SYDNEY
COORDINATED
ADAPTIVE
TRAFFIC
SYSTEM

TRAFFIC TECHNOLOGY BRANCH

SCATS
DATA FORMATS

RTA-TC-226

27 October 1999

Issue: (A)	1-August	1997
Issue: (B)	20-February	1998
Issue: (B.2)	31 July	1998
Issue: (C)	30 April	1999
Issue: (D)	27 June	1999
Issue: (E)	27 October	1999

Document No: RTA-TC-226

Roads and Traffic Authority of NSW
Transport Management Centre
Traffic Technology Branch
25 Garden Street
EVELEIGH, NSW 1430
AUSTRALIA

International:
Telephone 61 2 8396 1602
FAX 61 2 8396 1600

Local:
Telephone (02) 8396 1602
FAX (02) 8396 1600

Notes

SCATS REGION WALLMAP MESSAGES

A SCATS regional computer can be configured to supply data through a dedicated wallmap port to an external service. The port characteristics are the same as a standard SCATS terminal port with 8 bits, no parity.

The baud rate is as set in the system file **[1,2]SYSPARAM.DAT** (for a Micro/RSX operating system) or in file **[1,2]TERMINALS.CMD** (for an RSX11M operating system).

MESSAGE PACKET

Messages from the wallmap port on a regional computer are sent as packets. The packet format is:

81 Hex	*Start of packet*
Byte count	*Number of data bytes following*
Data…	*Variable length data area containing **one or more** messages*
Checksum	*The sum of all characters from the start of the message to the checksum (inclusive) is zero.*

DATA MESSAGES

Data messages are encapsulated within the message packet. A message packet is of variable length and may contain one or more data messages.

DATE AND TIME

The date and time message allows the map to receive the date and time to the nearest second. The message consists of 7 bytes, starting with an ID of decimal 255.

255	*Message identification (decimal)*
Year	*Year since 1900 to a maximum value of 127*
Month	*Month number within range of 1 to 12*
Day	*Day number with range of 1 to 12*
Hour	*Hour number within range of 0 to 23*
Minute	*Minutes within range of 0 to 59*
Seconds	*Seconds with range of 0 to 59*

REGION STATUS

The region status message contains a bit mask of the region status codes that are normally displayed on the top line of a SCATS workstation. It thus contains a summary of all alarms and conditions on the regional computer. It consists of 3 bytes:

254	*Message identification 254 decimal*		
data byte 1	*bit 0*	**UD**	*Failure to update RAM at an intersection*
	bit 1	**BSY**	*Disk write in progress (SYS.LX or INTFIL.DAT)*
	bit 2	**TIM**	*System time is incorrect*
	bit 3	**DET**	*An intersection has a detector alarm*
	bit 4	**LC**	*An intersection has a long Clearance alarm*
	bit 5	**MAJ**	*An intersection has a major alarm*
	bit 6	**LA**	*An intersection has a Lamp Failure*
	bit 7	**SC**	*An intersection Short Clearance*
data byte 2	*bit 0*	**FBK**	*A subsystem is on fallback*
	bit 1	**HID**	*A subsystem has a high density alarm*
	bit 2	**INF**	*A subsystem has an increment failure alarm*
	bit 3	**PL#**	*A plan lock is present*
	bit 4	**CL^**	*A subsystem has a cycle time locked*
	bit 5	**TMP**	*CPU over-temperature warning*
	bit 6	**TRM**	*An intersection plan has been trimmed*
	bit 7	**DWL**	*An intersection has been dwelled*

LAMP TEST

The lamp test messages allow an operator at a SCATS workstation to send a message to the map port for use in testing the display status of all indicators on the wallmap. The SCATS commands are **LT!** And **LT/** corresponding to a lamp on and a lamp off message. These are single byte messages.

253	*Lamp **on (LT!)** - 253 decimal*

252	*Lamp **off (LT/)** - 252 decimal*

INTERSECTION STATUS MESSAGE

The intersection status message is for a single intersection, identified by its SCATS slot number. It contains information relating to alarms, running mode, special status bits (MSS or Miscellaneous Status bits), and traffic conditions etcetera. Each message starts with a slot number ID.

SCATS 5 FORMAT

<table>
<tr><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th><th>Data</th></tr>
<tr><td colspan="8">Slot Number</td><td>ID byte (decimal 1 to 128)</td></tr>
<tr><td>M</td><td>F</td><td>FBK</td><td>+GR</td><td>< ></td><td colspan="3">current phase 1 to 7</td><td>Phase byte</td></tr>
<tr><td>PK</td><td>DWL</td><td>TRM</td><td></td><td></td><td></td><td>SI</td><td>MSS</td><td>Status byte</td></tr>
<tr><td>BO</td><td>FY</td><td>LF</td><td>MAJ</td><td>MIN</td><td>COM</td><td>DA</td><td></td><td>Alarms byte</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td rowspan="2">Optional MSS bits 1-16 if MSS “present” set</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="8">SI number (1 to 250)</td><td rowspan="2">Optional SI data if SI “present” bit set</td></tr>
<tr><td>1</td><td colspan="2">VK/VO</td><td colspan="5">DS/4</td></tr>
<tr><td colspan="8">Possible more SI data…</td><td>More SI data if bit 7 set in the DS byte of the preceding SI data</td></tr>
<tr><td colspan="8">SI number (1 to 250)</td><td rowspan="2">The last SI has a 0 in bit 7 of the DS byte</td></tr>
<tr><td>0</td><td colspan="2">VK/VO</td><td colspan="5">DS/4</td></tr>
</table>

SCATS 6 FORMAT

SCATS 6 can have 999 SIs compared to 250 in SCATS 5. As a result, and extra byte is required per SI. The SI number only requires 10 bits (the low 8 bits in the first byte and the top 2 bits in the first two bit positions of the next byte). The remaining 6 bits of the second byte contain the VK/VO data and a continuation bit if more SI data follows. The last byte for the SI contains the DS value.

