$ESCALAV * ESCALAI * \sqrt{3}$$

PROG_VEL define la velocidad del puerto de comunicaciones, de acuerdo con la tabla:

Valor	Vel (Bps)
0	9600
1	300
2	600
3	1200
4	2400
5	4800
6	19200



3. - COMANDOS.

3.1. – COMANDOS DE LECTURA.

Un comando de lectura consta de

ID---tipo de comando---Dirección---Número de palabras---CRCL---CRCH

ID es el byte de identidad. El tipo de comando es 03H, o 04H. La dirección es la primera posición donde se encuentran los datos. Número de palabras es un word, de 16 bits. Si solamente se desea leer un byte, el número de palabras también se debe poner a 1. Todos los datos pueden leerse por medio de comandos individuales, pero se pueden emplear también comandos múltiples para las variables más comúnmente leídas, con las limitaciones siguientes:

Todos las variables deben ser del mismo formato: de coma flotante, o long.
La longitud total del bloque no debe exceder de 12 variables.

3.2. – EJEMPLOS DE COMANDOS DE LECTURA.

Número de serie.

P.:	0C7H	04H	04H	0B0H	00H	05H	21H	B8H		
	ID	CMD	DIRECCIÓN		Nº WORDS		CRCL	CRCH		
R.:	0C7H	04H	0AH		53H	41H	43H	49H	33H	31H
	ID	CMD	Nº BYTES		S	A	C	I	3	1
	30H	30H	33H	46H	68H	4CH				
	0	0	3	F	CRCL	CRCH				

Escala de tensión.

P.:	0C7H	04H	3H	0E9H	00H	02H	0B1H	1DH		
	ID	CMD	DIRECCIÓN		No.WORDS		CRCL	CRCH		
R.:	0C7H	04H	04H		00H	00H	42H	0DCH		
	ID	CMD	No.BYTES		----- 110V -----					
	06CH	0B1H								
	CRCL	CRCH								



Lectura múltiple.

P.:	01H	04H	30H	0CCH	00H	18H	3FH	3FH
	ID	CMD	DIRECCIÓN		Nº WORDS		CRCL	CRCH
R.:	01H	04H	30H	43H	60H	0B5H	07H	
	ID	CMD	Nº BYTES		-----	Vfr	-----	
	43H	60H	0DAH	00H	43H	063H	0C6H	00H
	-----	Vfs	-----		-----	Vft	-----	
	43H	0C5H	37H	00H	43H	0C5H	2EH	00H
	-----	Vrs	-----		-----	Vst	-----	
	43H	0C5	1FH	00H	3EH	89H	6CH	00H
	-----	Vtr	-----		-----	Pfr	-----	
	00H	00H	00H	00H	00H	00H	00H	00H
	-----	Pfs	-----		-----	Pft	-----	
	00H	00H	00H	00H	00H	00H	00H	00H
	-----	Qfr	-----		-----	Qfs	-----	
	00H	00H	00H	00H	0EDH	0ACH		
	-----	Qft	-----		CRCL	CRCH		



3.3. – COMANDOS DE ESCRITURA.

Los comandos de escritura constan de

ID--- Tipo de Comando--- Dirección--- Número de palabras--- Número de bytes---CRCL---CRCH

ID es la identidad. Tipo de comando es 06H, o 10H. Número de palabras es de 16 bits. Número de bytes es un byte.

No se admiten comandos múltiples. Cada variable debe ser escrita por medio de un comando individual.

3.4. – EJEMPLOS DE COMANDOS DE ESCRITURA..

Escritura de ESC_I (1000 + 5 = 1005D, 03DBH) a 500 A.

P.:	0C7H	10H	3H	DBH	00H	02H	04H
	ID	CMD	DIRECCIÓN	Nº WORDS	Nº BYTES		
	00H	00H	043H	0FAH	07BH	076H	
	-----	VALOR	-----	CRCL	CRCH		

R.: (Acknowledge)

0C7H	10H	3H	DBH	00H	02H	0C0H	0DFH
ID	CMD	DIRECCIÓN	Nº WORDS	CRCL	CRCH		



4. FORMATO DE LOS DATOS.

Se usan los siguientes formatos de datos.

ASCII: Caracteres por ej. :nº de serie. Se envían en el orden especificado en las tablas.

BYTE: Ocho bits. Para estado o control. Se envían como words.

WORD: Dos bytes. Se envían MSB-LSB.

LONG: Cuatro bytes. Se envían MSB-____-____-LSB. (Ver Anexo 3).

IEEE: Cuatro bytes. Se envían S + EXP – Mh – Mm – ML . (Ver Anexo 3).



ANEXO 1. ALGORITMO CRC TIPO "CRC16".

1. – POLINOMIO GENERADOR.

El polinomio usado es:

$$X^{16} + X^{15} + X^2 + 1 = 18005H$$

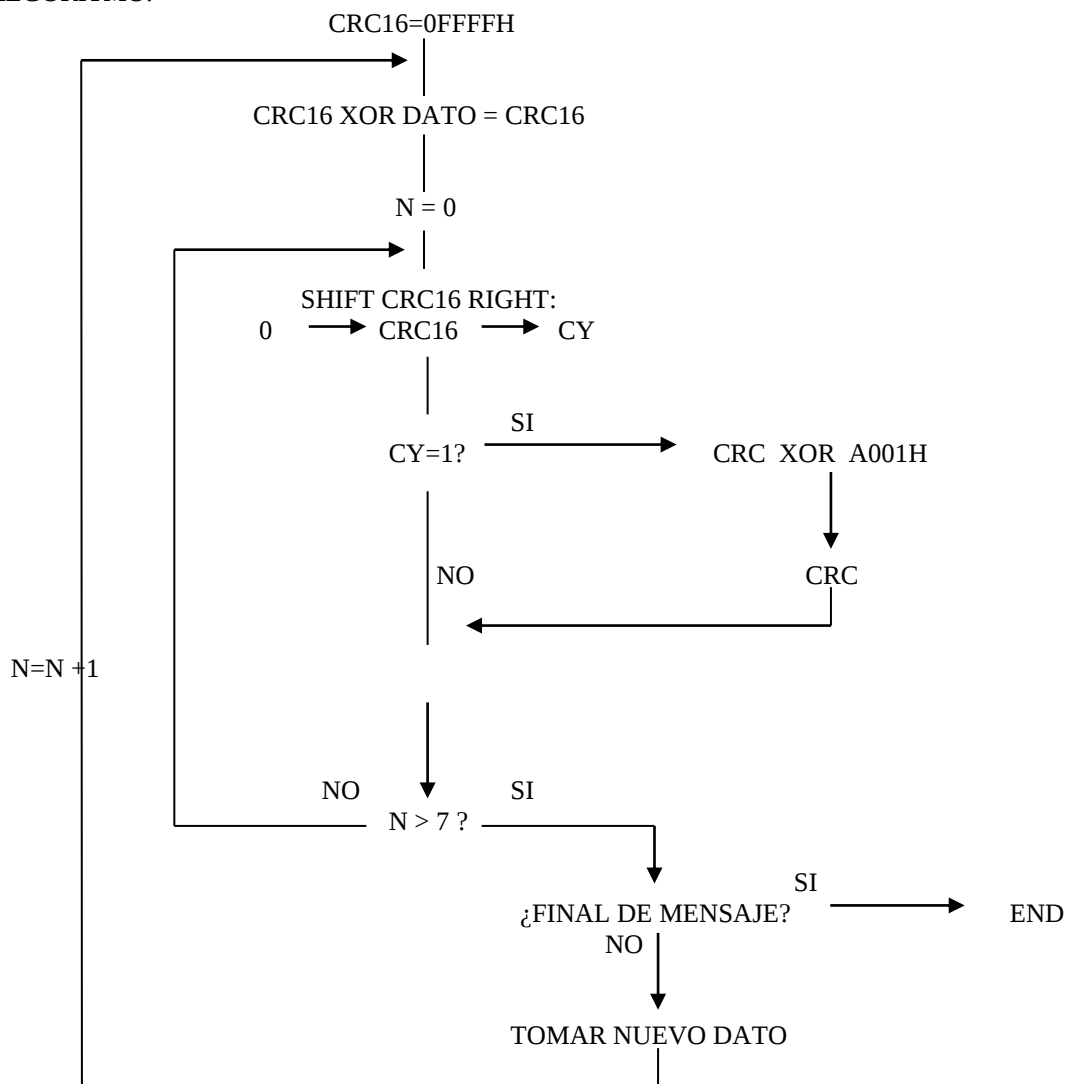
$$X^{16} + X^{15} + X^2 + 1 = 18005H$$

Para calcular el CRC, se invierte el polinomio, omitiendo el bit menos significativo.

POLINOMIO CRC16 : 1 1000 0000 0000 0101 = 18005H

POLINOMIO DE TRABAJO : 1010 0000 0000 0001 = A001H.

2. - ALGORITMO.



"DATO" es el byte recibido o que se desea transmitir.

"CRC16" es una palabra de 16 bits. El resultado de la operación se deja asimismo en CRC16. Si el crc recibido se incluye en el algoritmo, el resultado final será cero.



ANEXO 2. NOTACIÓN USADA. (IEEE 754).

Los números en coma flotante utilizan la notación IEEE754. Como la precisión que se obtiene con este tipo de notación está muy por encima de la precisión del equipo, el byte correspondiente a la mantisa baja es siempre cero. Esto puede producir en ciertos casos una pequeña discrepancia entre los datos escritos desde el ordenador principal y los datos que se leen. Por ejemplo, una cifra tal como 220.000 se puede leer del equipo como 219.987.

BYTE1	BYTE2	BYTE3	BYTE4
SIGNO+	MANTISA	MANTISA	MANTISA
EXPONENT	ALTA	MEDIA	BAJA

SIGNO: signo del número.

0H significa un número positivo.

1H significa un número negativo.

MANTISA: FRACCIÓN 0, XXX XXXX XXXX XXXX XXXX XXXX

M1	M23

EXPONENTE: Es el exponente del número con offset 127.

0: 127. (7FH)

1: 128. (80H)

-1: 126. (7EH)

Para calcular el valor:

VALOR: $(-1)^S \cdot 2^{EXP-127} \cdot (1 + \text{FRACCIÓN})$

FRACCIÓN: $\sum_{i=1}^{i=23} 2^{-i} \cdot M(i)$

La configuración de bytes es como sigue:

BYTE1:

7	6	5	4	3	2	1	0
SIGNO -----		EXPONENTE				-----	
E7	E6	E5	E4	E3	E2	E1	

BYTE2:

7	6	5	4	3	2	1	0
EXP -----		MANTISA ALTA -----					
E0	M1	M2	M3	M4	M5	M6	M7

BYTE3:

7	6	5	4	3	2	1	0
----- MANTISA				MEDIA -----			
M8	M9	M10	M11	M12	M13	M14	M15

BYTE4: (Siempre nulo).

7	6	5	4	3	2	1	0
----- MANTISA BAJA -----							
M16	M17	M18	M19	M20	M21	M22	M23



ANEXO 3. FORMATO DE TRANSMISIÓN DE DATOS .

Los datos en formato IEEE se envían:

1. BYTE SIGNO + EXPONENTE
2. BYTE MANTISA ALTA
3. BYTE MANTISA MEDIA
4. BYTE MANTISA BAJA (SIEMPRE CERO)

Este modo de transmisión es el que denominamos MODO JBUS.

En algunas aplicaciones se requiere un orden inverso:

1. BYTE MANTISA MEDIA
2. BYTE MANTISA BAJA (SIEMPRE CERO)
3. BYTE SIGNO + EXPONENTE
4. BYTE MANTISA ALTA

Este modo de transmisión es el que denominamos MODBUS.

Por defecto, las unidades se programan en modo JBUS.

Esta diferencia se aplica igualmente a los números en formato LONG:

MSB, msb, LSB, lsb corresponde a formato JBUS.

LSB, lsb, MSB, msb corresponde a formato MODBUS.

Para seleccionar el formato, se usa la variable de un byte TIPO_PROT.

00H selecciona el modo JBUS.

01H selecciona el modo MODBUS



S.A.C.I.
C/ La Granja,84 ALCOBENDAS
28108 Madrid
Tfno.: (34)91 5190245
Fax.: (34) 91 4169646



CP200xMB_E.doc

ED./FECHA	PUNTO	MOTIVO DEL CAMBIO	RESPONS.	Aprobado
1.1 11/06		DOCUMENTO ORIGINAL (procede de CP30MB_E.DOC)	F. OLMOS	

RESP. DEP. TÉCNICO

RESP. DEP. COMERCIAL

SAC-F-35



CP200x Protocolo de comunicaciones

ver 1.1 Nov - 2006
CP200xMB_E.doc
19

CP300X

PROTOCOLO DE COMUNICACIONES



1.- PROTOCOLO DE COMUNICACIONES.

El convertidor CP3000 dispone de una línea de comunicaciones para transmitir todos los valores medidos a un equipo central. Se dispone de dos versiones distintas, RS232 y RS485. La primera se usa en comunicaciones punto a punto, y la segunda en multipunto. Con la opción RS485 se pueden conectar, según aconseja la norma, hasta 32 medidores en la misma línea, con una longitud máxima de 1200m.

Existe una versión del convertidor que dispone de las dos líneas de comunicaciones a la vez. Las dos líneas utilizan el mismo protocolo, de forma que se aceptan los comandos indistintamente por ambas.

Las salidas de comunicaciones están aisladas de los circuitos del micro mediante acopladores ópticos, de forma que el bus se encuentra flotante, lo que evita la formación de bucles de tierra.

Aunque los niveles de señal son diferentes para RS232 y RS485, se usa el mismo protocolo en ambas, que cumple la norma de MODBUS/JBUS tipo RTU. Se trata de un protocolo de fácil implementación, muy extendido en autómatas y PLCs, del tipo maestro/esclavo, sin mensajes espontáneos. La transmisión se realiza a 9600 Bps, con 8 bits, sin paridad, y un bit de stop. El equipo permite la selección de otras velocidades de transmisión, de 300 a 19200 Bps.

Cada transacción consiste en el envío de una trama de petición generada por el maestro, y una trama de respuesta, generada por el esclavo. En el caso de una escritura, el esclavo devuelve un acknowledge. Si el mensaje no llega correctamente, el esclavo no devuelve nada.

Los comandos aceptados son:

03H	Read Holding Registers (3xxxx).
04H	Read Input Registers (4xxxx).
06H	Preset Single Register (6xxxx)
10H	Preset Multiple Registers



1.1. – DESCRIPCIÓN DE LAS TRAMAS.

1.1.1. - LECTURA (Comandos 03h ó 04H)

Consta de:

- | | |
|------------------------------|-----------------|
| - Identidad | Un byte |
| - Comando | 04H o 03H |
| - Dirección de datos | Dos bytes: H, L |
| - Número de registros a leer | Dos bytes: H, L |
| - CRC | Dos bytes: L, H |

1.1.2. – ESCRITURA.

Preset Single Register.

- | | |
|------------------------|-----------------|
| - Identidad | Un byte |
| - Comando | 06H |
| - Dirección de datos | Dos bytes: H, L |
| - Valor de la variable | Dos bytes |
| - CRC | Dos bytes: L, H |

Preset Multiple Registers

- | | |
|---------------------------------|-----------------|
| - Identidad | Un byte |
| - Comando | 10H |
| - Dirección de datos | Dos bytes: H, L |
| - Número de palabras a escribir | Dos bytes: H, L |
| - Número de bytes a escribir | Un byte |
| - Valor de la variable | Cuatro bytes |
| . | . |
| . | . |
| - Valor de la variable | Cuatro bytes |
| - CRC | Dos bytes: L, H |

1.1.3. – RESPUESTA.

