Programmer's guide to the AT dataset

M.Kesteven, R.Ferris
22 January 1987

1. Introduction

The dataset will respond to 4 classes of message:

- perform a control function;
- acquire monitor data;
- initialize the command decoding registers;
- read the command decoding registers.

This note describes the procedures to be followed in order to drive the unit correctly.

2. Protocol

Every control message to the dataset must consist of five bytes:

SYNC , which indicates the start of a message, and ADH, ADL, CMDH, CMDL. These last 4 spell out the nature of the request. They are discussed in detail in section 3.

Every monitor message must have three bytes: SYNC, ADH and ADL.

The dataset response to a message:

Number of chars.	Chars. sent	interpretation
0		the dataset recognizes that it is not the intended recipient of the message.
1	NAK	the message is valid as to syntax, but describes a function which is not implemented.
2	ACK, ACK	a control request was recognized.
3	ACK, MONH, MONL	a monitor request was recognized, and the monitor data is returned.

A NAK may replace an ACK if a parity error is detected. No further response to a message can be expected after the NAK.

A DC1 will replace ACK if the dataset has been reset. If this has occurred the single-bit control lines are cleared, and may need to be re-energised.

SYNC is the SYN character (15 hex); ACK and NAK have their ASCII definitions.

3. The message details.

Each dataset in an antenna is assigned a unique number, in the range 0 to 31, known as the dataset address. A dataset will respond to only those messages which contain its specific address, coded into ADH.

3.1 Control points.

ADH = 80 hex + dataset address

ADL carries two bits of information: the nature of the control function, and the control point to be exercised. The control functions available are :

a). There are 32 control lines which can be separately activated. These lines are latched to their new state. These are the lines which will need re-activating after a RESET. There are 32 control points in this category.

b). A byte (CMDL) or a word (CMDH, CMDL) can be sent on the data bus. There is an associated 6-bit address, providing 64 separate addresses. The user specifies whether 1 or 2 bytes are to be sent. There are 64 control points in this category.

c). A byte (CMDL) or a word (CMDH, CMDL) can be sent on the data bus, with an associated strobe on one of four strobe lines. There are 4 control points in this category.

d). There are 24 internal registers to which data (CMDH, CMDL) can be written. This function is principally provided to allow error count registers to be cleared. There are 24 control points in this category.

The algorithm for coding ADL is :

ADL = base + control point index

The base defines the control function, and the index selects the specific control point.

base	function	control point index
40 hex	single-bit control lines	0 to 31
60 hex	8-bit transmission, using the address bus	0 to 63
A0 hex	16-bit transmission	0 to 63
E0 hex	8-bit transmission using the strobe lines	0 to 3
E4 hex	16-bit transmission	0 to 3
E8 hex	write to internal registers	0 to 23

For example, to send a byte to the 8th address, we use : ADL = 60 hex + 7 = 67 hex.

It is possible to inhibit particular control points - that is, a dataset may recognize that a specific control point/control function is not implemented. The dataset will return NAK if this ADL is requested. The machinery to setup the disabling is described in section 3.3 **We do not plan to utilize this facility.**

CMDH and **CMDL** are the data bytes which may be transmitted.

3.2 Monitor requests

ADH = 00 + dataset address

ADL has essentially the same coding as for the control points; one additional function is provided : analog input. There are 8 differential analog inputs, and 56 single-ended inputs.

ADL = base + monitor point index

base	function	monitor point indices
00 hex	analog input (differential input)	0 to 7
00 hex	analog input (single ended)	8 to 64
40 hex	single-bit monitor lines	0 to 31
60 hex	8-bit input, using the address bus	0 to 63
A0 hex	16-bit input	0 to 63
E0 hex	8-bit input using the strobe lines	0 to 3
E4 hex	16-bit input	0 to 3
E8 hex	read an internal register	0 to 23

The results of the monitor activity will be returned in MONL (8-bit) and in MONH,MONL for 16-bit data; a 1-bit monitor is returned in the lsb of MONL.

3.3 Initialize the decoding registers.

ADH = C0 hex + dataset address

ADL spans the range 0 to FF hex.

CMDH and **CMDL** contain an internal code for specifying the control functions (in CMDH) and monitor functions (in CMDL) described in the previous sections. Table 1 contains the complete set of initialization messages.

The set-up data is sent to the non-volatile RAM. CMDH is sent to (ADL+256) and is the CONTROL_CODE; CMDL is sent to address ADL, and is the MONITOR_CODE.

When a control message (ADH = 80 hex + dataset address), or a monitor request (ADH = 0 + dataset address) is received, ADL is used as a pointer into table 1; CONTROL_CODE is extracted for a control request, and MONITOR_CODE is extracted for a monitor. The translation of these codes is given in tables 2 and 3.

To inhibit a specific control point or monitor point the msb of the _CODE is set to zero.