<table>
<tr><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th><th>Data</th></tr>
<tr><td colspan="8">Slot Number</td><td>ID byte (decimal 1 to 250)</td></tr>
<tr><td>M</td><td>F</td><td>FBK</td><td>+GR</td><td>< ></td><td colspan="3">current phase 1 to 7</td><td>Phase byte</td></tr>
<tr><td>PK</td><td>DWL</td><td>TRM</td><td></td><td></td><td></td><td>SI</td><td>MSS</td><td>Status byte</td></tr>
<tr><td>BO</td><td>FY</td><td>LF</td><td>MAJ</td><td>MIN</td><td>COM</td><td>DA</td><td></td><td>Alarms byte</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td rowspan="2">Optional MSS bits 1-16 if MSS “present” set</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="8">SI number (1 to 999) – low 8 bits</td><td rowspan="3">Optional SI data if SI “present” bit set in the Status byte</td></tr>
<tr><td>1</td><td colspan="2">VK/VO</td><td colspan="4">Bits 2-4 unused</td><td>SI (bits 9-10)</td></tr>
<tr><td colspan="8">DS (0 to 255)</td></tr>
<tr><td colspan="8">Possible more SI data…</td><td>More SI data if bit 7 is set in the VK/VO byte of the preceding SI data</td></tr>
<tr><td colspan="8">SI number (1 to 999) – low 8 bits</td><td rowspan="3">The last SI has a 0 in bit 7 of the VK/VO byte</td></tr>
<tr><td>1</td><td colspan="2">VK/VO</td><td colspan="4">Bits 2-4 unused</td><td>SI (bits 9-10)</td></tr>
<tr><td colspan="8">DS (0 to 255)</td></tr>
</table>

INTERSECTION STATUS MESSAGE (continued)

The data in the Phase byte, Status byte and Alarms byte of the message to the Wallmap port is the same in SCATS 5 and SCATS 6, as is the VK/VO value returned.

Phase byte

M	*The intersection is running Masterlink*
F	*The intersection is running Flexilink **(no M or F = Isolated)***
FBK	*The intersection has been forced to fallback mode*
+GR	*The running phase is in the intergreen period*
< >	*The intersection is running the stretch phase*

Status byte

PK	*The intersection alarms are parked*
DWL	*The phase is being dwelled*
TRM	*The intersection plan has been manually trimmed*
SI	*SI data exists in this message*
MSS	*Miscellaneous data bits exist in this message*

Alarms byte

BO	*Lamps are blacked out*
FY	*Lamps are flashing yellow*
LF	*The intersection has a lamp fail alarm*
MAJ	*The intersection has a major alarm (BD,CK,FL,GT,GW,IR,PE,SF or UD)*
MIN	*The intersection has a minor alarm (CF,IH,LC,NF,OD,SC,SI or XU)*
COM	*Communications alarm present (DZ,NC,PF,ST or WD)*
DA	*The intersection has a detector alarm*

Strategic Input (VK/VO) data

DS/4	*Highest lane degree of saturation divided by four (limited to 127/4)*
VK/VO	*VK (reconstituted volume) divided by VO (counted volume) ratio value, where:*
0	*VK:VO ration is less than 2:1*
1	*VK:VO ratio is more than 2:1 but less than or equal to 3:1*
2	*VK:VO ratio is more than 3:1 but less than or equal to 4:1*

3	*VK:VO ratio is more than 4:1*

SENDING INTERSECTION STATUS MESSAGES

Intersection status messages are sent when:

- A new phase starts at an intersection
- An intersection's alarms change
- A intersection plan trim or lock goes on or off
- MSS bits change at an intersection
- A slot has its intersection number changed
- An intersection has its alarms parked
- AL/ clears any alarms at an intersection
- The region to CMS link fails

UPDATING ALL MESSAGES

Send an upper case **U** followed by a **carriage return** to the wallmap port to request an update of all messages on the region.

SCATS GRAPHICS SUBSYSTEM COLOURS

COLOUR	VALUE	CONDITIONS	MEANING
White	**0**	*CL < SCL and RL < XCL-5* **or** *SS has no LM=M intersections*	*Low CL, light traffic*
Green	**1**	*XCL > CL => SCL*	*Medium CL, light traffic*
Yellow	**2**	*CL >= XCL and RL < XCL-5*	*High CL, medium traffic*
Orange	**3**	*CL >= XCL and RL >= XCL-5 and VK/VO <= 2.4*	*High CL, heavy traffic*
Blue	**4**	*CL < SCL and RL >= XCL-5* **or** *Increment Fail*	*Low CL, heavy traffic*
Red	**7**	*CL >= XCL and RL >= XCL-5 and VK/VO > 2.4 (Hi Density)*	*High CL, congested traffic*

- Note that FALLBACK can be set for all conditions except if SS has no LM=M intersections.
- SCL is the highest SCL value for a subsystem (LCL is used if SCL is undefined).
- HCL is used instead of XCL if XCL is not defined.

SYSTEM MONITOR DATA

If a regional computer is connected to a CMS system, System Monitor or SM data collected on the VAX will be preceded by a comment record. A comment record contains a description of the collection period and the susbsystem number for which data was collected.