Consta de:

- | | |
|-------------------------------|-----------------|
| - Identidad | Un byte |
| - Comando (el mismo recibido) | 04H, o 03H |
| - Número de bytes enviados | Un byte |
| - Valor de la variable | Cuatro bytes |
| . | . |
| . | . |
| . | . |
| - Valor de la variable | Cuatro bytes |
| - CRC | Dos bytes: L, H |

1.1.4. – TRAMA DE ACKNOWLEDGE.

Consta de:

- | | |
|---|-----------------|
| - Identidad | Un byte |
| - Comando (el mismo recibido) | 10H, 06H |
| - Dirección (la misma recibida) | Dos bytes: H, L |
| - Número de palabras escritas (las mismas) | Dos bytes: H, L |
| - CRC | Dos bytes: L, H |



1.2. – MAPA DE DIRECCIONES.

El direccionamiento de los registros es reubicable, mediante un registro base situado en la posición absoluta 0000H, llamado BASE_ADD. La dirección real se forma sumando el valor contenido en el registro base y el valor de offset señalado en las tablas. Por ejemplo, la dirección del registro de Identidad, (offset = 200H), será [BASE_ADD] + 200H. Esto se aplica igualmente al propio registro base, al que se puede acceder en la posición absoluta 0H, o a la relativa [BAS_ADD] + 0H. El registro base puede tener cualquier valor comprendido entre 0 y 0F000H , ambos inclusive. De esta forma el mapa puede ser colocado en cualquier zona de memoria, adaptándose así a los requisitos de algunos autómatas, que necesitan una determinada posición para una determinada clase de variables.

Tanto el comando 03H, como el 04H son válidos para la lectura, así como el 06H, y el 10H para la escritura.

Este tipo de direccionamiento cumple con los requisitos del protocolo MODBUS totalmente, que considera cada registro como una palabra de 16 bits. La diferencia entre dos registros consecutivos es por tanto dos bytes, y la diferencia entre dos variables de tipo flotante es cuatro bytes.

Las variables usadas en el CP300x se indican en las tablas siguientes.

1.2.1. – VARIABLES DE CALIBRACIÓN.

OFFSET (DEC)	VARIABLE (NOMBRE)	TIPO	R/W	US.	COD.	BLOCK R/W
400	VR100	LONG	R	NO	SI	NO
402	VS100	LONG	R	NO	SI	NO
404	VT100	LONG	R	NO	SI	NO
406	PR100	LONG	R	NO	SI	NO
408	PS100	LONG	R	NO	SI	NO
410	PT100	LONG	R	NO	SI	NO
412	QR100	LONG	R	NO	SI	NO
414	QS100	LONG	R	NO	SI	NO
416	QT100	LONG	R	NO	SI	NO
418	IR100	LONG	R	NO	SI	NO
420	IS100	LONG	R	NO	SI	NO
422	IT100	LONG	R	NO	SI	NO
424	IR0	WORD	R	NO	SI	NO
425	VR0	WORD	R	NO	SI	NO
426	IS0	WORD	R	NO	SI	NO
427	VS0	WORD	R	NO	SI	NO
428	IT0	WORD	R	NO	SI	NO
429	VT0	WORD	R	NO	SI	NO

Estas variables se usan solamente para la calibración del equipo. No deben ser modificadas por el usuario, aunque pueden ser leídas.



1.2.2. – VARIABLES DE CONFIGURACIÓN.

OFFSET (DEC)	VARIABLE (NOMBRE)	TIPO	R/W	US.	COD.	BLOCK R/W
0	BASE_ADD	WORD	R/W	SI	NO	NO
1	ESCALAV	IEEE	R/W	SI	NO	NO
5	ESCALAI	IEEE	R/W	SI	NO	NO
9	REF_ENER	IEEE	R/W	SI	NO	NO
19	OUT_LOW0	IEEE	R/W	SI	NO	NO
21	OUT_HIGH0	IEEE	R/W	SI	NO	NO
23	VAR_LOW0	IEEE	R/W	SI	NO	NO
25	VAR_HIGH0	IEEE	R/W	SI	NO	NO
31	OUT_LOW1	IEEE	R/W	SI	NO	NO
33	OUT_HIGH1	IEEE	R/W	SI	NO	NO
35	VAR_LOW1	IEEE	R/W	SI	NO	NO
37	VAR_HIGH1	IEEE	R/W	SI	NO	NO
43	OUT_LOW2	IEEE	R/W	SI	NO	NO
45	OUT_HIGH2	IEEE	R/W	SI	NO	NO
47	VAR_LOW2	IEEE	R/W	SI	NO	NO
49	VAR_HIGH2	IEEE	R/W	SI	NO	NO
15	VALOR_ALAR0	IEEE	R/W	SI	NO	NO
17	VALOR_ALAR1	IEEE	R/W	SI	NO	NO
75	FREC_NOM	IEEE	R/W	SI	NO	NO
27	OUT_MED0	IEEE	R/W	SI	NO	NO
29	VAR_MED0	IEEE	R/W	SI	NO	NO
39	OUT_MED1	IEEE	R/W	SI	NO	NO
41	VAR_MED1	IEEE	R/W	SI	NO	NO
51	OUT_MED2	IEEE	R/W	SI	NO	NO
53	VAR_MED2	IEEE	R/W	SI	NO	NO
212	PROG_VELOC	IEEE	R/W	SI	NO	NO

Estas variables se usan para la configuración del equipo.

BASE_ADD: Es la dirección base para formar la dirección absoluta de cada variable.

ESCALAV: Valor de la tensión de primario. Si no se emplean transformadores de tensión, este valor debe coincidir con el valor de ajuste de secundario.

ESCALAI: Valor de la corriente nominal de primario. Si no se usan transformadores de intensidad, este valor debe coincidir con el valor de secundario (1 ó 5 A).

REF_ENER: Es el valor del impulso de energía, que proporcionan las salidas digitales. Afecta igualmente a la energía activa y a la reactiva. Se puede programar cualquier valor, pero es recomendable usar un múltiplo de diez que sea coherente con la programación general del equipo, de forma que no se produzcan demasiados impulsos o demasiado pocos en la salida digital. Por ejemplo, para un equipo de 200 kW de potencia nominal, un valor de 1 kWh por impulso daría 200 impulsos por hora, que es adecuado. Un valor de 10 kWh sería asimismo adecuado, pero un valor de 0,1 kWh produciría demasiados impulsos por hora.

OUT_LOWx, OUT_HIGHx : Son los valores mínimo y máximo de la señal de salida. La salida debe escogerse entre las distintas alternativas (tensión o corriente, y 1-5-20 mA, o 1-5-10 V). Dentro de estos márgenes, se puede seleccionar el principio y fin de la señal de salida. Por ejemplo, si se elige el



margen de 5 mA, se puede hacer que la señal de salida se mueva entre -2,5 y +2,5 mA. La salida de 4-20 mA se consigue dentro del margen de 20 mA, y se programa el principio a 4, y el final a 20. Estos dos valores corresponden a X0 y X1 en las figuras que describen las salidas analógicas, y deben ser escritos en valor absoluto, en formato en coma flotante.

VAR_LOWx, VAR_HIGHx : Son los valores de la variable asociada a la salida. Significan el valor mínimo y máximo respectivamente de dicha variable, y corresponden a los valores Y0 y Y1 en las figuras. Se escriben igualmente en coma flotante, pero referidos al valor nominal de a variable deseada, de forma que de programa entre el 0 y el 120%. Se puede elegir cualquier valor, pero la precisión va disminuyendo conforme este valor es más pequeño. No es aconsejable por tanto, elegir para Y1 un valor inferior al 50%.

OUT_MEDx, son los valores correspondientes a los puntos medios cuando la característica de salida es quebrada. Deben estar comprendidos entre los dos anteriores.

VAR_MEDx son los valores de la variable cuando la característica es quebrada. Deben estar comprendidos entre los valores <y0-<y1, pero no pueden ser negativos.

La selección de característica lineal o quebrada se realiza mediante el registro CONF_OUT (ver el pnto 3.4.-). Los bits usados son el 0 para la salida 0, bit 4 para la salida 1, y el bit 8 para la salida 2.

VALOR_ALARx : Indica el valor por encima del cual se activa la alarma. Este valor se escribe también en porcentaje de la variable asociada. Solamente es positivo, y la comparación se realiza exclusivamente en módulo. Puede tomar un valor comprendido entre 0 y 120%.

FREC_NOM: Aunque el equipo es capaz de funcionar a cualquier frecuencia comprendida entre 45 y 65 Hz, se necesita definir la frecuencia a la cual debe actuar la alarma si se selecciona, y la salida analógica asociada.

PROG_VELOC define la velocidad de la línea serie de comunicaciones, según la tabla.

PROG_VELOC	BAUD RATE
0	9600
1	300
2	600
3	1200
4	2400
5	4800
6	19200



1.2.3. – VARIABLES DE CALIBRACIÓN DE SALIDA..

OFFSET (DEC)	VARIABLE (NOMBRE)	TIPO	R/W	US.	COD.	BLOCK R/W
430	DAC0_0_I20	WORD	R/W	NO	SI	NO
431	DAC0_I20	WORD	R/W	NO	SI	NO
432	DAC0_0_I5	WORD	R/W	NO	SI	NO
433	DAC0_I5	WORD	R/W	NO	SI	NO
434	DAC0_0_I1	WORD	R/W	NO	SI	NO
435	DAC0_I1	WORD	R/W	NO	SI	NO
436	DAC1_0_I20	WORD	R/W	NO	SI	NO
437	DAC1_I20	WORD	R/W	NO	SI	NO
438	DAC1_0_I5	WORD	R/W	NO	SI	NO
439	DAC1_I5	WORD	R/W	NO	SI	NO
440	DAC1_0_I1	WORD	R/W	NO	SI	NO
441	DAC1_I1	WORD	R/W	NO	SI	NO
442	DAC2_0_I20	WORD	R/W	NO	SI	NO
443	DAC2_I20	WORD	R/W	NO	SI	NO
444	DAC2_0_I5	WORD	R/W	NO	SI	NO
445	DAC2_I5	WORD	R/W	NO	SI	NO
446	DAC2_0_I1	WORD	R/W	NO	SI	NO
447	DAC2_I1	WORD	R/W	NO	SI	NO
457	DAC0_0_V10	WORD	R/W	NO	SI	NO
458	DAC0_V10	WORD	R/W	NO	SI	NO
459	DAC0_0_V5	WORD	R/W	NO	SI	NO
460	DAC0_V5	WORD	R/W	NO	SI	NO
461	DAC0_0_V1	WORD	R/W	NO	SI	NO
462	DAC0_V1	WORD	R/W	NO	SI	NO
463	DAC1_0_V10	WORD	R/W	NO	SI	NO
464	DAC1_V10	WORD	R/W	NO	SI	NO
465	DAC1_0_V5	WORD	R/W	NO	SI	NO
466	DAC1_V5	WORD	R/W	NO	SI	NO
467	DAC1_0_V1	WORD	R/W	NO	SI	NO
468	DAC1_V1	WORD	R/W	NO	SI	NO
469	DAC2_0_V10	WORD	R/W	NO	SI	NO
470	DAC2_V10	WORD	R/W	NO	SI	NO
471	DAC2_0_V5	WORD	R/W	NO	SI	NO
472	DAC2_V5	WORD	R/W	NO	SI	NO
473	DAC2_0_V1	WORD	R/W	NO	SI	NO
474	DAC2_V1	WORD	R/W	NO	SI	NO

Estos registros se usan para calibración de las diferentes opciones de salida. Como los equipos se hallan provistos de todas las posibles opciones, se realiza la calibración de todas ellas, de forma que el usuario pueda cambiar de opción sin necesidad de una nueva calibración. El usuario no debe intentar cambiar estos registros.



1.2.4. – SALIDAS ANALÓGICAS Y ALARMAS.

OFFSET (DEC)	VARIABLE (NOMBRE)	TIPO	R/W	US.	COD.	BLOCK R/W
217	CONF_OUT	WORD	R/W	YES	NO	NO
224	CONT_ALAR0	WORD	R/W	YES	NO	NO
225	CONT_ALAR1	WORD	R/W	YES	NO	NO
218	CONF_DAC0	BYTE	R/W	YES	NO	NO
219	CONF_DAC1	BYTE	R/W	YES	NO	NO
220	CONF_DAC2	BYTE	R/W	YES	NO	NO
213	ALARMA0	BYTE	R/W	YES	NO	NO
214	ALARMA1	BYTE	R/W	YES	NO	NO
222	CONF_HARD	BYTE	R/W	YES	NO	NO
223	CONF_MANUAL	BYTE	R/W	YES	NO	NO

Estos registros definen el modo de funcionamiento de las salidas y de las alarmas.

CONF_OUT: Es un registro de dos bytes con el significado siguiente:

Bit 15	no usado	
Bit 14	Man/Aut 2	1 : Salida manual
Bit 13	Man/Aut 1	0 : Salida correspondiente al valor de la variable
Bit 12	Man/Aut 0	
Bit 11	Volt/Current 2	1 : Salida en tensión
Bit 10	PWM Adjust 2	1 : Permiso de ajuste manual
Bit 9	Disable output 2	1: Salida disable
Bit 8	Kinked output 2	1: Quebrada
Bit 7	Volt/current 1	
Bit 6	PWM Adjust 1	
Bit 5	Disable output 1	
Bit 4	Kinked output 1	
Bit 3	Volt/current 0	
Bit 2	PWM Adjust 0	
Bit 1	Disable output 0	
Bit 0	Kinked output 0	

Los bits Man/Aut x permiten manejar la salida manualmente (bit=1), o automáticamente. En el primer caso la salida sigue el valor de la variable AN_OVERx. Esta variable se puede escribir, y permite producir una salida entre el 0 y el 120 % del margen elegido, lo que posibilita la comprobación de los circuitos de salida. La selección automática es la normal, y hace que la salida siga el valor de la variable asociada de acuerdo con el contenido de los registros CONF_DACx.

Los bits Volt/current comunican al microprocesador la configuración de la salida realizada por HW. Este hecho es necesario, debido a que la parte analógica del equipo se encuentra aislada del microprocesador, y éste necesita saber qué factores de calibración debe usar. El margen de salida seleccionado se escribe en el registro CONF_HARD.

Los bits PWM Adjust permiten la modificación de los valores de ajuste para cada salida y margen. No deben ser modificados por el usuario.

Los bits Disable output =1, desactivan la salida y la ponen a cero.



CONT_ALARx: Definen el valor del retardo de alarma. Se escriben en múltiplos de 5ms, y por tanto, un retardo de dos segundos se expresa como $2*200=400$.

