

# **CP2800 Data Dictionary + Serial Communication Supplement**

For CP2800 firmware version 1.7+

Document version 2.0

2007-10-19 DPR

## Introduction

The purpose of this document is to describe the CP2800 "protocol"; that is the serial interface packet format and vocabulary.

**The CP2800 uses SMDP (Sycon Multi Drop Protocol) for data transportation.** The SMDP protocol is a multi slave, single master protocol. The CP2800 supports the multi slave on two-wire RS485 as well as single slave (point-to-point) on RS232. In the extreme case, there can be many (up to 138) CP2800s on one RS485 two-wire network. In the simplest case, there can be one CP2800 connected directly to a PC via a null-modem cable (female-female DB9 2+3 switched). **To Set the CP2800 to use RS232 (for point-to-point communication with a PC) set the SMDP address (in the COMM menu of the CP2800) to 16.** Setting the SMDP address to any other value will make the CP2800 behave as a RS485 2-wire slave. See SMDP manual for more details.

The CP2800 keeps most of it's data in a "data dictionary". This dictionary contains Boolean (0 = FALSE) and integer (4 byte) values. The software in the CP2800 constantly reads and updates the data dictionary. The dictionary can be accessed through serial communication and is the main warehouse of information.

## Formatting the SMDP DATA Field to access the data dictionary

The CP2800 uses command opcode (CMD) 0x80 for access to dictionary variables. All transfers using this opcode (0x80) are pure binary.

The SMDP data field format for reading a dictionary variable is:

Send:           <'c'><hashval-int><array index-byte>  
Returns:       <'c'><hashval-int><array index-byte><data-long>

The SMDP data field format for writing a dictionary variable is:

Send:           <'a'><hashval-int><array index-byte><data-long>  
Returns:       Nothing

Where <'a'> or <'c'> is one byte, ASCII a or c (0x61 or 0x63)

<hashval-int> is the 2-byte hash of the dictionary variable (**see table 1**)

<array index-byte> is the 1-byte index into the data object (for arrays)

<data-long> is the 4-byte big endian data field, the value read or written to the dictionary variable.

Important notes:

1. For any scalar variable, the array index is ignored.
2. For any array variable, if the array index is out of bounds, an index of 0 is used, and is considered OK.
3. "Short data" (Booleans for example) are cast into the long field.
4. All byte ordering is big endian (MS first).
5. Note that the master always sends exactly 4 bytes in the data field of the SMDP packet ('c' + 2-byte hash + 1-byte index) for reads, and 8 bytes for writes ('a' + 2-byte hash + 1-byte index + 4 bytes data).

Example 1:

*Read the dictionary variable COMP\_MINUTES (hash 0x454C)*

Send:       0x63|0x45|0x4C|0x00

Returns: 0x63|0x45|0x4C|0x00|0x00|0x01|0x36|0x23

Value returned is 00013623(hex) (79395 decimal) minutes.

Example 2:

*Start the compressor by writing 1 to EV\_START\_COMP\_REM (hash 0xD501)*

Send: 0x61|0xD5|0x01|0x00|0x00|0x00|0x00|0x01

Returns:

Note that these just show the SMDP data field formatting, **WITHOUT BYTE STUFFING OR OTHER PROTOCOL FRAMING. SEE THE SMDP MANUAL FOR FRAMING AND BYTE STUFFING INFO.**

**CAUTION: Do not write to a dictionary variable without fully understanding the effects. Do not explore unlisted hashes. Damage to equipment or injury or death to personnel may result.**

### **Data Dictionary Entries-partial list**

Name [index]	Description	Hash code
<b>Assorted Variables</b>		
CODE SUM	Firmware checksum	2B0D
MEM LOSS	TRUE if nonvolatile memory was lost	801A
CPU TEMP	CPU temperature (0.1°C)	3574
BATT OK	TRUE ('1') if clock OK	A37A
BATT LOW	TRUE ('1') if clock battery low	0B8B
COMP MINUTES	Elapsed compressor minutes	454C
MOTOR CURR A	Compressor motor current draw, in Amps	638B
<b>These match the state of the remote inputs</b>		
RI RMT COMP START	1 if TRUE, 0 if FALSE	BAF7
RI RMT COMP STOP	1 if TRUE, 0 if FALSE	3D85
RI RMT COMP ILOK	1 if TRUE, 0 if FALSE	B15A
RI SLVL	1 if TRUE, 0 if FALSE	95E3
<b>Temperatures</b>		
TEMP TNTH DEG [0]	Input water temperature (0.1°C)	0D8F
TEMP TNTH DEG [1]	Output water temperature (0.1°C)	0D8F
TEMP TNTH DEG [2]	Helium temperature (0.1°C)	0D8F
TEMP TNTH DEG [3]	Oil temperature (0.1°C)	0D8F
TEMP_TNTH_DEG_MINS [0-3]	Minimum temperatures seen (0.1°C). Uses same index as above, eg index 2 is minimum helium temperature seen.	6E58
TEMP_TNTH_DEG_MAXES [0-3]	Maximum temperatures seen (0.1°C). Uses same index as above, eg index 2 is maximum helium temperature seen.	8A1C
CLR_TEMP_PRES_MMMARKERS	Write a 1 to this to reset all min/max markers for pressures and temperatures.	D3DB
TEMP_ERR_ANY	'1' if any temperature sensor has failed.	6E2D
<b>Pressures</b>		
PRES TNTH PSI [0]	High side pressure (0.1 PSIA)	AA50
PRES TNTH PSI [1]	Low side pressure (0.1 PSIA)	AA50
PRES_TNTH_PSI_MINS	Minimum pressures seen (0.1 PSIA). Uses same index as above, eg index 1 is	5E0B