Byte	DATA	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1-4	ID	A comment record has a zero in the first 4 bytes							
5	Type	^S for SM data *(^V for VS data)*							
6-11	Region	Region name in ASCII							
12	Year	Start year (since 1900)							
13	Month	Start month							
14	Day	Start day							
15	Year	End year (since 1900)							
16	Month	End month							
17	Day	End day							
18	Hour	Daily period 1 - start hour							
19	Minutes	Daily period 1 - start minutes							
20	Hour	Daily period 1 - end hour							
21	Minutes	Daily period 1 - end minutes							
22	Hour	Daily period 2 - start hour							
23	Minutes	Daily period 2 - start minutes							
24	Hour	Daily period 2 - end hour							
25	Minutes	Daily period 2 - end minutes							
26	SS	First subsystem number							
…	*Additional SS numbers*	*The following bytes from byte 27 up to byte 88 can contain more subsystem numbers. (the maximum subsystem number is 63).* • *A positive SS number means that a full SM was collected for that SS.* • *A negative SS number means that only the SM header (no SA or LK data) was collected.* • *A zero SS number terminates the SS numbers collected.*							
128	0	Last byte is always zero - only 1 comment record for SM data							

SM HEADER - OLD FORMAT

This format is used on all PDP SCATS 1 regions.

Bit 7	*Bit 6*	*Bit 5*	*Bit 4*	*Bit 3*	*Bit 2*	*Bit 1*	*Bit 0*	*byte*
1	**0**	**SS no. (0-63)**						***1***
FBK	Year since 1900 (0-127)							***2***
ISS	LP#	CL^	SP#	Month (1-12)				***3***
FTP CL	FTP LP	FTP SP	Day (1-31)					***4***
VF'	VF"	DV#	Hour (0-23)					***5***
M-	M+	Mins (0-59)						***6***
1	SM=HD		SA mode			SK=SV	SK=VO	***7***
Married	Scats V4	LP (0-3=plan 1-4)			SP (0-7=plans 1-8)			***8***
LP vote		ISS SP vote (no ISS = bits 0-2 only)						***9***
Nominal CL								***10***
Actual CL (includes rotation)								***11***
RL								***12***
Representative SA								***13***
DS (SCATS v5.x only)								***14***

SA MODE

This mode is in byte 7, bits 2, 3 and 4.

Bit 4	*Bit 3*	*Bit 2*	*Value*	*Meaning*
0	0	0	0	*VOVK*
0	0	1	1	*CVK*
0	1	0	2	*GVK*
0	1	1	3	*VK*

Bit 4	*Bit 3*	*Bit 2*	*Value*	*Meaning*
1	0	0	4	*unused*
1	0	1	5	*CVO*
1	1	0	6	*GVO*
1	1	1	7	*VO*

SA VOTE TITLE

Byte 7 also determines the type of SA vote data.

Bit 7	*Bit 1*	*Bit 0*	*Title*	*Meaning*
1	0	0	ADS	*Average DS*
1	1	0	AVK	*Average VK*
1	1	1	AVO	*Average VO*

SM HEADER - NEW FORMAT

This is the format of an System Monitor record collected on a CMS system from a SCATS 6 region.

If System Monitor data is collected **locally** on a SCATS 6 region, each record will be preceded by a two byte record length (low byte first). If the record length is **odd**, the record will contain a trailing null character as padding to make the overall record length **even**!

Bit 7	*Bit 6*	*Bit 5*	*Bit 4*	*Bit 3*	*Bit 2*	*Bit 1*	*Bit 0*	*byte*
0								***1***
SS no.								***2***
FBK	Year since 1900 (0-127)							***3***
ISS	LP#	CL^	SP#	Month (1-12)				***4***
FTP CL	FTP LP	FTP SP	Day (1-31)					***5***
VF’	VF”	DV#	Hour (0-23)					***6***
M-	M+	Mins (0-59)						***7***
1	SM=HD		SA mode			SK=SV	SK=VO	***8***
Married	Scats V4	LP (0-3=plan 1-4)			SP (0-7=plans 1-8)			***9***
LP vote		ISS SP vote (no ISS = bits 0-2 only)						***10***
Nominal CL								***11***
Actual CL (includes rotation)								***12***
RL								***13***
Representative SA (low byte)								***14***
Representative SA (high byte)								***15***
DS (SCATS v5.x only)								***16***

SA MODE

This mode is in byte 8, bits 2, 3 and 4.

Bit 4	*Bit 3*	*Bit 2*	*Value*	*Meaning*
0	0	0	0	*VOVK*
0	0	1	1	*CVK*
0	1	0	2	*GVK*
0	1	1	3	*VK*

Bit 4	*Bit 3*	*Bit 2*	*Value*	*Meaning*
1	0	0	4	*unused*
1	0	1	5	*CVO*
1	1	0	6	*GVO*
1	1	1	7	*VO*

SA VOTE TITLE

Byte 8 also determines the type of SA vote data.

Bit 7	*Bit 1*	*Bit 0*	*Title*	*Meaning*
1	0	0	ADS	*Average DS*
1	1	0	AVK	*Average VK*
1	1	1	AVO	*Average VO*

SA/LK DATA - OLD FORMAT

This format is produced by all PDP 11 SCATS systems for the SA and the LK data record. The data is for an SA if bit 5 in byte 5 is clear, else it is for an LK.

Bit 7	***Bit 6***	***Bit 5***	***Bit 4***	***Bit 3***	***Bit 2***	***Bit 1***	***Bit 0***	***Byte***
1	1	0	1=VPH	Bit mask for lanes in use				***1***
Intersection no. (Low byte)								***2***
Intersection no. (high byte)								***3***
SA or LK number								***4***
SA flags		LK	SA mode			VF'/LQ	VF"	***5***
SG	Phase bit mask (or signal group in bits 0-4 if SG set)							***6***
Phase time								***7***
DS for first lane of lane bitmap - if >200 show *nnn, if >100 show >nnn else show !nnn								***8***
Either 1 or 2 byte volume for first lane of lane bitmap depending on "SA mode"								***9-***
The number of sets of DS and volume data in the message is determined by the bit mask for lanes in use in byte 1 (4 lanes maximum)								
Volume for last lane used								
Vote value								***n***

SA/LK DATA - NEW FORMAT

This is the format of an SA or LK record collected on a CMS system from a SCATS 6 regional computer. The data is for an SA if bit 5 in byte 5 is clear, else it is for an LK.