CONF_DACx:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	-	X	X	X	X	X

Este registro define la variable asociada a la salida de acuerdo con la tabla:

Valor en CONF_DAC (XX)	Variable
0H	Vfr
1H	Vfs
2H	Vft
3H	Vrs
4H	Vst
5H	Vtr
6H	Pfr
7H	Pfs
8H	Pft
9H	Qfr
AH	Qfs
BH	Qft
CH	Ifr
DH	Ifs
EH	Ift
FH	Sfr
10H	Sfs
11H	Sft
12H	Cosr
13H	Coss
14H	Cost
15H	Prst
16H	Qrst
17H	Srst
18H	Cos
19H	Sinr
1AH	Sins
1BH	Sint
1CH	Sin
1DH	Freq



ALARM_A: Este byte define el nivel de alarma y la variable asociada. La variable se define de acuerdo con la tabla precedente, y el bit 7 define si la alarma es por mínimo o máximo nivel. El bit 6 indica el estado de alarma (1= activa).

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Max/min	Activa	-	X	X	X	X	X

CONF_HARD:

Este registro define el margen de salida, de acuerdo con la tabla:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	O21	O20	O11	O10	O01	O00

Salida 2

Salida 1

Salida 0

Ox1	Ox0	Margen
1	1	-
1	0	1V o 1 mA
0	1	5 V o 5 mA
0	0	10 V o 20 mA



1.2.5. – VARIABLES DE LECTURA.

Estas variables se pueden leer en cualquier momento. La variable SEQUENCE indica la correcta secuencia de las tres tensiones cuando indica cero. Cualquier valor distinto indica fallo de secuencia. La comprobación se realiza solamente después del arranque, y se visualiza además en el led TE, que parpadea durante tres segundos en caso de fallo.

OFFSET (DEC)	VARIABLE (NOMBRE)	TIPO	R/W	US.	COD.	BLOCK R
120	VFR	IEEE	R	SI	NO	SI
122	VFS	IEEE	R	SI	NO	SI
124	VFT	IEEE	R	SI	NO	SI
126	VRS	IEEE	R	SI	NO	SI
128	VST	IEEE	R	SI	NO	SI
130	VTR	IEEE	R	SI	NO	SI
132	PFR	IEEE	R	SI	NO	SI
134	PFS	IEEE	R	SI	NO	SI
136	PFT	IEEE	R	SI	NO	SI
138	QFR	IEEE	R	SI	NO	SI
140	QFS	IEEE	R	SI	NO	SI
142	QFT	IEEE	R	SI	NO	SI
144	IFR	IEEE	R	SI	NO	SI
146	IFS	IEEE	R	SI	NO	SI
148	IFT	IEEE	R	SI	NO	SI
150	SFR	IEEE	R	SI	NO	SI
152	SFS	IEEE	R	SI	NO	SI
154	SFT	IEEE	R	SI	NO	SI
156	COSR	IEEE	R	SI	NO	SI
158	COSS	IEEE	R	SI	NO	SI
160	COST	IEEE	R	SI	NO	SI
162	PRST	IEEE	R	SI	NO	SI
164	QRST	IEEE	R	SI	NO	SI
166	SRST	IEEE	R	SI	NO	SI
168	COSENO	IEEE	R	SI	NO	SI
170	FREC_RED	IEEE	R	SI	NO	SI
172	SENR	IEEE	R	SI	NO	SI
174	SENS	IEEE	R	SI	NO	SI
176	SENT	IEEE	R	SI	NO	SI
178	SENO	IEEE	R	SI	NO	SI
216	SEQUENCE	BYTE	R	SI	NO	NO

1.2.6. – VARIABLES DE IDENTIFICACIÓN.

OFFSET (DEC)	VARIABLE (NOMBRE)	TIPO	R/W	USER	CODE	BLOCK R/W
200	SER.NUMBER	STRING 10	R	NO	YES	NO
205	ID	BYTE	R/W	SI	NO	NO
206	TIPO	STRING 6	R	SI	NO	NO
77	ESCV_LAB	IEEE	R/W	SI	NO	NO
79	ESCI_LAB	IEEE	R/W	SI	NO	NO
81	VAUX_LAB	IEEE	R/W	SI	NO	NO
83	FREQ_LAB	IEEE	R/W	SI	NO	NO

SER. NUMBER es una variable de diez caracteres, que define el número de serie del equipo al ser fabricado.

ID representa la identidad del equipo conectado en red. Es un byte, y se puede cambiar a cualquier valor entre 0 y 255. No deben usarse los valores 0, 199, y 255. La identidad 199 es un valor genérico, y cualquier comando dirigido a ella será aceptado por todas las unidades conectadas.

1.2.7. –VARIABLES DE CONTROL.

OFFSET (DEC)	VARIABLE (NOMBRE)	TIPO	R/W	US.	COD.	BLOCK R/W
209	CONFIG_DIG	BYTE	R/W	SI	NO	NO
210	OUT_DIG	BYTE	R/W	SI	NO	NO
454	CODE_ACC	BYTE	W	SI	NO	NO
211	TIPO_PROT	BYTE	R/W	SI	NO	NO
212	PROG_VEL	BYTE	R/W	SI	NO	NO
213	ESCALAP	IEEE	R	SI	NO	NO
67	AN_OVER0	IEEE	R/W	SI	NO	NO
69	AN_OVER1	IEEE	R/W	SI	NO	NO
71	AN_OVER2	IEEE	R/W	SI	NO	NO

OUT_DIG:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	0	RL1	RL0

Controla el estado de los relés de salida. Un 1 en el bit hace que el relé asociado se cierre. Un 0 libera el contacto.



CONFIG_DIG: Selecciona el modo de trabajo de las salidas digitales, de acuerdo con la tabla:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	X	X	X

Bit 2	Bit1	Bit0	Salida RL1	Salida RL0
0	0	0	Ep+	Eq+
0	0	1	Digital	Digital
0	1	0	Alarma	Alarma
0	1	1	Ep+	Ep-
1	0	0	Eq+	Eq-
1	0	1	--	--
1	1	0	--	--
1	1	1	--	--

TIPO_PROT:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	-	-	-	-	-	X

Cuando vale 1, significa protocolo MODBUS. Si =0, significa JBUS.

La diferencia estriba únicamente en el orden en que los datos se envían por la línea serie.

En JBUS, los datos se envían:

S+EXP Mantisa H Mantisa M Mantisa L

En MODBUS:

Mantisa M Mantisa L S+EXP Mantisa H

ESCALAP: Extresa la potencia nominal. Su valor permite comprobar que los valores programados de tensión y corriente son correctos. Corresponde a:

$$ESCALAV * ESCALAI * \sqrt{3}$$

AN_OVERx: Permite producir una señal de salida analógica. Es flotante, y puede valer de -100% a +100%. La salida real depende de la programación de la variable elegida. Por ejemplo, si se definió la salida como 0-5 mA para 0-80% de P, un 100% significa $5 \cdot 100 / 80$, esto es, 6,25 mA.

PROG_VEL define la velocidad del puerto de comunicaciones, de acuerdo con la tabla:

Value	Speed (Bps)
0	9600
1	300
2	600
3	1200
4	2400
5	4800
6	19200



1.2.8.- CONTADORES DE ENERGÍA.

OFFSET (DEC)	VARIABLE (NOMBRE)	TIPO	R/W	US.	COD.	BLOCK R/W
302	ACT_POS	LONG	R/W	YES	NO	NO
304	ACT_NEG	LONG	R/W	YES	NO	NO
306	REACT_IND	LONG	R/W	YES	NO	NO
308	REACT_CAP	LONG	R/W	YES	NO	NO

1.2.9. – VARIABLES DE THD.

OFFSET (DEC)	VARIABLE (NOMBRE)	TIPO	R/W	US.	COD.	BLOCK R/W
500	VR_REAL	IEEE	R	SI	NO	NO
502	VS_REAL	IEEE	R	SI	NO	NO
504	VT_REAL	IEEE	R	SI	NO	NO
506	IR_REAL	IEEE	R	SI	NO	NO
508	IS_REAL	IEEE	R	SI	NO	NO
510	IT_REAL	IEEE	R	SI	NO	NO
512	VR_XTR	IEEE	R	SI	NO	NO
514	VS_XTR	IEEE	R	SI	NO	NO
516	VT_XTR	IEEE	R	SI	NO	NO
518	IR_XTR	IEEE	R	SI	NO	NO
520	IS_XTR	IEEE	R	SI	NO	NO
522	IT_XTR	IEEE	R	SI	NO	NO
524	VR_THD	IEEE	R	SI	NO	NO
526	VS_THD	IEEE	R	SI	NO	NO
528	VT_THD	IEEE	R	SI	NO	NO
530	IR_THD	IEEE	R	SI	NO	NO
532	IS_THD	IEEE	R	SI	NO	NO
534	IT_THD	IEEE	R	SI	NO	NO
536	THD_VR	IEEE	R	SI	NO	NO
538	THD_VS	IEEE	R	SI	NO	NO
540	THD_VT	IEEE	R	SI	NO	NO
542	THD_IR	IEEE	R	SI	NO	NO
544	THD_IS	IEEE	R	SI	NO	NO
546	THD_IT	IEEE	R	SI	NO	NO

V_x_REAL contiene el valor medido de V_x.

I_x_REAL contiene el valor medido de I_x.

V_x_XTR or I_x_XTR contiene el valor calculado del fundamental.

V_x_THD or I_x_THD contiene el valor de la THD en valor absoluto, voltios o amperios.

THD_V_x or THD_I_x contiene el valor de THD en valor porcentaje.



2. - COMANDOS.

2.1. – COMANDOS DE LECTURA.

Un comando de lectura consta de

ID---tipo de comando---Dirección---Número de palabras---CRCL---CRCH

ID es el byte de identidad. El tipo de comando es 03H, o 04H. La dirección es la primera posición donde se encuentran los datos. Número de palabras es un word, de 16 bits. Si solamente se desea leer un byte, el número de palabras también se debe poner a 1. Todos los datos pueden leerse por medio de comandos individuales, pero se pueden emplear también comandos múltiples para las variables más comúnmente leídas, con las limitaciones siguientes:

Todos las variables deben ser del mismo formato: de coma flotante, o long.

La longitud total del bloque no debe exceder de 12 variables.

2.2. – EJEMPLOS DE COMANDOS DE LECTURA.

Número de serie.

P.:	01H	04H	32H	0DAH	00H	05H	1FH	4AH		
	ID	CMD	DIRECCIÓN		Nº WORDS		CRCL	CRCH		
R.:	01H	04H	0AH		53H	41H	43H	49H	38H	30H
	ID	CMD	Nº BYTES		S	A	C	I	8	0
			32H	31H	39H	41H	4BH	8AH		
			2	1	9	A	CRCL	CRCH		

Valores nominales.

P.:	01H	04H	30H	30H	00H	06H	7FH	07H		
	ID	CMD	DIRECCIÓN		No.WORDS		CRCL	CRCH		
R.:	01H	04H	0CH		43H	67H	00H	00H		
	ID	CMD	No.BYTES		----- 231V -----					
			40H	A0H	00H	00H	41H	20H	00H	00H
			----- 5 A -----		----- 10 Wh -----					
			0C5H	35H						
			CRCL	CRCH						



Lectura múltiple.

P.:	01H	04H	30H	0CCH	00H	18H	3FH	3FH
	ID	CMD	DIRECCIÓN		Nº WORDS		CRCL	CRCH
R.:	01H	04H	30H	43H	60H	0B5H	07H	
	ID	CMD	Nº BYTES		----- Vfr		-----	
	43H	60H	0DAH	00H	43H	063H	0C6H	00H
	----- Vfs		-----		----- Vft		-----	
	43H	0C5H	37H	00H	43H	0C5H	2EH	00H
	----- Vrs		-----		----- Vst		-----	
	43H	0C5	1FH	00H	3EH	89H	6CH	00H
	----- Vtr		-----		----- Pfr		-----	
	00H	00H	00H	00H	00H	00H	00H	00H
	----- Pfs		-----		----- Pft		-----	
	00H	00H	00H	00H	00H	00H	00H	00H
	----- Qfr		-----		----- Qfs		-----	
	00H	00H	00H	00H	0EDH	0ACH		
	----- Qft		-----		CRCL		CRCH	



2.3. – COMANDOS DE ESCRITURA.

Los comandos de escritura constan de

ID---Tipo de Comando---Dirección---Número de palabras---Número de bytes---CRCL---CRCH

ID es la identidad. Tipo de comando es 06H, o 10H. Número de palabras es de 16 bits. Número de bytes es un byte.

No se admiten comandos múltiples. Cada variable debe ser escrita por medio de un comando individual.

2.4. – EJEMPLOS DE COMANDOS DE ESCRITURA..

Escritura de AN_OVER0

P.:	01H	10H	31H	60H	00H	02H	04H
	ID	CMD	DIRECCIÓN	Nº WORDS	Nº BYTES		
	42H	0C8H	00H	00H	39H	90H	
	----- VALOR -----				CRCL	CRCH	

R.: (Acknowledge)

	01H	10H	31H	60H	00H	02H	4FH	2AH
	ID	CMD	DIRECCIÓN	Nº WORDS	CRCL		CRCH	



3. FORMATO DE LOS DATOS.

Se usan los siguientes formatos de datos.

ASCII: Caracteres por ej. :nº de serie. Se envían en el orden especificado en las tablas.

BYTE: Ocho bits. Para estado o control. Se envían como words.

WORD: Dos bytes. Se envían MSB-LSB.

LONG: Cuatro bytes. Se envían MSB-____-____-LSB. (Ver Anexo 3).

IEEE: Cuatro bytes. Se envían S + EXP – Mh – Mm – ML . (Ver Anexo 3).



ANEXO 1. ALGORITMO CRC TIPO "CRC16".

1. – POLINOMIO GENERADOR.

El polinomio usado es:

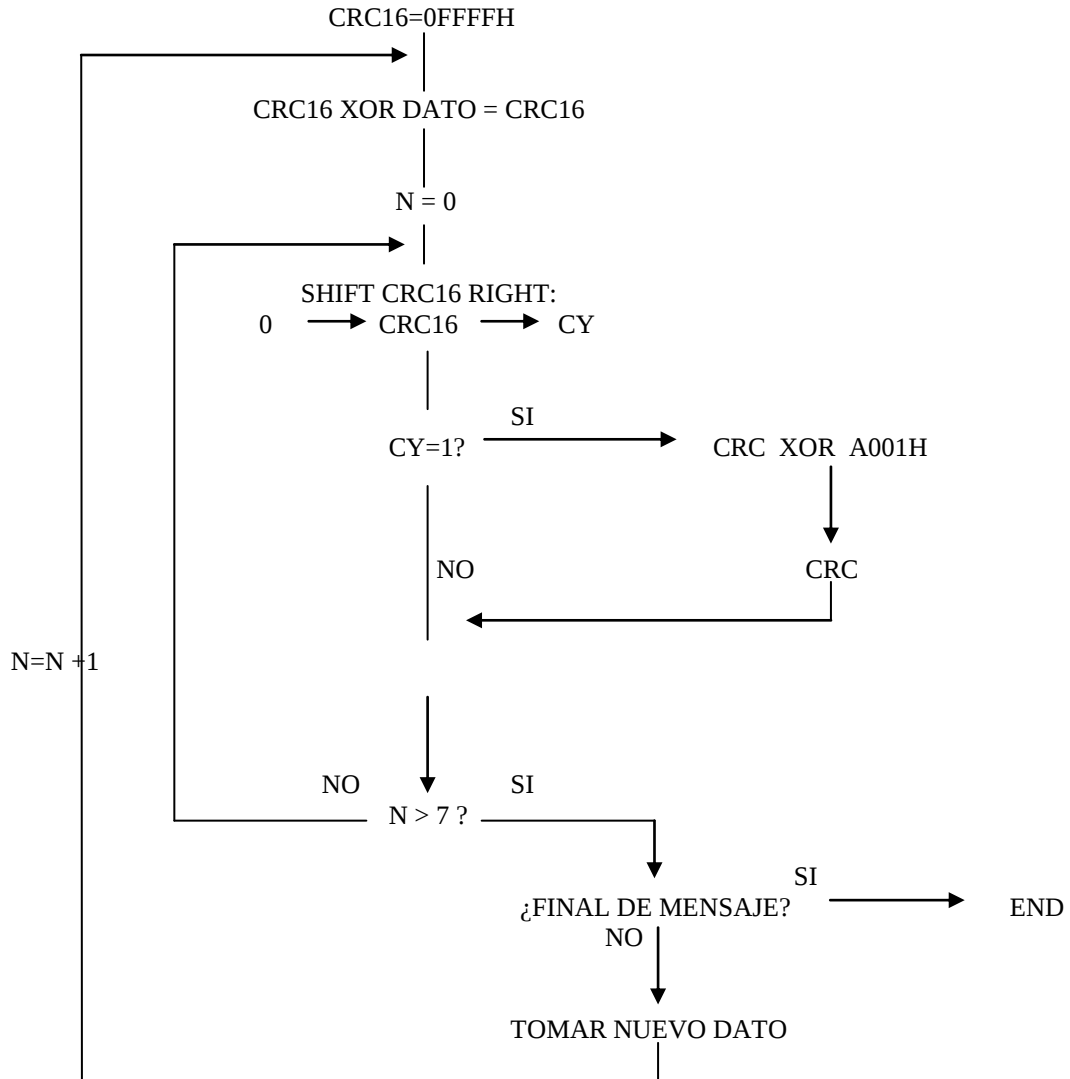
$$X^{16} + X^{15} + X^2 + 1 = 18005H$$

Para calcular el CRC, se invierte el polinomio, omitiendo el bit menos significativo.