	minimum low side pressure seen.	
PRES_TNTH_PSI_MAXES	Maximum pressures seen (0.1 PSIA). Uses same index as above, eg index 1 is maximum low side pressure seen.	7A62
CLR_TEMP_PRES_MMMARKERS	Write a 1 to this to reset all min/max markers for pressures and temperatures.	D3DB
PRES_ERR_ANY	'1' if any pressure sensor has failed	F82B
H_ALP	Average low side pressure (0.1 PSIA)	BB94
H_AHP	Average high side pressure (0.1 PSIA)	7E90
H_ADP	Average delta pressure (0.1 PSIA)	319C
H_DPAC	1 <sup>st</sup> derivative of high side pressure ("bounce") (0.1 PSIA)	66FA
<b>Cryo Diode Temperatures</b>		
DIODES_UV	Diode voltage reading (uV, 0.000001V)	8EEA
DIODES_TEMP_CDK [0-1]	Diode temperature reading (0.01°K)	5813
DIODES_ERR [0-1]	True if diode failed	D644
DCAL_SEL	TRUE if using custom curve	9965
<b>Compressor control and status</b>		
EV_START_COMP_REM	Write a '1' to this to start compressor	D501
EV_STOP_COMP_REM	Write a '0' to this to stop compressor	C598
COMP_ON	'1' if compressor is on, else 0	5F95
ERR_CODE_STATUS	'0' of no error, otherwise indicates an error or warning. See table 2 of cmas_man document, "public error number", for a list of error code meanings.	65A4

## **Complete list of available dictionary variables**

The table below contains a complete list of available dictionary variables, without the hash codes. If you are interested in querying one of the variables with an unlisted hash code, contact Cryomech to obtain the hash code.

AIN MV RAW	BAD PWR	BATT LOW	BATT OK
BEV GOT START	BEV GOT STOP	CLR TEMP PRES MMARKERS	CODE SUM
COMP MINUTES	COMP ODO BAD	COMP ON	CPU TEMP
DBG PIN1 TASKSHOW	DBG PIN2 TASKSHOW	DCAL SEL	DIODES ERR
DIODES TEMP CDK	DIODES UV	DIO CNV OFFS	ERR ACKER
ERR CODE INDEX	ERR CODE STATUS	ERR D1 TOO COLD	ERR DP TOO HIGH
ERR GTP HIGH	ERR HEAD STALL	ERR HE PRES HI	ERR HE PRES LOW
ERR HISTORY	ERR HISTORY INX	ERR IWT HIGH	ERR MOT CUR LOW
ERR OTP HIGH	ERR OWT HIGH	ERR PHASE REV	ERR SYS HOSED
ERR VCC HIGH	ERR VCC LOW	EV START COMP FP	EV START COMP REM
EV STOP COMP FP	EV STOP COMP REM	GOOD PWR	GOT AD CARD
HOLDTIMER	HOW RESET	H ADP	H AHP
H ALP	H CHECK ACT	H DPAC	ILOK DREM UNSAT
INHOUSE TEST	LOCKOUT	LOCKOUT PV	MAX CPU TEMP
MEM BLESS	MEM LOSS	MIN OFF TIMER	MOTOR CURR A
P5V MV FILT	PRES CNVTED	PRES ERR	PRES ERR ANY
PRES ERR BOTH	PRES TNTH PSI	PRES TNTH PSI MAXES	PRES TNTH PSI MINS
PRODBEHAV CODE	PROD BTYPE	PROD ID	PROD SRNO
PWR LINE ERROR	PWR LINE FREQS	PWR PHASE ERROR	REL COMP MOT CONTACTOR
RI RMT COMP ILOK	RI RMT COMP START	RI RMT COMP STOP	RI SLVL
RO COMP NOT ON	RO HE PRESS HI	RO HE PRESS LOW	RO HIGH GAS TEMP
RO MOTOR ERR	RO NOT OK	RO OUT WATER TEMP	RO POWER ERR
RTC COMMIT	RTC DAY	RTC HOURS	RTC MINUTES
RTC MONTH	RTC SECONDS	RTC YEAR	SC ADDRESS
SC BAUD SET	SPECIALTC	SYS HOSED CODE	SYS TRAP CODE
SYS WARN CODE	TEMPS CNVTED	TEMP ERR	TEMP ERR ANY
TEMP TNTH DEG	TEMP TNTH DEG MAXES	TEMP TNTH DEG MINS	TOOCOLD ERRGATE
UIFC STASH	UIFC TIME STASH	UNITS CNV PRES	UNITS CNV TEMP
UNITS REQ PRES	UNITS REQ TEMP	WANT COMP ON	