If System Monitor data is collected **locally** on a SCATS 6 region, each record will be preceded by a two byte record length (low byte first). If the record length is **odd**, the record will contain a trailing null character as padding to make the overall record length **even**!

Bit 7	*Bit 6*	*Bit 5*	*Bit 4*	*Bit 3*	*Bit 2*	*Bit 1*	*Bit 0*	*Byte*
0	1	0	1=VPH	Bit mask for lanes in use				***1***
Intersection no. (Low byte)								***2***
Intersection no. (high byte)								***3***
SA or LK number (Low byte)								***4***
SA or LK number (High byte)								***5***
SA flags		LK	SA mode			VF'/LQ	VF"	***6***
SG	Phase bit mask (or signal group in bits 0-4 if SG set)							***7***
Phase time								***8***
DS for first lane of lane bitmap - if >200 show *nnn, if >100 show >nnn else show !nnn								***9***
Either 1 or 2 byte volume for first lane of lane bitmap depending on "SA mode"								***10…***
The number of sets of DS and volume data in the message is determined by the bit mask for lanes in use in byte 1 (4 lanes maximum)								
Volume for last lane used								
Vote value								***n***

SA FLAGS

Byte 6 (new format) or byte 5 (old format), contain the SA flags in bits 6 and 7 which have the following meaning:

Bit 7	*Bit 6*	*Value*	*Display*	*Meaning*
0	**0**	**0**	**S**	**SA controls both CL and SP plan voting**
0	**1**	**1**	**#**	**SA controls CL only**
1	**0**	**2**	**^**	**SA controls SP plan vote only**
1	**1**	**3**	*****	**SA controls nothing (for monitoring)**

SA MODE

Byte 6 (new format) or byte 5 (old format), also contains the SA volume mode in bits 2 to 4, which determines how many volume bytes each lane has, using the following table.

Bit 4	*Bit 3*	*Bit 2*	*Value*	*Meaning*	*Number of bytes*
0	**0**	**0**	**0**	***VOVK***	*1 byte VO, 1 byte VK*
0	**0**	**1**	**1**	***CVK***	*2 bytes (low byte first)*
0	**1**	**0**	**2**	***GVK***	*2 bytes (low byte first)*
0	**1**	**1**	**3**	***VK***	*1 byte*
1	**0**	**0**	**4**	***unused***	-
1	**0**	**1**	**5**	***CVO***	*2 bytes (low byte first)*
1	**1**	**0**	**6**	***GVO***	*2 bytes (low byte first)*
1	**1**	**1**	**7**	***VO***	*1 byte*

DISPLAYING DS VALUES

The lane separator on an SM display is normally an exclamation mark. However, this changes depending on the value of the DS. If DS > 200, use an asterisk (eg. *223). If the DS is between 101 and 200, use a greater than symbol (eg. >123) else use the exclamation mark (eg. ! 78).

DISPLAYING THE VOTE VALUE

The last byte of the message contains the vote value, the last item displayed on an SM data line.

- If the VPH bit is clear in byte 1, the value is displayed as is.
- If the VPH bit is set, convert the value to vehicles per hour by multiplying the value by 20 to convert from a flow in three minutes.

PLAN 0 DATA

This message is only sent if the SM is from a subsystem running ISS mode.

The number of bytes in the message varies with the number of phases at the critical intersection, and ranges from 3 bytes for a 2 phase intersection, up to 8 bytes for a 7 phase intersection.

OLD FORMAT

This is the format from a PDP 11 region.

Bit 7	*Bit 6*	*Bit 5*	*Bit 4*	*Bit 3*	*Bit 2*	*Bit 1*	*Bit 0*	*Byte*
1	**1**	**1**	**0**	**0**	**0**	**0**	**0**	***1***
A phase time								***2***
B phase time								***3***
C phase time (if message has at least 4 bytes)								***4***
D phase time (if message has at least 5 bytes)								***5***
E phase time (if message has at least 6 bytes)								***6***
F phase time (if message has at least 7 bytes)								***7***
G phase time (if message has 8 bytes)								***8***

NEW FORMAT

This is the format of a plan 0 record collected on a CMS system from a SCATS 6 region. Bit 7 in the first byte is zero for consistency with all other SM records from a SCATS 6 regional computer.

If System Monitor data is collected **locally** on a SCATS 6 region, each record will be preceded by a two byte record length (low byte first). If the record length is **odd**, the record will contain a trailing null character as padding to make the overall record length **even**!

Bit 7	*Bit 6*	*Bit 5*	*Bit 4*	*Bit 3*	*Bit 2*	*Bit 1*	*Bit 0*	*Byte*
0	**1**	**1**	**0**	**0**	**0**	**0**	**0**	***1***
A phase time								***2***
B phase time								***3***
C phase time (if message has at least 4 bytes)								***4***
D phase time (if message has at least 5 bytes)								***5***
E phase time (if message has at least 6 bytes)								***6***
F phase time (if message has at least 7 bytes)								***7***
G phase time (if message has 8 bytes)								***8***

EXPLANATION OF PHASE TIME

If bit 7 is set in the phase time (ie. the unsigned byte value is equal to or greater than 128), it signifies the stretch phase (<>). Subtract 128 from the value and display the remainder inside angled brackets. Eg. A value of 163 gives <35>

If the value is less than 128, but exceeds 80, it is for a fixed time phase (ie. a phase time in seconds). Subtract 80 from the byte value and display the remainder, preceded by a cross hatch. Eg. A value of 92 gives #12.

Values of 80 or less are percentage times, displayed as is.