POLINOMIO CRC16 : 1 1000 0000 0000 0101 = 18005H

POLINOMIO DE TRABAJO : 1010 0000 0000 0001 = A001H.

2. - ALGORITMO.



"DATO" es el byte recibido o que se desea transmitir.

"CRC16" es una palabra de 16 bits. El resultado de la operación se deja asimismo en CRC16. Si el crc recibido se incluye en el algoritmo, el resultado final será cero.

ANEXO 2. NOTACIÓN USADA. (IEEE 754).

Los números en coma flotante utilizan la notación IEEE754. Como la precisión que se obtiene con este tipo de notación está muy por encima de la precisión del equipo, el byte correspondiente a la mantisa baja es siempre cero. Esto puede producir en ciertos casos una pequeña discrepancia entre los datos escritos desde el ordenador principal y los datos que se leen. Por ejemplo, una cifra tal como 220.000 se puede leer del equipo como 219.987.

BYTE1	BYTE2	BYTE3	BYTE4
SIGNO+	MANTISA	MANTISA	MANTISA
EXPONENT	ALTA	MEDIA	BAJA

SIGNO: signo del número.

0H significa un número positivo.

1H significa un número negativo.

MANTISA: FRACCIÓN 0, XXX XXXX XXXX XXXX XXXX XXXX

M1	M23

EXPONENTE: Es el exponente del número con offset 127.

0: 127. (7FH)

1: 128. (80H)

-1: 126. (7EH)

Para calcular el valor:

VALOR: $(-1)^S \cdot 2^{EXP-127} \cdot (1 + \text{FRACCIÓN})$

FRACCIÓN: $\sum_{i=1}^{i=23} 2^{-i} \cdot M(i)$

La configuración de bytes es como sigue:

BYTE1:

7	6	5	4	3	2	1	0
SIGNO -----				EXPONENTE -----			
E7	E6	E5	E4	E3	E2	E1	

BYTE2:

7	6	5	4	3	2	1	0
EXP -----		MANTISA ALTA -----					
E0	M1	M2	M3	M4	M5	M6	M7

BYTE3:

7	6	5	4	3	2	1	0
----- MANTISA				MEDIA -----			
M8	M9	M10	M11	M12	M13	M14	M15

BYTE4: (Siempre nulo).

7	6	5	4	3	2	1	0
----- MANTISA BAJA -----							
M16	M17	M18	M19	M20	M21	M22	M23



ANEXO 3. FORMATO DE TRANSMISIÓN DE DATOS .

Los datos en formato IEEE se envían:

1. BYTE SIGNO + EXPONENTE
2. BYTE MANTISA ALTA
3. BYTE MANTISA MEDIA
4. BYTE MANTISA BAJA (SIEMPRE CERO)

Este modo de transmisión es el que denominamos MODO JBUS.

En algunas aplicaciones se requiere un orden inverso:

1. BYTE MANTISA MEDIA
2. BYTE MANTISA BAJA (SIEMPRE CERO)
3. BYTE SIGNO + EXPONENTE
4. BYTE MANTISA ALTA

Este modo de transmisión es el que denominamos MODBUS.

Por defecto, las unidades se programan en modo JBUS.

Esta diferencia se aplica igualmente a los números en formato LONG:

MSB, msb, LSB, lsb corresponde a formato JBUS.

LSB, lsb, MSB, msb corresponde a formato MODBUS.

Para seleccionar el formato, se usa la variable de un byte TIPO_PROT.

00H selecciona el modo JBUS.

01H selecciona el modo MODBUS



S.A.C.I.

C/ La Granja,84 ALCOBENDAS
28108 Madrid ESPAÑA
Tfno.: (34 1) 519.02.45
Fax.: (34 1) 416.96.46



CP30_PROTMB_E.doc

ED./FECHA	PUNTO	MOTIVO DEL CAMBIO	RESPONS.	Aprobado
1.1 7/01		DOCUMENTO ORIGINAL (procede de CP3protI.DOC)	F. OLMOS	
1.2 11/05	1.-	ADICIÓN OPCIÓN LÍNEA SERIE AUXILIAR	F. OLMOS	
18-7-06	1.2.8	ADICIÓN DE CONTADORES DE ENERGÍA RENUMERACIÓN	F. OLMOS	

RESP. DEP. TÉCNICO

RESP. DEP. COMERCIAL

SAC-F-35



CP400X

**TECHNICAL DESCRIPTION, COMMUNICATION
PROTOCOL, AND INSTRUCTION MANUAL**



INDEX

1.- GENERAL DESCRIPTION.

2.- TECHNICAL DESCRIPTION.

3.- VARIANTS.

4.- DIMENSIONS AND MOUNTING.

5.- CABLING.

6.-COMMUNICATION PROTOCOL.

7.- OUTPUT CONFIGURATION.

8.- RELATED SOFTWARE.

9.- SAFETY RECOMMENDATIONS.

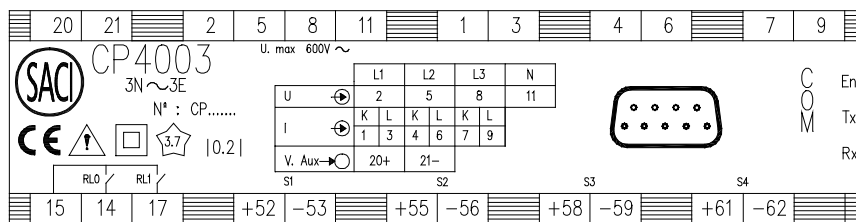


1.- GENERAL DESCRIPTION.

The CP400X Multitransducer model receives the current and voltage signals from an electric power system, and measures all the main variables related with it, allowing to send them by means of four analog outputs, two digital outputs, and one serial line, to the adequate receivers. Any analog output can be programmed to represent any live variable, for instance voltage, current, or power. The digital outputs can be configured to represent energy pulses or alarms. Through the serial output every parameter is accessible, allowing to send them to a computer to provide recording and statistical processes.

The unit uses a 16 bit microprocessor, and a ten bit plus sign A/D converter allows to get a very high accuracy, great flexibility in input and output ranges programming, and powerful communication capabilities.

Special care has been taken in the design, not only to provide good accuracy, but also high reliability, a strong immunity to EMC, -in order to use the equipment in industrial environment-, and high isolation characteristics. They are prepared for DIN rail mounting, and all the connections are done by means of screws on the front. An example of marking (for a three systems unit), is shown in the drawing.



2.- TECHNICAL DESCRIPTION.

2.1. – MEASURING PRINCIPLE.

The measuring is done in a totally digital way. After passing through the voltage and current transformers, the signals are sampled by the A/D converter. The microprocessor controls the A/D sampling, following the network frequency in order to get 32 samples per period for each signal, stores them and processes them calculating the values as the following:

$$V_i = \sqrt{\sum V_{ij}^2 / 32}$$

$$I_i = \sqrt{\sum I_{ij}^2 / 32}$$

$$P_i = \sum (V_{ij} * I_{ij}) / 32 \quad Q_i = \sum (V_{i,j-8} * I_{ij}) / 32 \quad S_i = \sqrt{P_i^2 + Q_i^2}$$

$$P_t = \sum (P_i) \quad Q_t = \sum (Q_i) \quad S_t = \sqrt{P_t^2 + Q_t^2}$$

The calculations are carried out every signal period, - 20 or 16.66 ms-, but the actual signal is the eight periods average, in order to minimize fluctuations.

In the case of three system devices, the calculations are done for the three voltage phases and the three current phases. In two systems devices, the calculations refer only to the input signals used. That is only is done the multiplication $V_{12} * I_1$, and $V_{32} * I_3$. These values added provide the total active power, but they do not have a meaning corresponding to any specific phase. They appear in the positions defined by P1 and P3, (or Q1 and Q3), and then P2 and Q2 do not appear. As there is no I_2 , this value can not be read, as well as the variables related to it, as PF2 and QF2. V_{31} , however is calculated by

$$V_{31}^2 = V_{12}^2 + V_{32}^2 + \frac{1}{2} (V_{12} * V_{32})$$

This fact means that in the case a single phase signal is applied both to V1 and V3, although there is not a voltage difference between them, the device will answer a value corresponding to this calculation.

The frequency is measured using the microprocessor internal timers. Its value is used when the energy is computed, as well as to provide the sample synchronization, which must be exactly 32 samples per signal and period. The rest of the parameters are calculated from these basic measures, totaling up to the table shown.

PARAMETER	Total	L1	L2	L3
Phase voltage	Mean	V1	V2	V3
Line voltage	Mean	V12	V23	V31
Line current	Mean	I1	I2	I3
Active power	P	P1	P2	P3
Reactive power	Q	Q1	Q2	Q3
Apparent power	S	S1	S2	S3
Power factor	PF	PF1	PF2	PF3
Reactive power factor	QF	QF1	QF2	QF3
Frequency	F	-	-	-
Active energy (pos.)	Ep+	-	-	-
Active energy (neg.)	Ep-	-	-	-
Reactive energy (ind.)	Egind(+)	-	-	-
Reactive energy (cap.)	Egcap(-)	-	-	-
THD (%) V	-	V1	V2	V3
THD (%) I	-	I1	I2	I3

THD values are calculated by means of the DFT for each signal, evaluating the fundamental content, and calculating the rest as THD value. For instance, for V_{i1} ,

$$THD(V_i) = \sqrt{(V_i^2 - V_{i1}^2) / V_{i1}^2}$$



2.2. – BLOCK DIAGRAM.

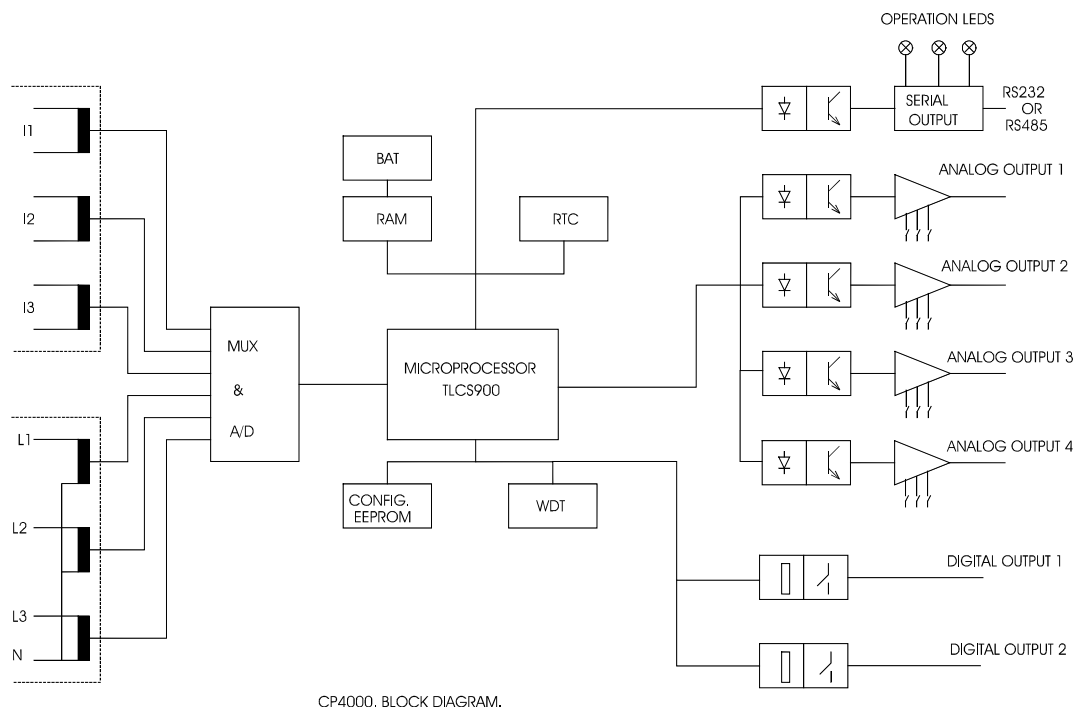
A diagram showing the following blocks is in the figure in the following figure.

2.2.1. – Input transformers.

As in the rest of transducers manufactured by SACI, both current and voltage transformers are used in the input stage, providing isolation and the level change required.

2.2.2. – Multiplexer and A/D converter.

An A/D converter, ten bit plus sign, is used to digitize the signals. An internal multiplexer allows to switch the signal to the converter. The sampling is done exactly at 32 samples per signal cycle. The microprocessor changes the timing accordingly to the supply frequency, and the device can work both at 50 and 60 Hz.



CP4000, BLOCK DIAGRAM.

DBCP403.CDR

2.2.3. – Microprocessor.

The device used is the TLCS900 from TOSHIBA. It is a 16 bit unit, 32Kbytes ROM, 1Kbyte RAM, two serial ports, ten bit A/D converter, and timers.

2.2.4. – EEPROM.

The configuration data, as full scale definition, or identity, are stored in an 4Kbit EEPROM. In this memory are stored also the calibration data, as the calibration process is done by means of SW commands.



2.2.5. – RAM.

A nonvolatile RAM is used for storing the data related to energy measurement. Two options are possible for its supply. A Ni-Cd battery can be chosen for those applications requiring long time storage, at least two months. In the cases in which no more than three days are demanded, a supercap capacitor is used, which although provides less hold time, increases the long term reliability of the equipment.

2.2.6. - WDT.

A watchdog timer is used to reset the microprocessor in the case of SW failure due to a strong electrical perturbation.

2.2.7. – RTC.

A real time clock is available in order to provide the date and hour for the device.

2.2.8. – SERIAL OUTPUT.