3.4 Read the decoding registers.

ADH = 40 hex + datast address

ADL spans the range 0 to FF hex, as in 3.3

MONH and MONL contain the contents of the decoding register specified by ADL.

Table 1

List of the set-up command messages which must be sent to a dataset before it can respond to control/monitor requests. These registers are in non-volatile RAM, so this table should not need to be re-loaded.

a. Communication with points external to the dataset

SYNC	ADH	ADL	CMDH	CMDL	
16h	Cxh	00h	00h	81h	Implementing 64 Analog I/P monitor channels; Commands to these addresses are not not allowed.
16h	Cxh	01h	00h	81h	
..	..	..	..	..	
16h	Cxh	3Fh	00h	81h	
16h	Cxh	40h	82h	82h	Implementing 32 single-bit Control O/P and Monitor I/P lines
..	..	..	..	..	
16h	Cxh	5Fh	82h	82h	
16h	Cxh	60h	84h	84h	Implementing 64 addresses for 8-bit Control and Monitor data on the bidirectional bus
..	..	..	..	..	
16h	Cxh	9Fh	84h	84h	
16h	Cxh	A0h	88h	88h	Implementing 64 addresses for 16-bit Control and Monitor data on the bidirectional bus
..	..	..	..	..	
16h	Cxh	DFh	88h	88h	
16h	Cxh	E0h	90h	90h	Implementing 4 R/W lines for 8-bit Control and Monitor data on the bidirectional bus.
..	..	..	..	..	
16h	Cxh	E3h	90h	90h	
16h	Cxh	E4h	A0h	A0h	Implementing 4 R/W lines for 16-bit Control and Monitor data on the bidirectional bus.
..	..	..	..	..	
16h	Cxh	E7h	A0h	A0h	

b. Status registers: Communication within the dataset

16h	Cxh	E8h	E0h	C0h	RESET_COUNT
16h	Cxh	E9h	E0h	C0h	RESTART_ERRS
16h	Cxh	EAh	E0h	C0h	ABORT_ERRS
16h	Cxh	EBh	E0h	C0h	EXEC_ERRS
16h	Cxh	ECh	00h	00h	
16h	Cxh	EDh	00h	00h	
16h	Cxh	EEh	E0h	C0h	VALID_CMDS
16h	Cxh	EFh	E0h	C0h	VALID_MONS
16h	Cxh	F0h	00h	00h	
16h	Cxh	F1h	00h	00h	
16h	Cxh	F2h	00h	00h	
16h	Cxh	F3h	00h	00h	
16h	Cxh	F4h	00h	00h	
16h	Cxh	F5h	00h	C0h	LAST_CMD_ADL
16h	Cxh	F6h	00h	C0h	LAST_CMDH
16h	Cxh	F7h	00h	C0h	LAST_CMDL
16h	Cxh	F8h	00h	00h	
16h	Cxh	F9h	00h	00h	
16h	Cxh	FAh	00h	00h	
16h	Cxh	FBh	F0h	C0h	clear RESET flag
16h	Cxh	FCh	00h	C0h	Analog configuration
16h	Cxh	FDh	00h	E0h	WRITE_PROTECT_SWITCH
16h	Cxh	FEh	00h	C0h	DS serno.
16h	Cxh	FFh	D0h	C0h	ADL_RANGE check

Table 2

Decoding the CONTROL_CODES

(Message : ADH = 80 hex + dataset address; ADL, CMDH, CMDL)

00h	.. ignore; no action
82h	.. set control line number (ADL - 40h) : Line goes HIGH if CMDL is even; LOW if CMDL is odd.
84h	.. send CMDL to external address (ADL - 60h)
88h	.. send CMDH,CMDL to external address (ADL - A0h)
90h	.. send CMDL with the (ADL - E0h) WR line strobe
A0h	.. send CMDH, CMDL with the (ADL - E4h) WR strobe,
D0h	.. set ADL_RANGE_CHECK if CMDL is odd; clear ADL_RANGE_CHECK if CMDL is even.
E0h	.. clear the associated status register (at ADL + 1024, in non-volatile RAM)
F0h	.. clear the associated status register (ADL + 1024), and clear the RESET flag.

Table 3

Decoding the MONITOR_CODEs

(Message : ADH = 0 + dataset address; ADL)

00h .. ignore; no action

81h .. read analog channel number ADL; result (12-bit) in MONH, MONL

82h .. read monitor line number (ADL - 40h); result in lsb of MONL;
(MONL = 1 .. line LOW; = 0 .. line HIGH)

84h .. read from external address (ADL - 60h); send back to ACC as MONL

88h .. read from external address (ADL - A0h); return as MONH, MONL

90h .. read MONL with the (ADL - E0h) RD line strobe

A0h .. read MONH, MONL with the (ADL - E4h) RD strobe,

C0h .. read status register (at ADL + 1024); return in MONL

E0h .. reads the non-volatile RAM WRITE_PROTECT_SWITCH register; load value into (ADL + 1024), and return value in MONL.