SAMPLE SM DISPLAY

This a sample system monitor for subsystem 12.

```
18:18 SS12M+ PL8.2  PV19.2 CL 110-03 RL106'SA 42 DS135
  INT SA/LK PH   PT! DS VO VK! DS VO VK! DS VO VK! DS VO VK!ADS
  253 S 33 ' 1   31! 98 14 15! 65  9 10!  -     -!  -     -! 69
  253 S 38 ' 2   33! 51  5  6>104 10 14! 68  7 10!  -     -! 73
  253 S 43^  3   11!  0  0  0!  -     -!  -     -!  -     -! 31
  253 S 62^  4   11! 44  2  2!  -     -!  -     -!  -     -! 36
  253 S 86 'D    32! 45  8  6! 54  4  7! 42  7  6!  -     -! 52
  253 L 35^' 1   31! 98 14 15! 65  9 10!  -     -!  -     -! 860
  253 L 40^  2   33!  -     ->104 10 14! 68  7 10!  -     -! 600
    A=<34>  B=13  C= 1  D=40  E=13
```

SCATS SERIAL COMMUNICATIONS PROTOCOL

A SCATS regional computer can be configured to send and receive data through a dedicated port to an external service such as **TRITRAM** (which supplies simulated traffic data) or **BIPS** (the Bus Information and Priority System).

The port characteristics are the same as a standard SCATS terminal port with 8 bits, no parity.

The baud rate is as set in the system file **[1,2]SYSPARAM.DAT** (for a Micro/RSX operating system) or in file **[1,2]TERMINALS.CMD** (for an RSX11M operating system).

MESSAGE PACKET

Messages from the port on a regional computer are sent as packets. The packet format is:

SOH	*Start of header*
Byte count	*Number of data bytes following (255 maximum)*
Data…	*Variable length data area containing **one or more** messages*
Checksum	*The sum of all characters from the start of the message to the checksum (inclusive) is zero.*

START OF HEADER CODE

The message protocol caters for message acknowledgement and retransmission of lost messages. The lower two bits of the SOH determine the message number and the ACK code as follows. This allows acknowledgement as part of the next data message.

SOH	*BIT 1 message no.*	*BIT 0 ACK*
80H	0	0
81H	0	1
82H	1	0
83H	1	1

When initiating communications, the following points should be taken in account.

At startup, SCATS initialises it "**last message received**" to zero. This means that the first message that SCATS can process is an 82H or 83H message. If the remote program starts by sending an 80H or 81H message, SCATS will only acknowledge the receipt of the message but will not send any requested data. This is because, SCATS assumes that it has previously received and processed this message type.

To guarantee synchronisation at startup, it is recommended to send off an 80H message with no data. That is, a 3 byte message of 80H 00H 80H. SCATS will reply with a an ACK of 04H. The next message from the remote program must then be an 82H. SCATS will reply with an ACK of 05H.

From then on, subsequent messages from the remote program must then toggle the message bit and set the appropriate ACK bit for messages to remain synchronised.

Note that if SCATS receives a message with the same message number as the last message received, SCATS will always reply with an ACK and will not process the message.

Also note that SCATS may first send an ACK prior to sending data if transmission of the data is delayed in SCATS.

ACK/NAK MESSAGES

Alternatively, if no data needs be sent, an ACK message will suffice.

ACK	Checksum
NAK	Checksum

The ACK code for a message with an SOH of 80H or 81H is 4.

The ACK code for a message with an SOH of 82H or 83H is 5.

The NAK code for a message with an SOH of 80H or 81H is 15H

The NAK code for a message with an SOH of 82H or 83H is 14H

If data is ready to be sent in the reverse direction, ACK need not be sent. Instead, BIT0 of the data message SOH is an implied ACK. If neither ACK, NAK or a reply message is received within a timeout period of about 2 seconds, the data message will be retransmitted.

DATA MESSAGES

Data messages are encapsulated within the message packet. A message packet is of variable length and may contain one or more data messages. An ID byte of 63 indicates more data to follow.

SCATSIM SERIAL COMMUNICATIONS

INTERSECTION DATA MESSAGE

SCATS TO SIMULATION

Item	*Size*	*Remarks*
Slot number	1 byte	bits 0-5 = slot number in range 1-32 bit 6 = miscellaneous flags follow bit 7 = signal group status follows (2 bytes - low byte first)
Signal groups	2 bytes	bit 0 = group 1, bit 1 = group 2 etc
Miscellaneous flags	1 byte	bit 0 = FVO data follows (2 bytes of detector flags) bit 1 = BVO data follows (2 bytes of detector flags)
Next slot…		

Worst case is 8 data bytes per slot. If all data to send will not fit in a single message, the last byte of data will be an ID of 63 indicating more to follow.

SIMULATION TO SCATS

VOLUME/OCCUPANCY DATA

Item	*Size*	*Remarks*
Slot number	1 byte	bits 0-5 = slot number within range 1-32 bit 6 = vol/occ data follows (3 bytes per detector) bit 7 = pedestrian demands follow
Detector status	2 bytes	16 bits for 16 vehicle detectors
Pedestrian demands	1 byte	8 bits for 8 push buttons
Volume/occupancy	3 bytes	byte 1 = detector number 1-16 - bit 7 set=more detectors follow byte 2 = low byte of occupancy byte 3 = bits 0-1 high 2 bits of occupancy, bits 2-7=volume
Vol/occ for next detector…		Present only if the continuation bit 7 is set in byte 1 of the preceding vol/occ data.

TIME FROM SIMULATION

Message 62	byte 1
Hour	byte 2
Minutes	byte 3
Seconds	byte 4

BUS PRIORITY SYSTEM

The SCATS serial communications protocol must be used for communicating with the Bus Information and Priority System (BIPS).

The data messages are as follows:

Messages from BIPS to SCATS

Request for Intersection data

Data	*Size*	*Remarks*
Message type 1	1 byte	request for intersection data
Sequence number	1 byte	Number of this request
Flags	1 byte	set bit 0 to request CL and master subsystem no. set bit 1 to request Offset, CG, no. of phases and plan data
Intersection count	1 byte	number of intersections in this message
Intersection no.	2 bytes	low byte of desired intersection no. high byte of desired intersection no.
Append 2 byte intersection no. for each additional intersection		

Replace remaining phase time

This has the effect of making the current phase run longer or shorter than the time calculated by SCATS.