A serial output communication line is available in two formats: RS232, or RS485. In both cases, as in the rest of devices manufactured by SACI, is isolated from the measuring circuit, as well as from the analog or digital outputs. In this way, the serial line can be left floating, and can be connected to earth in only one point if needed. The RS485 link can be chosen as 2 or 4 wires, and several baud rates can be selected.

2.2.9. – ANALOG OUTPUTS.

Each analog output is isolated also from the measuring circuit by means of DC/DC converters and optoisolators. It is possible to select for any of them any live variable. The selection of variable is done by means of SW commands. The hardware configuration, however, is done by means of mechanical switches, and must be specified in the order.

2.2.10. – DIGITAL OUTPUTS.

Two digital outputs are provided. The mode of operation is programmable, and can be:

- A. - Energy pulses.
- B. - Alarms.
- C. - General outputs commanded by the computer.

Relays have been chosen for this purpose, and they do not impose any restriction in the load to be connected to them. In this way, if used for alarms or general purpose outputs, they can handle directly a 3A load at 250 V AC.

2.2.11. – Optical coupling.

Both analog outputs and communication line are provided with optical isolation. In this way, the outputs can be freely connected. In the case of serial line, this allows to connect the screen of the cable to earth in only one point in the installation.

2.2.12. – Power supply.

The power supply provides the different voltages needed for the circuits operation.



3.- VARIANTS.

The instruments are not intended to be changed in the field. Each option must be defined in the initial order.

3.1.- Models:

CP4000	Single phase
CP4001	Three phase, balanced
CP4002	Three phase, three wires, unbalanced
CP4003	Three phase, four wires, unbalanced

3.2.- Voltage input :

100, 230 or 400 V $\pm$ 50% (phase to phase value)

3.3.- Current input:

1 or 5 A.

3.4.- V aux:

- a) 63.5, 110, 230, or 400 V AC, $\pm$ 20%.
- b) 85...264 V AC.
- c) 24, 48, 110 V DC $\pm$ 20%.
- d) 100...260 V DC.

3.5.- Analog outputs:

- a) $\pm$ 1, $\pm$ 5, or $\pm$ 20 mA (internal switches).
- b) $\pm$ 1, $\pm$ 5, or $\pm$ 10 V (internal switches).
- c) None.

3.6.- Serial output:

- a) RS232.
- b) RS485.

3.7.-Digital outputs:

Relay type.

Pulse duration 100 ms $< T_1 < 120$ ms.

Time between pulses $T_2 > 100$ ms.

4.- DIMENSIONS AND MOUNTING.

4.1.- Dimensions.

See annex 5.

4.2.- Mounting

The device is mounted in DIN 50022 rail, or by means of two screws to the back panel.

Before connecting of the device, check the following:

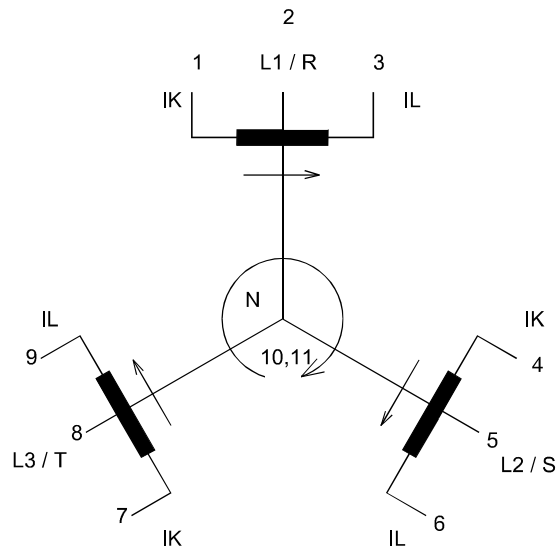
- The label data correspond to the installation values, voltage, current, etc.
- The phase sequence is right, and each phase is connected to the terminal marked.
- The current transformers are correctly mounted and cabled. The current must enter into the transformer from the K side. In this way, the terminal k must be connected to the terminal 1 in the CP4000, and the terminal l to the terminal 3.



5.- CABLING.

5.1.- Electrical wiring.

The electrical marking corresponds to the standard marking for three phase lines, as depicted:



See connection diagrams in annex 6.

5.2.-Analog outputs.

Any measured parameter can be sent through any analog output, by means of SW commands sent through the serial communication line. The output range must have been selected first among the versions above. Once the range has been selected, the possible actual output must be programmed as the following diagrams show. Any value inside the range can be programmed for the points X0, X1, or X2, or Y0, Y1 or Y2, without any other limitation than:

$$X0 < X1 < X2$$

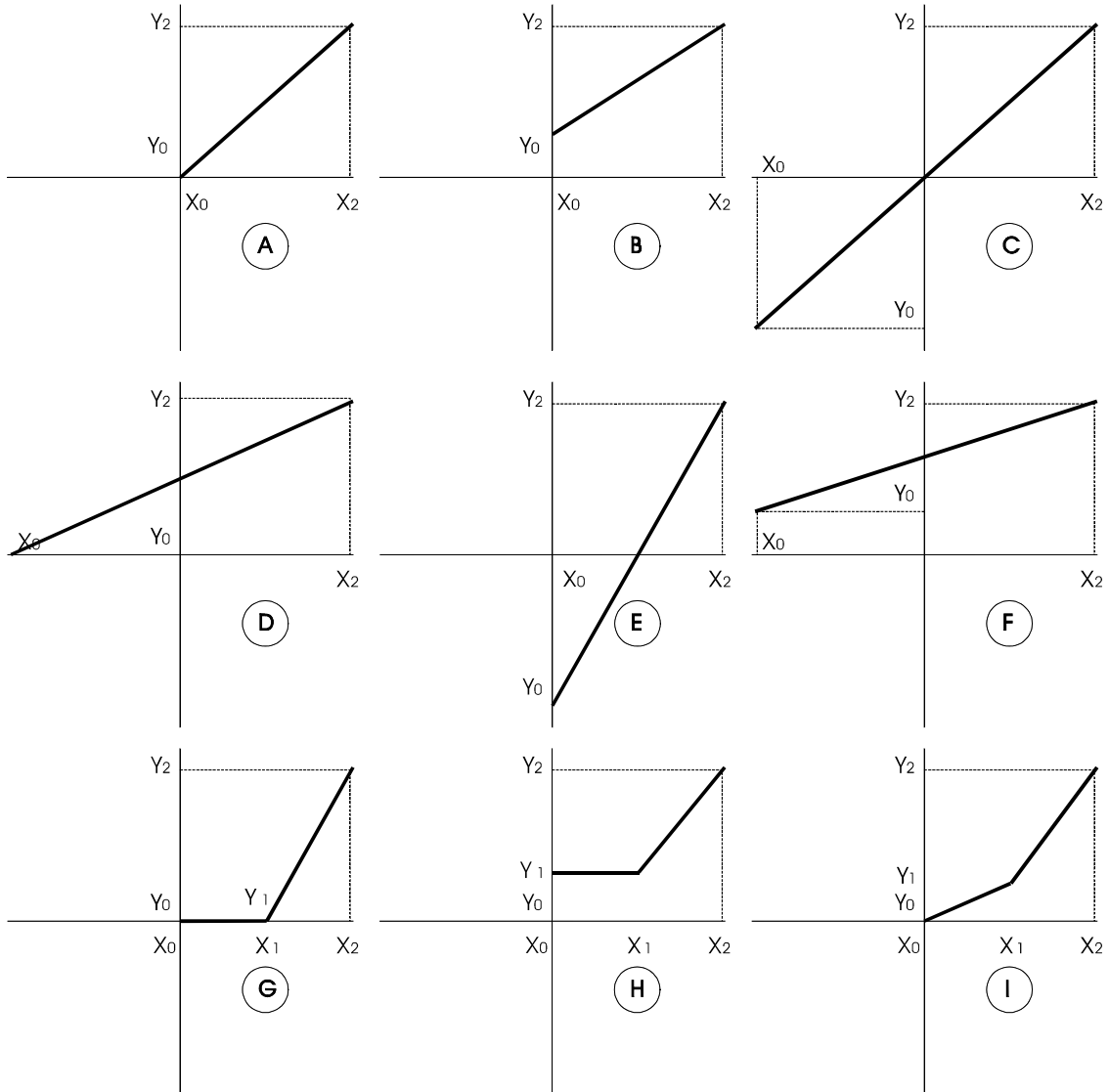
$$Mmin < Y0 < Y1 < Y2 < Mmax$$

Where Mmin is the lowest value for the output range, and Mmax is the highest one.

In the case the output is straight, without any kink, the values X1 and Y1 can be omitted.

The annex drawings show the different ways to program the analog outputs.

Each analog output is fed from a small, isolated, DC/DC converter, 15 V output. This fact limits the burden capability to a maximum of 12/Io ohms, for instance, 600 ohms for 20 mA output range.



CP2/3/4000. OUTPUT TRANSFER FUNCTIONS.

Cp2/3/41.cdr



5.3.- Serial line.

Two types of serial line can be ordered: RS232, and RS485.

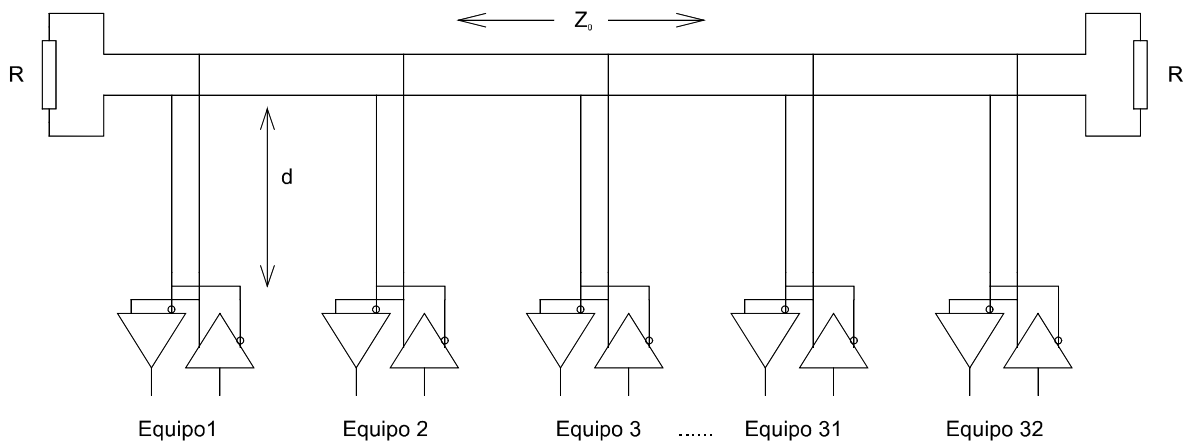
	Name	Pin	PC
RS232:	GND	5	5
	TD	2	3
	RD	3	2
	RTS	8	7
	CTS	7	8
	DSR	-	6
	DTR	-	4
RS485	GND	4	
	L+	9	
	L-	5	
	LA+	8	
	LA-	7	

RS232 pins are named as the computer to be connected. TD means data transmitted by the computer, and received by the CP4000X. RTS and CTS signals are used only for loop back purposes. They allow knowing if the transducer is powered up.

RS485 connections are named L+ and L-. They must be connected to homologous terminals in the network, that is, the same wire goes to all terminals marked L+, and the other wire goes to all terminals marked L-. These terminals are marked as DATA A or DATA B in some RS485 converters.

If four wires connection is required, the terminals marked LA+ and LA- must be used. If the RS232 to RS485 IFRA's are used, the terminals with the same name will be connected together. If the device is going to be connected to an existing network, L+ and L- are the transmission line, and LA+ and LA- the reception line.

The following schematic diagram shows the typical connection for two wires.



If the transmission line length is longer than a few hundred meters, termination resistors are required, nominally 120 ohms.

In both cases, LEDs are provided to help in the commissioning. TE is always on, unless the device is transmitting, and indicates the transducer is powered up and working. RX is normally off, until a signal is received. TX is normally off until a signal is sent.



5.4.- Digital outputs.

Operation mode must be selected first, as shown in the table.

RL0	RL1
Reactive energy	Active energy+
Active energy -	Active energy +
Reactive energy Ind.(+)	Reactive energy Cap.(-)
Alarm 0	Alarm 1
Output 0	Output 1

Both for active or reactive energy, an energy pulse must be defined. Each time this quantity is measured, the associated relay is closed during 100 ms.

In the case of alarm mode, the variable must be defined, as well as the mode, -(maximum or minimum)-, and the value expressed in unitary ratio over the FSD. The value can be negative, and must be between -1.20 and +1.20.

The uncommitted mode does not require any value to be programmed.

The output relays have also a common point. No protection device is included, and then the manufacturer's characteristics must be carefully respected, and some kind of external sectioning device must be provided. The relays used are the model JS from FUJITSU/TAKAMISAWA, able to switch 70 VA/250 V AC, or 3 A on resistive load.

5.4.- Power supply.

A separate connection is provided for the power supply. In this way, a different supply can be used, allowing to maintain the unit working, - for instance to be able to communicate-, without measuring voltage. The AC supply consists of a multi secondary transformer, rectifiers, filters, etc. This transformer is designed for one voltage input, allowing a $\pm 20\%$ operating range. The DC supply consists of a DC/DC main converter and several secondary converters. A universal power supply can be also provided allowing to be connected from 85 to 264 V AC, or 90 to 230 V DC.



6.- COMMUNICATION PROTOCOL.

The CP4000 transducers are equipped with a serial line in order to be able to communicate all the measured variables to any device with serial input capability. There are two different versions, one as RS232 standard, and the other as RS485. The first one can be used when the connection is point to point, and the second in multidrop connections. In this case, up to 32 devices, - as the standard specifies-, can be connected to the same communication line, to a maximum length of 1200 meters, allowing gathering data coming from any measurement unit.

Both versions are isolated from the rest of circuitry by means of optical couplers. This avoids problems in earth connection, breaking any grounding loops, and allows to connect the cable screen to earth in the best point of the installation.

Although the signal levels are different, the protocol used in both is the same, and complies with the RTU JBUS/MODBUS protocol. This is a master-slave protocol, very common in PLC's. The communication is always started by the master, and no spontaneous messages are allowed. The standard transmission rate is 9600 Bps, 8 bits, no parity, and one stop bit. Other speeds are possible, programming the device, from 300 to 19.200 Bps.

Each transaction consists of one request frame, generated by the master, and one reply frame, generated by the slave. In the case the master sends a writing command, the slave sends an acknowledge message. No provisions have been done for answering exception conditions. If any command is received in a correct way, it will be executed. If not, no answer will be generated.

Function codes accepted are:

03H	Read Holding Registers (3xxxx).
04H	Read Input Registers (4xxxx).
06H	Preset Single Register (6xxxx)
10H	Preset Multiple Registers



6.1. – FRAME DESCRIPTION.

6.1.1. - DATA POLLING (Commands 03h or 04H)

Each frame consists of:

- | | |
|----------------------------------|-----------------|
| - Identity number | one byte |
| - Command code | 04H or 03H |
| - Data address | two bytes: H, L |
| - Number of registers to be read | two bytes: H, L |
| - CRC | two bytes: L, H |

6.1.2. – WRITING COMMANDS.

Preset Single Register.

- | | |
|-------------------|-----------------|
| - Identity number | one byte |
| - Command code | 06H |
| - Data address | two bytes: H, L |
| - Variable value | two bytes |
| - CRC | two bytes: L, H |

Preset Multiple Registers

- | | |
|---------------------------------|-----------------|
| - Identity number | one byte |
| - Command code | 10H |
| - Data address | two bytes: H, L |
| - Number of words to be written | two bytes: H, L |
| - Number of bytes to be written | one byte |
| - Variable value | four bytes |
| - . | . |
| - . | . |
| - Variable value | four bytes |
| - CRC | two bytes: L, H |

6.1.3. – ANSWER FRAME

Each frame consists of:

- | | |
|---------------------------------------|-----------------|
| - Identity | one byte |
| - Command code (the same as received) | 04H |
| - Number of bytes sent | one byte |
| - Variable value | four bytes |
| - . | . |
| - . | . |
| - . | . |
| - Variable value | four bytes |
| - CRC | two bytes: L, H |

6.1.4. – ACKNOWLEDGE FRAME.

It consists of:

- | | |
|--|-----------------|
| - Identity | one byte |
| - Command code (the same as received) | 10H |
| - Data address (the same as received) | two bytes: H, L |
| - Number of words written (the same as received) | two bytes: H, L |
| - CRC | two bytes: L, H |



6.2. - ADDRESS MAP.

The addressing mode for CP400X has been changed from the one in the CP300X. The main difference between them has been the addition of a base register. This base register is situated at the top of the map in absolute address 0000H, and is named BASE_ADD. The actual address for a variable is formed summing the content of this position to the offset address stated in the following tables. For instance, the address for the Network ID, (offset value = 205D), will be [BASE_ADD] + 200D. This applies not only to any variable to be read or written, but also to the BASE_ADD itself. Then it can be accessed directly through the absolute address 0000H, or using the same convention, through the [BASE_ADD] + 0000H. Any value can be programmed for the BASE_ADD, from 0 to 61440D, - 0 to 0F000H -, both inclusive. In this way, the map can be situated in any physical position, circumventing the problems we have found with some PLC's which are not able to address positions higher than 9999D.

There has been another significant change, referring to the reading and writing commands accepted. In this version both 03H and 04H commands are valid for reading a value, and 06H and 10H commands are valid for writing. Command 06H can be used for writing 16 bit wide variables (WORD type), and command 10H can be used to write any single variable.

The addressing scheme fulfils totally the MODBUS requirements for registers length. The MODBUS protocol considers each register as a 16 bit word. Then, the difference between two consecutive address numbers is two bytes. If the variable is four bytes long, the difference will be four bytes, or two digits.

The variables used in the CP400x are explained as follows.

6.2.1. – CALIBRATION VARIABLES.

OFFSET (DEC)	VARIABLE (NAME)	TYPE	R/W	USER	CODE	BLOCK R/W
400	VR100	LONG	R	NO	YES	NO
402	VS100	LONG	R	NO	YES	NO
404	VT100	LONG	R	NO	YES	NO
406	PR100	LONG	R	NO	YES	NO
408	PS100	LONG	R	NO	YES	NO
410	PT100	LONG	R	NO	YES	NO
412	QR100	LONG	R	NO	YES	NO
414	QS100	LONG	R	NO	YES	NO
416	QT100	LONG	R	NO	YES	NO
418	IR100	LONG	R	NO	YES	NO
420	IS100	LONG	R	NO	YES	NO
422	IT100	LONG	R	NO	YES	NO
424	IR0	BYTE	R	NO	YES	NO
425	VR0	BYTE	R	NO	YES	NO
426	IS0	BYTE	R	NO	YES	NO
427	VS0	BYTE	R	NO	YES	NO
428	IT0	BYTE	R	NO	YES	NO
429	VT0	BYTE	R	NO	YES	NO

These variables are used for hardware calibration. The customer must not use them, as they affect to the full scale input range. Prior to be changed, the access code must be sent. If code is not sent, when asked, the CP400x will answer the calibration value stored previously.



6.2.2. – CONFIGURATION VARIABLES.

OFFSET (DEC)	VARIABLE (NAME)	TYPE	R/W	USER	CODE	BLOCK R/W
0	BASE_ADD	WORD	R/W	YES	NO	NO
1	ESCALAV	IEEE	R/W	YES	NO	NO
5	ESCALAI	IEEE	R/W	YES	NO	NO
9	REF_ENER	IEEE	R/W	YES	NO	NO
19	OUT_LOW0	IEEE	R/W	YES	NO	NO
21	OUT_HIGH0	IEEE	R/W	YES	NO	NO
23	VAR_LOW0	IEEE	R/W	YES	NO	NO
25	VAR_HIGH0	IEEE	R/W	YES	NO	NO
31	OUT_LOW1	IEEE	R/W	YES	NO	NO
33	OUT_HIGH1	IEEE	R/W	YES	NO	NO
35	VAR_LOW1	IEEE	R/W	YES	NO	NO
37	VAR_HIGH1	IEEE	R/W	YES	NO	NO
43	OUT_LOW2	IEEE	R/W	YES	NO	NO
45	OUT_HIGH2	IEEE	R/W	YES	NO	NO
47	VAR_LOW2	IEEE	R/W	YES	NO	NO
49	VAR_HIGH2	IEEE	R/W	YES	NO	NO
15	VALOR_ALAR0	IEEE	R/W	YES	NO	NO
17	VALOR_ALAR1	IEEE	R/W	YES	NO	NO
75	FREC_NOM	IEEE	R/W	YES	NO	NO
27	OUT_MED0	IEEE	R/W	YES	NO	NO
29	VAR_MED0	IEEE	R/W	YES	NO	NO
39	OUT_MED1	IEEE	R/W	YES	NO	NO
41	VAR_MED1	IEEE	R/W	YES	NO	NO
51	OUT_MED2	IEEE	R/W	YES	NO	NO
53	VAR_MED2	IEEE	R/W	YES	NO	NO
55	OUT_LOW3	IEEE	R/W	YES	NO	NO
57	OUT_HIGH3	IEEE	R/W	YES	NO	NO
59	VAR_LOW3	IEEE	R/W	YES	NO	NO
61	VAR_HIGH3	IEEE	R/W	YES	NO	NO
63	OUT_MED3	IEEE	R/W	YES	NO	NO
65	VAR_MED3	IEEE	R/W	YES	NO	NO

These variables are used to configure the device, programming in it:

ESCALAV: Means the primary nominal voltage value. If voltage transformers are not used, this value corresponds to the hardware nominal voltage.

ESCALAI: Means the primary nominal current value. If current transformers are not used, this value corresponds to the hardware nominal current, (i.e., 1 or 5 A).

REF_ENER: Means the energy value to produce an energy pulse by the digital output. This value affects equally to the Active or Reactive energy. Its value can be any, but is recommended to use a multiple of ten, related with the power nominal value, in order to not generate a very high or very low pulse rate. (For instance, if primary power was 200 Kw, a possible value for this variable could be 1Kwh, which gives 200 pulses by hour. A value of .1 Kwh gives a very high rate. A value of 10 Kwh is also adequate).

OUT_LOWx, OUT_HIGHx : They mean the lowest and highest output values. The output must be chosen among the different ranges (voltage or current, and 1-5-20 mA, or 1-5-10 V). Inside this range,



the beginning, and ending values can be programmed. For instance, when 5 mA is selected, the lowest output value can be programmed to be -2,5 mA, and the highest value to +2,5 mA. These values correspond to X0 and X2 in the commercial description, and must be written in absolute values, that is, for instance, 2 mA in floating point format.

VAR_LOWx, VAR_HIGHx : They mean the lowest and highest input values, associated to the output. They correspond to Y0 and Y2 in the commercial description. Any value inside the nominal values is allowed, but if the input range is reduced, the input precision is reduced too. Then, it is not recommended to choose a value for Y2-Y0 lower than half scale. These values are written in percentage, also in floating point format, for instance, 80.00 for 80%.

OUT_MEDx, are the middle points for kinked characteristic output. It must be located between the OUT_LOW and OUT_HIGH values.

VAR_MEDx are the variable values corresponding to the middle point kinked output. They must be located also between the VAR_LOW and VAR_HIGH values, but they must not be negative. The value corresponds to the percentage of the nominal output value.

The selection between linear and kinked output is programmed by means of register CONF_OUT (see point 6.2.4.-). The bits used are bit 0 for output 1, bit 4 for output 2, bit 8 for output 3, and bit 12 for output 4.

VALOR_ALARx : Means the alarm value above or below that, the digital output will switch. This value is expressed in percentage value referred to the variable associated, and can be any number between 1 and +120.

FREC_NOM: It means the base frequency for analog output and alarm function. As the device can be operated at any frequency, this value defines only the base point around which the alarm and output characteristic will be calculated.

6.2.3. – OUTPUT HARDWARE CALIBRATION VARIABLES.

OFFSET (DEC)	VARIABLE (NAME)	TYPE	R/W	USER	CODE	BLOCK R/W
430	DAC0 0 I20	WORD	R/W	NO	YES	NO
431	DAC0 I20	WORD	R/W	NO	YES	NO
432	DAC0 0 I5	WORD	R/W	NO	YES	NO
433	DAC0 I5	WORD	R/W	NO	YES	NO
434	DAC0 0 I1	WORD	R/W	NO	YES	NO
435	DAC0 I1	WORD	R/W	NO	YES	NO
436	DAC1 0 I20	WORD	R/W	NO	YES	NO
437	DAC1 I20	WORD	R/W	NO	YES	NO
438	DAC1 0 I5	WORD	R/W	NO	YES	NO
439	DAC1 I5	WORD	R/W	NO	YES	NO
440	DAC1 0 I1	WORD	R/W	NO	YES	NO
441	DAC1 I1	WORD	R/W	NO	YES	NO
442	DAC2 0 I20	WORD	R/W	NO	YES	NO
443	DAC2 I20	WORD	R/W	NO	YES	NO
444	DAC2 0 I5	WORD	R/W	NO	YES	NO
445	DAC2 I5	WORD	R/W	NO	YES	NO
446	DAC2 0 I1	WORD	R/W	NO	YES	NO
447	DAC2 I1	WORD	R/W	NO	YES	NO
448	DAC3 0 I20	WORD	R/W	NO	YES	NO
449	DAC3 I20	WORD	R/W	NO	YES	NO
450	DAC3 0 I5	WORD	R/W	NO	YES	NO
451	DAC3 I5	WORD	R/W	NO	YES	NO
452	DAC3 0 I1	WORD	R/W	NO	YES	NO
453	DAC3 I1	WORD	R/W	NO	YES	NO

These variables are used to calibrate the different output ranges. The transducer is equipped with mA or V outputs, and then, only three different scales are used. Hardware adjustments are not made by means of potentiometers, but by software commands, and storing their values in EEPROM. To change these values, the access code must be sent first. The user must not try to change these values.

6.2.4. – ANALOG OUTPUT AND ALARM CONFIGURATION.

OFFSET (DEC)	VARIABLE (NAME)	TYPE	R/W	USER	CODE	BLOCK R/W
217	CONF_OUT	WORD	R/W	YES	NO	NO
224	CONT_ALAR0	WORD	R/W	YES	NO	NO
225	CONT_ALAR1	WORD	R/W	YES	NO	NO
218	CONF_DAC0	BYTE	R/W	YES	NO	NO
219	CONF_DAC1	BYTE	R/W	YES	NO	NO
220	CONF_DAC2	BYTE	R/W	YES	NO	NO
213	ALARMA0	BYTE	R/W	YES	NO	NO
214	ALARMA1	BYTE	R/W	YES	NO	NO
222	CONF_HARD	BYTE	R/W	YES	NO	NO
223	CONF_MANUAL	BYTE	R/W	YES	NO	NO
221	CONF_DAC3	BYTE	R/W	YES	NO	NO

These variables are used for configuring the analog outputs mode, and to setting the alarm conditions.

CONF_OUT: It is a double byte variable and it is used as follows:

Bit 15	Volt/Current 4	1 : Voltage output
Bit 14	DAC Adjust 4	1 : Allows output adjust
Bit 13	Disable output 4	1: Reset output to 0
Bit 12	Kinked output 4	1: Kinked
Bit 11	Volt/current 3	1: Voltage output
Bit 10	DAC Adjust 3	1 : Allows output adjust
Bit 9	Disable output 3	1: Reset output to 0
Bit 8	Kinked output 3	1: Kinked
Bit 7	Volt/Current 2	1 : Voltage output
Bit 6	DAC Adjust 2	1 : Allows output adjust
Bit 5	Disable output 2	1: Reset output to 0
Bit 4	Kinked output 2	1: Kinked
Bit 3	Volt/current 1	1: Voltage output
Bit 2	DAC Adjust 1	1 : Allows output adjust
Bit 1	Disable output 1	1: Reset output to 0
Bit 0	Kinked output 1	1: Kinked

Bits Volt/current tell the microprocessor if the output is a voltage or a current value. This is needed as the V/I selection is done by switches, and the microprocessor must know the mode and its range. The range selected is written in the position CONF_HARD.

Bits DAC Adjust allow to change the HW settings for the D/A converters. If they are set to 1, the variables DACx_x_x can be modified. This belongs to the calibration process, and then, these bits must not be changed by the user.

Bits Disable output set each output to 0 value.

Bits Kinked select the kinked output for each one.

Byte CONT_ALARx: In these words must be written the delay associated with the setting of alarms.



The value is a multiple of 5 ms, then a two seconds delay must be written as $2 \times 200 = 400$.

Byte CONF_DACx:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	-	X	X	X	X	X

This byte defines which variable is associated to the analog output, as the table:

Value written in CONF_DAC (XX)	Variable
0H	Vfr
1H	Vfs
2H	Vft
3H	Vrs
4H	Vst
5H	Vtr
6H	Pfr
7H	Pfs
8H	Pft
9H	Qfr
AH	Qfs
BH	Qft
CH	Ifr
DH	Ifs
EH	Ift
FH	Sfr
10H	Sfs
11H	Sft
12H	Cosr
13H	Coss
14H	Cost
15H	Prst
16H	Qrst
17H	Srst
18H	Cos
19H	Sinr
1AH	Sins
1BH	Sint
1CH	Sin
1DH	Freq

Byte CONF_MANUAL:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	-	-	X	X	X	X



This byte defines the manual mode for any analog output. If set to 1, the value sent through the output is the value represented by the AN_OVERx variable. This allows to simulate the output to check if the signal is being sent to the receiver.

ALARMAx: This byte allows to define the alarm level and the variable associated with it. The variable is selected according to the previous table, the same as the CONF_DAC byte. The bit 7 defines if the alarm setting is on maximum or minimum (1= max). Bit 6 indicates the alarm condition, meaning that the signal has reached its alarm level, although perhaps, due to the delay, the output relay is not yet set.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Max/min	Active	-	X	X	X	X	X

CONF_HARD:

This byte allows to define the output range, according to the bits Ox1, Ox0:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
O41	O40	O31	O30	O21	O20	O11	O10

Ox1	Ox0	Range
1	1	-
1	0	1 V or 1 mA
0	1	5 V or 5 mA
0	0	10 V or 20 mA



6.2.5. – READING VARIABLES.

These variables can be read at any moment. Its meaning is self explanatory. The variable SEQUENCE when =0, indicates the correct connection of the three voltage phases. Any value different from zero indicates wrong sequence. The sequence test is done only after a reset, and is intended as help for the installation. It is visualized also by means of the Enable LED, which in the case of wrong sequence, flashes during about three seconds.