Data	*Size*	*Remarks*
Message type 2	1 byte	replace remaining phase time
Intersection count	1 byte	number of intersections in this message
Intersection no.	2 bytes	low byte of intersection no. high byte of intersection no.
Running phase (in case original phase has ended)	1 byte	bits 0 to 2 - phase (1=A…7=G). bit 7 set for "***No gapping allowed***"
New time until phase ends	1 byte	replaces current value if phase has not terminated. zero = leave existing value untouched - allows application of "no gap" by itself if necessary.
Append 2 byte intersection no. and 2 byte data for each additional intersection		

Replace plan data

This allows full control over an intersection.

Data	*Size*	*Remarks*
Message type 3	1 byte	replace plan data
Intersection count	1 byte	number of intersections in this message
Intersection no.	2 bytes	low byte of intersection no. high byte of intersection no.
No. of phases	1 byte	bits 0-2 = phase bit 7 = lock the plan
New plan data	2 bytes	the following data is present for each phase in ascending order of phase (ie ABC…): low byte: bits 0-6 = phase split bit 7 = NG (no gap) high byte: bits 0-2 = next phase in sequence (NIS) bit 3 = NS (no skip) bit 4 = AS (CL dependant permanent demand) bit 5 = TG (time gain) bit 6 = FS (CL dependant forced skip) bit 7 = FG (false green) NB: **NS** and **FS** both set = permanent demand (**PD**)
Data for next phase…		
Data for next intersection…		

APPLY A DWELL

A dwell forces an intersection to the nominated phase where it will rest until the dwell is removed.

Data	*Size*	*Remarks*
Message type 4	1 byte	apply a dwell
Intersection count	1 byte	number of intersections in this message
Intersection no.	2 bytes	low byte of intersection no. high byte of intersection no.
Phase to be dwelled	1 byte	bits 0-2
Next intersection number and phase to be dwelled…		

REMOVE A DWELL

Data	*Size*	*Remarks*
Message type 5	1 byte	remove a dwell
Intersection count	1 byte	number of intersections in this message
Intersection no.	2 bytes	low byte of intersection no. high byte of intersection no.
Phase	1 byte	bits 0-2
Next intersection number and phase to have a dwell removed…		

RELOAD PLAN DATA

This message asks SCATS to reload the active plan with the data from the currently selected stored plan and is used after BIPS has sent down new plan data that has now been finished with. This ensures that the phase flags are restored as per the SCATS plan data.

Data	*Size*	*Remarks*
Message type 6	1 byte	reload plan data
Intersection count	1 byte	number of intersections in this message
Intersection no.	2 bytes	low byte of intersection no. high byte of intersection no.
Next intersection number		

Messages from SCATS to BIPS

INTERSECTION DATA

Data	*Size*	*Remarks*
Message type 1	1 byte	intersection data - *reply to a type 1 request*
Sequence number	1 byte	This is the same number as sent by BIPS
Flags	1 byte	bit 0 = CL and Master Subsystem no. is included in message bit 1 = Offset, CG, no. of phases and plan data is included
Intersection count	1 byte	number of intersections in this message (3 maximum)*)*
Intersection no.	2 bytes	low byte, high byte (range 1 to 64999)
Running phase data	1 byte	bits 0-2 = running phase (1=A…7=G) bits 3-5 = phase step (0=LST, 1=MIN, 2=VIG, 3=GRN, 4=ECG, 5=R/Y, 6=YEL, 7=RED bit 6 = the phase is dwelled bit 7 = the phase cannot gap
Next scheduled phase	1 byte	bits 0-2 = next phase scheduled to run (ie demanded) 0 means no demands
Remaining green time for phase	1 byte	range: 0 to 255 (excludes intergreen - eg **YEL**+**RED** and is the time until SCATS instructs the intersection to terminate the current phase). **Zero means phase is terminating.**
Cycle length	1 byte	range: 20 to 240 ***(includes rotation)***
Master Subsystem no.	1 byte	range: 0 to 63
Plan Offset	2 bytes	**low byte**: offset from CG=0 (in seconds, signed) **high byte**: bits 0-2 = phase that ends at that offset bit 3 = this is a slave intersection - no offset bit 4 = intersection is double cycling
Cycle generator step	1 byte	range: 0 to 99
No. of phases	1 byte	bits 0-2 = no. of phases in plan - range: 2-7
Running plan data	2 bytes	the following data is present for each phase in ascending order of phase (ie ABC…): low byte: bits 0-6 = phase split bit 7 = NG (no gap) high byte: bits 0-2 = next phase in sequence (NIS) bit 3 = NS (no skip) bit 4 = AS (CL dependant permanent demand) bit 5 = TG (time gain) bit 6 = FS (CL dependant forced skip) bit 7 = FG (false green) NB: **NS** and **FS** both set = permanent demand (**PD**)
Data for next phase…		

Data for next intersection…

USE OF MESSAGES FROM SCATS TO BIPS

Determining the Offset between two Intersections

The data items in the message from SCATS to BIPS that are used to determine offset are, for each of the pair of intersections:-

- master subsystem number
- cycle length
- plan offset
- cycle generator step

1. Determine whether the two intersections have a defined offset ie are coordinated.
   - if both master subsystems are zero, there is NO defined offset
   - if the master subsystem numbers for the two intersections are the same (but not zero), there is a defined offset
   - if the master subsystem numbers for the two intersections are NOT the same, there is NO defined offset
2. Determine whether the offset is has reached stability ie no rotation:-
   - if the cycle lengths for the two intersections are the same, the offset is stable
   - if the cycle lengths for the two intersections are NOT the same, the offset is NOT stable
   - NB cycle lengths which differ by less than 3 seconds can be considered the same.
3. Determine whether an offset can be calculated. An offset CANNOT be calculated:-
   - if there is no defined offset (step 1), or
   - if there is no (or incomplete) plan offset data, or
   - if the intersection is a slave (bit 3 set)
   - if one or both values of CL are zero (intersection probably not Masterlink)
4. Calculate the offset in seconds. Notation is:-