OFFSET (DEC)	VARIABLE (NAME)	TYPE	R/W	USER	CODE	BLOCK R
120	VFR	IEEE	R	YES	NO	YES
122	VFS	IEEE	R	YES	NO	YES
124	VFT	IEEE	R	YES	NO	YES
126	VRS	IEEE	R	YES	NO	YES
128	VST	IEEE	R	YES	NO	YES
130	VTR	IEEE	R	YES	NO	YES
132	PFR	IEEE	R	YES	NO	YES
134	PFS	IEEE	R	YES	NO	YES
136	PFT	IEEE	R	YES	NO	YES
138	QFR	IEEE	R	YES	NO	YES
140	QFS	IEEE	R	YES	NO	YES
142	QFT	IEEE	R	YES	NO	YES
144	IFR	IEEE	R	YES	NO	YES
146	IFS	IEEE	R	YES	NO	YES
148	IFT	IEEE	R	YES	NO	YES
150	SFR	IEEE	R	YES	NO	YES
152	SFS	IEEE	R	YES	NO	YES
154	SFT	IEEE	R	YES	NO	YES
156	COSR	IEEE	R	YES	NO	YES
158	COSS	IEEE	R	YES	NO	YES
160	COST	IEEE	R	YES	NO	YES
162	PRST	IEEE	R	YES	NO	YES
164	QRST	IEEE	R	YES	NO	YES
166	SRST	IEEE	R	YES	NO	YES
168	COSENO	IEEE	R	YES	NO	YES
172	SENR	IEEE	R	YES	NO	YES
174	SENS	IEEE	R	YES	NO	YES
176	SENT	IEEE	R	YES	NO	YES
178	SENO	IEEE	R	YES	NO	YES
170	FREC_RED	IEEE	R	YES	NO	YES
216	SEQUENCE	BYTE	R	YES	NO	NO

6.2.6. – IDENTIFICATION VARIABLES.

OFFSET (DEC)	VARIABLE (NAME)	TYPE	R/W	USER	CODE	BLOCK R/W
200	SER.NUMBER	STRING 10	R	NO	YES	NO
205	ID	BYTE	R/W	YES	NO	NO
206	TIPO	STRING 6	R	YES	NO	NO
77	ESCV_LAB	IEEE	R/W	YES	NO	NO
79	ESCI_LAB	IEEE	R/W	YES	NO	NO
81	VAUX_LAB	IEEE	R/W	YES	NO	NO
83	FREQ_LAB	IEEE	R/W	YES	NO	NO

SERIAL NUMBER is a string of ten characters, specifying the number stored during manufacturing.

ID represents the identity number. It is a binary number, and its value can be from 0 to 255. Identity 199 must be reserved, and can not be used, as it is a generic address. This means that any device will answer to this number, as it was theirs. Special care must be take when the transducers are connected forming a network, as every write command will be understood by all of them.

Variables named _LAB are the marked in the front label.

6.2.7. – CONTROL VARIABLES.

OFFSET (DEC)	VARIABLE (NAME)	TYPE	R/W	USER	CODE	BLOCK R/W
209	CONFIG_DIG	BYTE	R/W	YES	NO	NO
210	OUT_DIG	BYTE	R/W	YES	NO	NO
454	CODE_ACC	BYTE	W	YES	NO	NO
211	TIPO_PROT	BYTE	R/W	YES	NO	NO
212	PROG_VEL	BYTE	R/W	YES	NO	NO
13	ESCALAP	IEEE	R	YES	NO	NO
67	AN_OVER0	IEEE	R/W	YES	NO	NO
69	AN_OVER1	IEEE	R/W	YES	NO	NO
71	AN_OVER2	IEEE	R/W	YES	NO	NO
73	AN_OVER3	IEEE	R/W	YES	NO	NO

OUT_DIG:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	0	RL1	RL0

Writing a 1 in the position marked sets the corresponding relay. A 0 resets the relay.

CONFIG_DIG: This variable allows to select the working mode of the digital outputs, according to the value written.



Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	X	X	X

Bit2	Bit1	Bit0	RL1	RL0
0	0	0	Ep+	Eq+
0	0	1	SW controlled mode	
0	1	0	Alarm mode	
0	1	1	Ep+	Ep-
1	0	0	Eq+	Eq-

TIPO_PROT:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	-	-	-	-	-	X

This is a one bit only variable. When =1 means MODBUS. When =0, means JBUS.

A special remark must be done on this subject. The difference is only in the order in which floating point format data are sent. In mode JBUS, data is sent as

S+EXP Mantissa H Mantissa M Mantissa L

In mode MODBUS data is sent as

Mantissa M Mantissa L S+EXP Mantissa H

ESCALAP: This read only variable express the nominal power. Its value allows to confirm if the current and voltage programming has been correctly done. Its answer will be equal to

$$ESCALAV * ESCALAI * \sqrt{3}$$

AN_OVERx: This variable allows to produce an analog value in its respective output. The format is IEEE, and its range is from -100% to +100%. The actual output depends on the output range previously defined, for instance, if -3 to +3 mA is the full scale deviation, +50% means 1.5 mA. When 4 to 20 mA is used, there is no sense in adjusting the negative range, and only the positive one must be chosen.

(However, if the negative range is adjusted, as the output is essentially bidirectional, is possible to get an analog output outside the 4-20 mA range).

PROG_VEL defines the baud rate for the serial communication line, according to:

Value	Speed (Bps)
0	9600
1	300
2	600
3	1200
4	2400
5	4800
6	19200



6.2.8.- ENERGY COUNTERS.

OFFSET (DEC)	VARIABLE (NAME)	TYPE	R/W	USER	CODE	BLOCK R/W
302	ACT_POS	LONG	R/W	YES	NO	NO
304	ACT_NEG	LONG	R/W	YES	NO	NO
306	REACT_IND	LONG	R/W	YES	NO	NO
308	REACT_CAP	LONG	R/W	YES	NO	NO

6.2.9. – THD VARIABLES.

OFFSET (DEC)	VARIABLE (NAME)	TYPE	R/W	USER	CODE	BLOCK R/W
500	VR_REAL	IEEE	R	YES	NO	NO
502	VS_REAL	IEEE	R	YES	NO	NO
504	VT_REAL	IEEE	R	YES	NO	NO
506	IR_REAL	IEEE	R	YES	NO	NO
508	IS_REAL	IEEE	R	YES	NO	NO
510	IT_REAL	IEEE	R	YES	NO	NO
512	VR_XTR	IEEE	R	YES	NO	NO
514	VS_XTR	IEEE	R	YES	NO	NO
516	VT_XTR	IEEE	R	YES	NO	NO
518	IR_XTR	IEEE	R	YES	NO	NO
520	IS_XTR	IEEE	R	YES	NO	NO
522	IT_XTR	IEEE	R	YES	NO	NO
524	VR_THD	IEEE	R	YES	NO	NO
526	VS_THD	IEEE	R	YES	NO	NO
528	VT_THD	IEEE	R	YES	NO	NO
530	IR_THD	IEEE	R	YES	NO	NO
532	IS_THD	IEEE	R	YES	NO	NO
534	IT_THD	IEEE	R	YES	NO	NO
536	THD_VR	IEEE	R	YES	NO	NO
538	THD_VS	IEEE	R	YES	NO	NO
540	THD_VT	IEEE	R	YES	NO	NO
542	THD_IR	IEEE	R	YES	NO	NO
544	THD_IS	IEEE	R	YES	NO	NO
546	THD_IT	IEEE	R	YES	NO	NO

Vx_REAL contains the value actually measured for Vx.

Ix_REAL contains the value actually measured for Ix.

Vx_XTR or Ix_XTR contain the value calculated for the fundamental.

Vx_THD or Ix_THD contain the total harmonic distortion in real value (volts or amps)

THD_Vx or THD_Ix contain the total harmonic distortion in percentage.



6.3. - COMMANDS.

6.3.1. – READING COMMANDS.

Any reading command consists of

ID---Command type---Address---Number of words---CRCL---CRCH

ID is the identity number. Command type accepted are 03H or 04H. Address corresponds to the summation of Base Reg. content plus the offset. Number of words is the number of registers to be read, and has two byte length.. If only one byte is needed, also the number of words written is 1. Every data is accessible by a specific command to its respective address. However, the habitual reading values can be accessed by means of block commands, with the following limitations:

The values read must be all IEEE variables.

The total reading must not exceed 12 variables.

In the following examples is assumed the Base Reg. content is 1000D (3E8H), and the protocol is MODBUS (“Big-endian ” mode).

6.3.2. – EXAMPLES OF READING COMMANDS.

Serial number reading.

Q.:	01H	04H	04H	0B0H	00H	05H	30H	0DEH		
	ID	CMD	ADDRESS		No. WORDS		CRCL	CRCH		
A.:	01H	04H	0AH		53H	41H	43H	49H	31H	30H
	ID	CMD	N°BYTES		S	A	C	I	1	0
	31H	32H	35H	41H	0BEH	0F7H				
	1	2	5	A	CRCL	CRCH				

Nominal Voltage. (ESCALAV)

Q.:	01H	04H	03H	0E9H	00H	02H	0A0H	07BH		
	ID	CMD	ADDRESS		No.WORDS		CRCL	CRCH		
A.:	01H	04H	04H		0	0	43H	0C8H	0CBH	22H
	ID	CMD	No.BYTES		----- 400 V -----			CRCL	CRCH	



Multiple reading

Q.:	01H	04H	04H	60H	00H	18H	0E0H	48H
	ID	CMD	ADDRESS		NO.WORDS		CRCL	CRCH
A.:	01H	04H	30H		5H	0H	43H	5EH
	ID	CMD	N.BY		----- Vfr=222 V -----			
	2BH	0H	43H	5EH	0AH	0H	43H	5EH
	-----Vfs=222,2 V ----				-----Vft=222 V -----			
	55H	0H	43H	0C0H	57H	0H	43H	0C0H
	----- Vrs= 384,7-----				-----Vst=384,7-----			
	47H	0H	43H	0C0H	0BEH	0H	44H	31H
	-----Vtr= 384,6-----				-----Pfr= 710,9 -----			
	0AAH	0H	44H	31H	9BH	0H	44H	31H
	-----Pfs= 710,6 -----				-----Pft= 710,4 -----			
	0ABH	0H	0C2H	86H	68H	0H	0C2H	89H
	----- Qfr= - 67,3 -----				-----Qfs= -68,7 -----			
	2H	0H	0C2H	80H	13H	0EEH		
	-----Qft= -64,0 -----				CRCL CRCH			



6.3.3. – WRITING COMMANDS.

The writing commands consists of

ID---Command type---Address---Number of words---Number of bytes---CRCL---CRCH

ID is the identity number. Command type is 06H or 10H. Number of words is two byte long.
Number of bytes is one byte long.

No block commands are allowed. Each variable must be written by a single command.

6.3.4. – EXAMPLES OF WRITING COMMANDS.

Writing of AN_OVER0

Q.:	01H	10H	4H	2BH	00H	02H	04H
	ID	CMD	ADDRESS		No.WORDS	No.BYTES	
	0H	0H	42H	48H	0B2H	52H	
	----- VALUE -----				CRCL	CRCH	

A.: Acknowledge frame

01H	10H	4H	2BH	00H	02H	30H	0F0H
ID	CMD	ADDRESS		NO.WORDS		CRCL	CRCH

6.4. DATA FORMATS.

The following data formats are used in the data transmission from the CP3000.

ASCII: As characters, for serial number, etc. They are sent in the order specified.

BYTE: Eight bits. For status or control parameters.

WORD: Two bytes. They are sent as MSB-LSB.

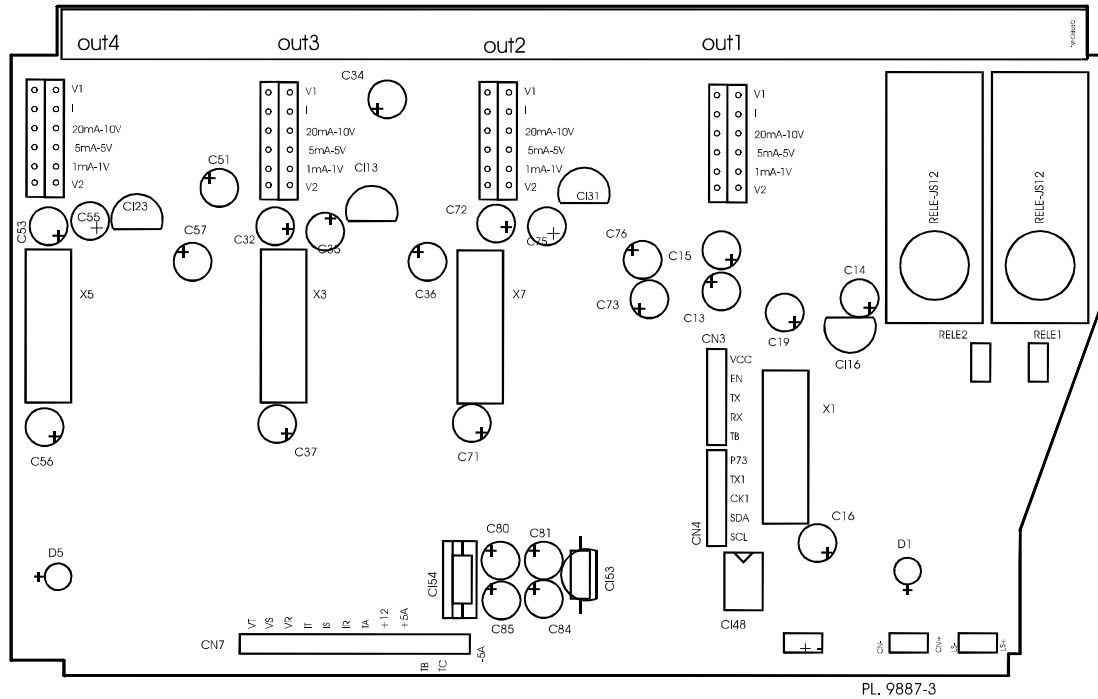
LONG: Four bytes. They are sent as MSB-____-____-LSB.

IEEE: Four bytes. They are sent as S+EXP-Mh-Mm-ML. (See annex).



7.- HARDWARE CONFIGURATION.

The device must be ordered for current or voltage output. Once defined, the range can be configured by means of internal jumpers, located in the output board, as depicted in the following figure.



For voltage output, both jumper V1 and V2 must be set. For current output, they must be open, and the jumper I must be closed. The range is selected through the rest of the jumpers. The units are adjusted in factory only for voltage or current ranges, and then, this is not an option changeable in the field. If the range is changed, the microprocessor must be informed by a command sent through the serial line, as the calibration values to be used change according to the range selected.

8.- RELATED SOFTWARE.

CP4000 V 1.05 is available. It allows to configure the device, and to read all the measures taken.

9. - SAFETY REQUIREMENTS.

The following safety precautions must be observed at all times during operation, mounting, service or repair. Failure to comply with these precautions, or with any other warning indications stated in this manual, violates the intended safety standards of design, manufacture and use of the product. SACI does not assume any liability in the case these precautions are not followed.

2. 1. - Do not connect or disconnect the device under voltage. Do not mount the equipment in field installations unless another person is present.

2.2. - Do not use the device in an explosive atmosphere, or in wet or damp areas.