   PP1 is plan offset for the first intersection

   CG1 is the cycle generator step for the first intersection

   PP2 is plan offset for the second intersection

   CG2 is the cycle generator step for the second intersection

   CL is the common cycle length (average if not exactly equal)

   - calculate the difference in the zero points of the two cycle generators by
     DIFF = (CG1 - CG2)*CL/100
   - calculate the offset as
     PP1 - PP2 - DIFF

The sign of the offset has the following meaning

- if the sign is **POSITIVE** it means that the specified phase at intersection 1 ends **AFTER** the specified phase at intersection 2
- if the sign is **NEGATIVE** it means that the specified phase at intersection 1 ends **BEFORE** the specified phase at intersection 2

EXAMPLE 1

CG1 = 30 PP1 = 24 C CL1 = 119

CG2 = 25 PP2 = 24 B CL1 = 121

Average CL = 120

DIFF = (CG1 - CG2)*CL/100

= (30- 25)*120/100

= 36 - 30 = 6

Offset = PP1 - PP2 - DIFF

= 24 - 24 - 6

= -6

ie the end of C phase at intersection 1 is 6 seconds before the end of B phase at intersection 2

EXAMPLE 2

CG1 = 25 PP1 = 24 C CL1 = 80

CG2 = 30 PP2 = -6 B CL1 = 80

Average CL = 80

DIFF = (CG1 - CG2)*CL/100

= (25 - 30)*80/100

= 20 - 24 = -4

Offset = PP1 - PP2 - DIFF

= 24 - (-6) - (-4)

= +34

ie the end of C phase at intersection 1 is 26 seconds after the end of B phase at intersection 2

TRAFFIC INCIDENT MANAGEMENT SYSTEM (TIMS)

TIMS TO VAX

The TIMS link software sends and receives messages via mailboxes to the SMCOLLECT program for requesting SCATS SM (System Monitor) data.

Program SMCOLLECT checks for the existence of the logical name SCATS_TIMS_ENABLED. If found, SMCOLLECT attempts connection to the TIMS link message channel.

REQUEST FOR SYSTEM MONITOR

Data	*Byte*	*Remarks*
^S	1	Message request code
Interval	2	Auto repeat time in minutes (unsigned) - 0=single shot
Region name	3-8	ASCII
Subsystem mask	9-16	64 bits - masks for SS 1-63

Note: To terminate auto repeat requests, send a non zero interval with a zero SS bitmask.

REQUEST FOR INTERSECTION STATUS

Data	*Byte*	*Remarks*
^N	1	Message request code
1	2	
Region name	3-8	*Optional! ASCII region name - If no region, then ALL regions*

REQUEST FOR UNUSUAL CONGESTION

Data	*Byte*	*Remarks*
^U (15Hex)	1	Message request code
0	2	NUL
Region name	3-8	*Optional - ASCII - if not used, set field to blanks*
Intersection number	9-10	Low byte of intersection number comes first.

TIMS SHUTDOWN

Data	*Byte*	*Remarks*
^Z	1	Shutdown message
State	2	1=OK, 0=FAULTY

VAX TO TIMS

INITIALISATION

Sent to the TIMS link software.

Data	*Byte*	*Remarks*
^I	1	Control I
^S	2	Control S
NULL	3	0
NULL	4	0

UNUSUAL CONGESTION DATA

Data	*Byte*	*Remarks*
^U	1	(hexadecimal 15)
Date/time	2-3	bits 0-5 = minutes bits 6-10 = hour bits 11-15 = day
Number of reports	4	0 = No report, bit 7 set to indicate one or more packets following
Region Name	5-10	(ASCII)
Subsystem	11	
Strategic Approach	12	
Intersection number	13-14	Low byte of intersection first
Direction	14	Direction (ASCII, 'N','S','E','W','X'=Combined)
Alarm Duration	16	in minutes
Congestion	17-20	byte 1 = minutes 1-8 byte 2 = minutes 9-16 byte 3 = minutes 17-24 byte 4 = minutes 25-39
Next set of data. Byte 5 to 20 are repeated for each unusual congestion report. Up to 8 reports can be transmitted with each packet (max length of 140 bytes).		

INTERSECTION STATUS DATA

This message is a snapshot of an intersection showing the operating status. It is sent when an alarm state or an operating state changes. It contains flags showing items such as:

- operating **mode** (0-7) where:

0=	Isolated	**4=**	Police Off
1=	Flexilink	**5=**	Police Manual
2=	Masterlink	**6=**	Police Red
3=	Hurry call	**7=**	Maintenance

- LM# is set if the operating mode (link mode) is locked (M#, F# or I#)
- running phase (1=A, 7=G)
- phase step where:

0=	**LST**	Late Start	**4=**	**ECG**	Early cut-off Green
1=	**MIN**	Minimum Green	**5=**	**YEL**	Yellow
2=	**VIG**	Variable Initial Green	**6=**	**RED**	Red
3=	**GRN**	Rest	**7=**	**R/Y**	Rd/Yellow

- active Special Facilities: Z- Z+ Y- Y+
- dwell state
- intersection alarms, vehicle detector alarms (1-24) and pedestrian push button alarms (1-8)
- running intersection plan (IP) 0-8 where 0=plan 0 trim - 15=plan 0 lock (PL0#)
- running offset plan (PP) 0-4 where 0=PP0 trim - 7=PP 0 lock (PP0#)
- subsystem locks is set if a plan locks (PM#) and/or a cycle lock (CL^) is active
- subsystem alarms: HID=High Density, FBK=Fallback
- FTP shows Fixed Time Plans are active
- clock alarm (0=none) - error range is +/-59 seconds - (‘80’x means LARGE error)