2.3. - Do not open the equipment. In the case of repair, follow the service instructions, and do not substitute any part or modify the circuits.

2.4. - Change only the internal switch position with the instrument disconnected.

2.5. - The units are not provided with any type of protection device in voltage or current circuits. An external switch or circuit breaker must be provided in the installation, close to the unit. If the rating of the general protective device is less than 10 A, this one can be used for this purpose.

2.6. – Do not disconnect the current inputs without shorting first the associated transformers, as dangerous voltages can be produced.



ANNEX 1. CRC ALGORITHM TYPE "CRC16".

1. - GENERATOR POLYNOMIAL.

El polynomial used is:

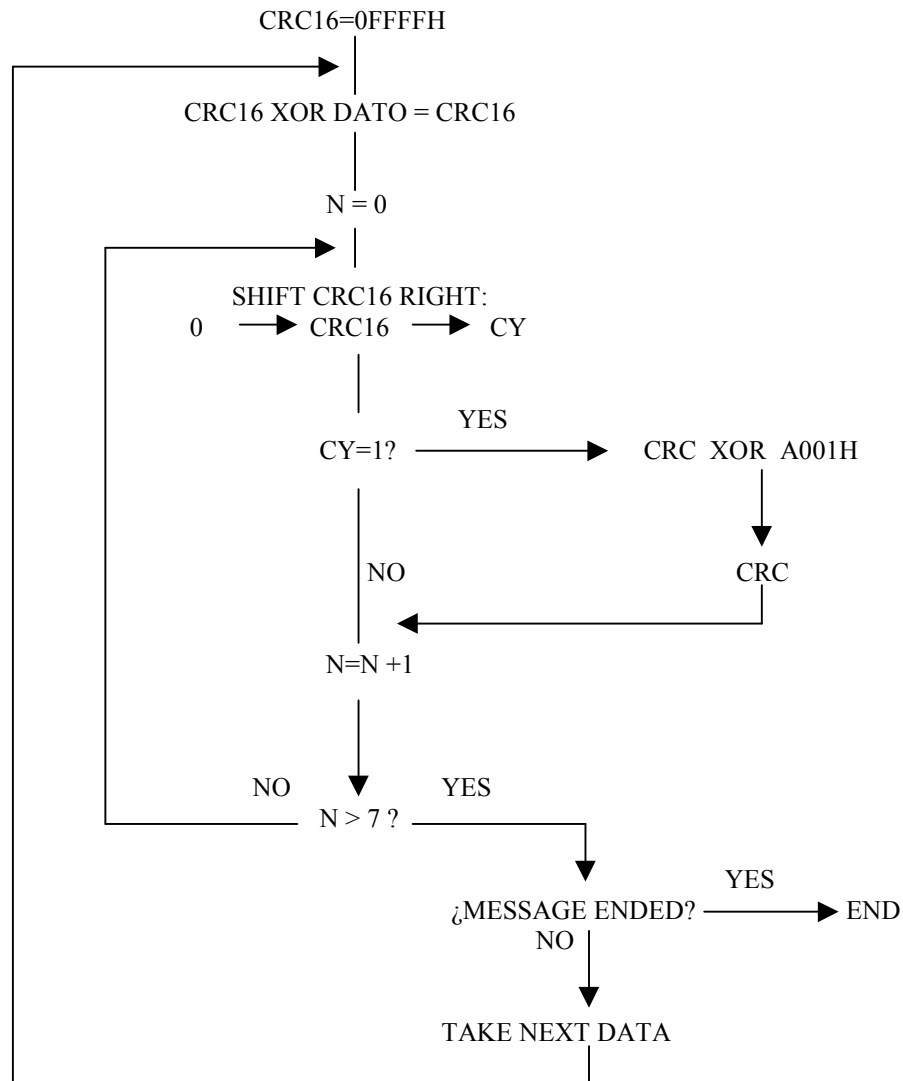
$$X^{16} + X^{15} + X^2 + 1 = 18005H$$

To get the CRC, this polynomial must be inverted, omitting the least significative bit.

CRC16 POLYNOMIAL: 1 1000 0000 0000 0101 = 18005H

WORKING POLYNOMIAL : 1010 0000 0000 0001 = A001H.

2. - ALGORITHM.



"DATA" is the byte received/or to be transmitted.

"CRC16" is a 16 bits word. The result is left in CRC16. If the received CRC characters are included in the algorithm, the final result will be zero.

ANNEX 2.IEEE NOTATION USED. (IEEE 754).

Notation IEEE754 is followed in floating point numbers. As extremely high precision is not required, the mantissa least significant byte is always zero. This could produce some disagreements between for instance, the value written and read for a high number, as 220,000 V, which can be read as 219,987.

BYTE1	BYTE2	BYTE3	BYTE4
SIGN+	MANTISSA	MANTISSA	MANTISSA
EXPONENT	HIGH	MEDIUM	LOW

SIGN: sign of the number.

0H means a positive number.

1H means a negative number.

MANTISSA: FRACTION 0, XXX XXXX XXXX XXXX XXXX XXXX

M1	M23

EXPONENT: Number exponent, with offset 127.

0: 127. (7FH)

1: 128. (80H)

-1: 126. (7EH)

To find the value:

VALUE: $(-1)^S \cdot 2^{EXP-127} \cdot (1 + \text{FRACTION})$

FRACTION: $\sum_{i=1}^{i=23} 2^{-i} \cdot M(i)$

THE BYTE CONFIGURATION IS AS FOLLOW:

BYTE1:

7	6	5	4	3	2	1	0
SIGN -----		EXPONENT				-----	
E7	E6	E5	E4	E3	E2	E1	

BYTE2:

7	6	5	4	3	2	1	0
EXP -----		MANTISSA HIGH				-----	
E0	M1	M2	M3	M4	M5	M6	M7

BYTE3:

7	6	5	4	3	2	1	0
----- MANTISSA MIDDLE -----							
M8	M9	M10	M11	M12	M13	M14	M15

BYTE4: (Always null).

7	6	5	4	3	2	1	0
----- MANTISSA LOW -----							
M16	M17	M18	M19	M20	M21	M22	M23



ANNEX 3. IEEE FLOATING POINT AND LONG INTEGERS DATA TRANSMISSION FORMAT.

IEEE data are sent in the following order:

1. SIGN + EXPONENT BYTE
2. HIGH MANTISSA
3. MEDIUM MANTISSA
4. LOW MANTISSA (ALWAYS ZERO)

This mode of transmission is referred as JBUS mode.

In certain applications, data is required in the following order:

1. MEDIUM MANTISSA
2. LOW MANTISSA (ALWAYS ZERO)
3. SIGN + EXPONENT BYTE
4. HIGH MANTISSA

This mode of transmission is referred as MODBUS mode.

Both formats are supported in the CP3000 protocol. The standard way is the JBUS mode.

This is applicable also to the long integer format. They are sent as MSB.....LSB in JBUS mode, and as LSB.....MSB in MODBUS mode.

To select the protocol type, the one byte variable, named TIPO_PROT, must be used.

00H select the mode to JBUS.

01H selects the mode to MODBUS



ANNEX 4.CRC CALCULATION.

Example of CRC calculation in BASIC

```
function crc16 (txt, lon) AS INTEGER
  DIM flag AS LONG
  DIM crc AS LONG
  DIM car AS INTEGER
  DIM bit AS INTEGER
  CRC= &HFFFF&
  FOR car =1 TO LON
    crc = crc XOR ASC(MID$ ( txt, car, 1))
    FOR bit = 0 TO 7
      flag = crc AND 1&
      crc = crc\ 2&
      IF flag = 1 THEN
        crc= crc XOR &HA001&
      END IF
    NEXT bit
  NEXT car
  crc16 = INT (crc AND &HFFFF&)
END FUNCTION
```

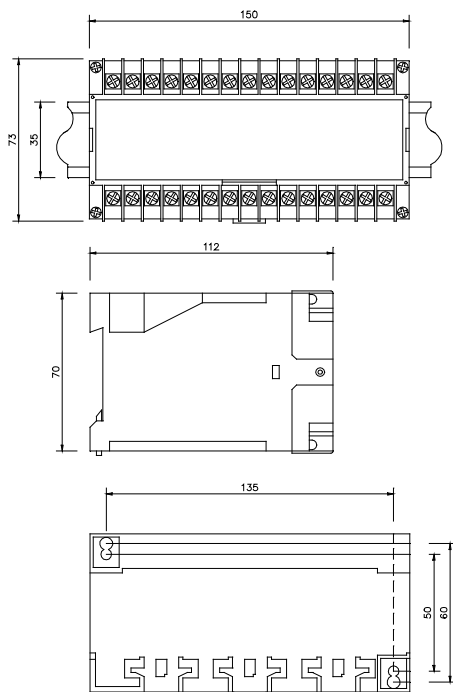
En C:

```
Void Saci_CalculoCRC ( unsigned char *Mensaje, int NumeroDeElementos)
{
  long flag, crcx;
  int car,bit;
  unsigned char v1,v2;
  crcx= 0xffff;
  for (car=0; car < NumeroDeElementos; car++)
  {
    crcx = crcx ^ Mensaje[car];
    for (bit=0; bit <8; bit++)
    {
      flag= crcx & 1
      crcx = crcx >>1 ;
      if ( flag== 1 ) { crcx = crcx ^ 0xa001; }
    }
  }
  crcx= crcx & 0xffff;
  v1 = ( unsigned char ) abs (crcx / 256 );
  v2 = ( unsigned char ) crcx - (v1*256);
  Mensaje [NumeroDeElementos] = v2;
  Mensaje [NumeroDeElementos+1] = v1;
}
```

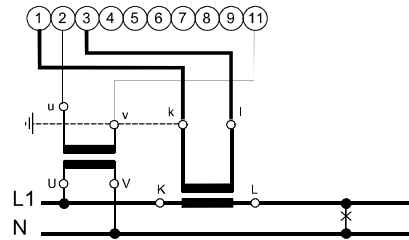
To check the message, calculate the CRC, including the CRC bytes received. If the result is zero, the message has been correctly received.



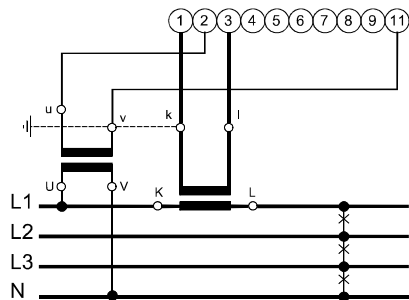
ANNEX 5. DIMENSIONS.



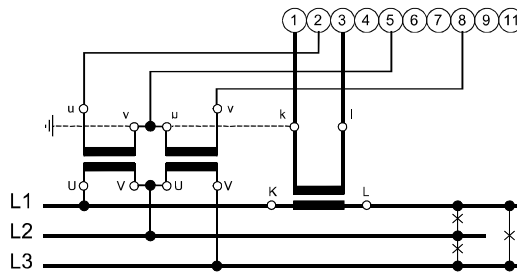
ANNEX 6. CONNECTION DIAGRAMS.



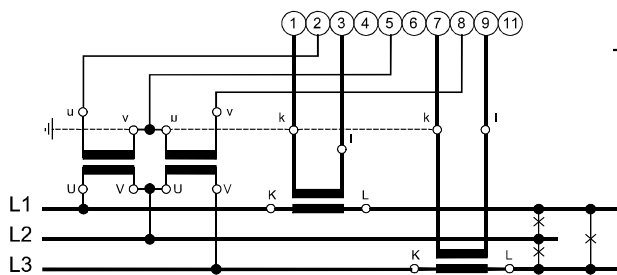
SINGLE PHASE
CP4000



THREE PHASE BALANCED
(A)
CP4001



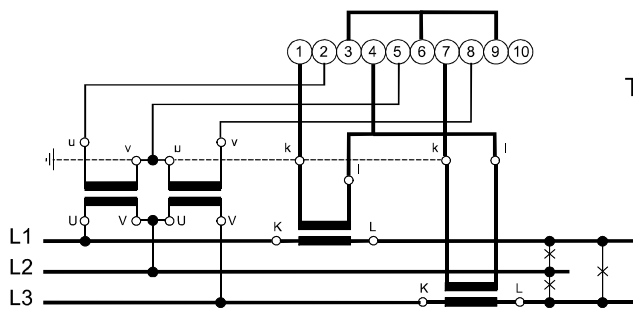
THREE PHASE BALANCED
(B)
CP4001



THREE PHASE UNBALANCED
THREE WIRE
(TWO SYSTEMS)
CP4002

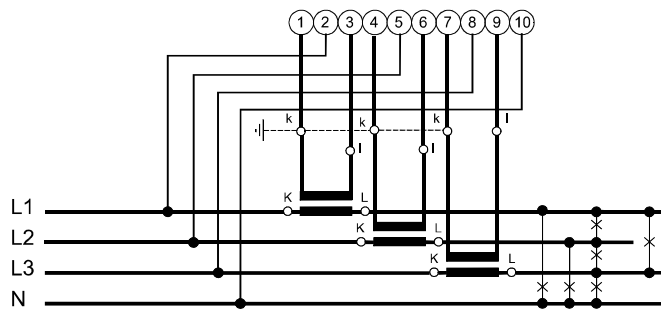
* Voltage transformers can be omitted in LV (<600 V) installations.

ESQCP4Al.cdr



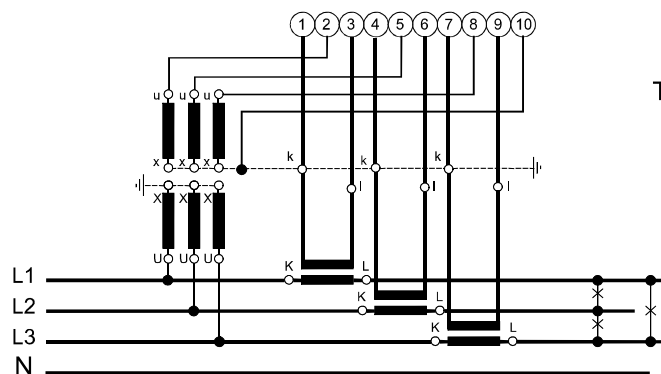
THREE PHASE UNBALANCED
THREE WIRE

CP4003



THREE PHASE UNBALANCED
FOUR WIRES
(THREE SYSTEMS)

CP4003



THREE PHASE UNBALANCED
THREE OR FOUR WIRES
(STAR CONNECTION)

CP4003

* Voltage transformers can be omitted in LV (<600 V) installations.

esqcp4bl.cdr

HOJA DE CAMBIOS

DOCUMENTO: CP40MB_I.DOC

ED.	FECHA	MOTIVO DEL CAMBIO	PUNTO AFECT.	RESPONS.	Aprobado
1.1	1/01	DOCUMENTO ORIGINAL. SE HA AÑADIDO RESPECTO A LOS ANTERIORES EL PROTOCOLO MODBUS ESTÁNDAR. LA VERSIÓN SW CORRESPONDIENTE ES CP-----		F. OLMOS	
2.1	7-1-01	CORRECCIONES DE ADDRESS-OFFSET	6.27,6.28	F. OLMOS	
2.2	29-3-01	ADICIÓN DE REGISTROS DE ENERGÍA	6.2.8	F.OLMOS	
2.3	2-3-03	DEF. BITS RELÉS. ELIM.CODEACC	6.2.7	F.OLMOS	

RESP. DEP. TÉCNICO

RESP. DEP. COMERCIAL

SAC-F-35