INTERSECTION STATUS RECORD

Byte	DATA	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1	Header	201 octal							
2	Byte count	16 decimal							
3	Intersection	intersection number *(low byte)*							
4	number	intersection number *(high byte)*							
5	Plans	SS locks	PP			IP			
6	Mode	mode *(see above)*				Z-	Z+	Y-	Y+
7	Phase step	v4.xx	v5.xx	phase		phase step			
8	Next phase	Dwell	HID	FTP	FBK	LM#	phase called		
9		PK	ST	SC	LC	OD	BO	WD	PF
10	Alarms		NC	FY	FL	UD	DA	BD	CK
11		CE	PE	SY	IV	NF	CF	GW	SF
12		DZ	IH	XU	SI	LF	GT	IR	
13	Detector	detector alarms *(1-8) bit 0=1*							
14	alarms 1-16	detector alarms *(9-16) bit 0=9*							
15	PB 1-8	push button alarms *(1-8) bit 0=1*							
16	CA+/-59	clock error *(in secs)* (‘80’x = CA**)							
17	DA 17-24	detector alarms *(17-24) bit 0=17*							
18	CL	cycle length							
19	Checksum	checksum *(negated sum of bytes 1-18)*							

SYSTEM MONITOR DATA FOR TIMS

SM HEADER

Bit 7	*Bit 6*	*Bit 5*	*Bit 4*	*Bit 3*	*Bit 2*	*Bit 1*	*Bit 0*	*byte*
^S record type								***1***
1	0	SS no. (0-63)						***2***
region name (with trailing blanks)								***3-8***
FBK	Year since 1900 (0-127)							***9***
ISS	LP#	CL^	SP#	Month (1-12)				***10***
FTP CL	FTP LP	FTP SP	Day (1-31)					***11***
VF'	VF"	DV#	Hour (0-23)					***12***
M-	M+	Mins (0-59)						***13***
1	SM=HD		SA mode			SK=SV	SK=VO	***14***
Married	Scats V4	LP (0-3=plan 1-4)			SP (0-7=plans 1-8)			***15***
LP vote		ISS SP vote (no ISS = bits 0-2 only)						***16***
Nominal CL								***17***
Actual CL (includes rotation)								***18***
RL								***19***
Representative SA								***20***
DS (SCATS v5.x only)								***21***

SA MODE

This mode is in byte 14, bits 2, 3 and 4.

Bit 4	*Bit 3*	*Bit 2*	*Value*	*Meaning*
0	0	0	0	*VOVK*
0	0	1	1	*CVK*
0	1	0	2	*GVK*
0	1	1	3	*VK*

Bit 4	*Bit 3*	*Bit 2*	*Value*	*Meaning*
1	0	0	4	*unused*
1	0	1	5	*CVO*
1	1	0	6	*GVO*
1	1	1	7	*VO*

SA VOTE TYPE

Byte 14 also determines the type of SA vote data.

Bit 7	*Bit 1*	*Bit 0*	*Title*	*Meaning*
1	0	0	ADS	*Average DS*
1	1	0	AVK	*Average VK*

1	1	1	AVO	*Average VO*

SA/LK DATA

This is format for the SA and the LK data record. The data is for an SA if bit 5 in byte 12 is clear, else it is for an LK. ***Note that LK data is NOT sent to TIMS!***

Bit 7	***Bit 6***	***Bit 5***	***Bit 4***	***Bit 3***	***Bit 2***	***Bit 1***	***Bit 0***	***Byte***
^S record type								***1***
1	1	0	1=VPH	Bit mask for lanes in use				***2***
region name (with trailing blanks)								***3-8***
Intersection no. (Low byte)								***9***
Intersection no. (high byte)								***10***
SA or LK number								***11***
SA flags		*LK*	SA mode			VF'/LQ	VF"	***12***
SG	Phase bit mask (or signal group in bits 0-4 if SG set)							***13***
Phase time								***14***
DS for first lane of lane bitmap - if >200 show *nnn, if >100 show >nnn else show !nnn								***15***
Either 1 or 2 byte volume for first lane of lane bitmap depending on "SA mode"								***16…***
The number of sets of DS and volume data in the message is determined by the bit mask for lanes in use in byte 2 (4 lanes maximum)								
Volume for last lane used								
Vote value								***n***

SA FLAGS

Bits 6 and 7 in byte 12 contain the SA flags which have the following meaning:

Bit 7	*Bit 6*	*Value*	*Display*	*Meaning*
0	**0**	**0**	**S**	**SA controls both CL and SP plan voting**
0	**1**	**1**	**#**	**SA controls CL only**
1	**0**	**2**	**^**	**SA controls SP plan vote only**
1	**1**	**3**	*****	**SA controls nothing (for monitoring)**

SA MODE

Byte 12, bits 2 to 4, define the SA volume mode, which in turn defines how many volume bytes each lane has, using the following table.

Bit 4	*Bit 3*	*Bit 2*	*Value*	*Meaning*	*Number of bytes*
0	**0**	**0**	**0**	***VOVK***	*1 byte VO, 1 byte VK*
0	**0**	**1**	**1**	***CVK***	*2 bytes (low byte first)*
0	**1**	**0**	**2**	***GVK***	*2 bytes (low byte first)*
0	**1**	**1**	**3**	***VK***	*1 byte*
1	**0**	**0**	**4**	***unused***	**-**
1	**0**	**1**	**5**	***CVO***	*2 bytes (low byte first)*
1	**1**	**0**	**6**	***GVO***	*2 bytes (low byte first)*
1	**1**	**1**	**7**	***VO***	*1 byte*

CMS MAP MESSAGES

REGION ENQUIRY

This is an ASCII message which is received from the map.

It requests the following data for a specific region:

- region status
- a specific intersection
- all intersections

Byte	Data
1-6	ASCII region name with trailing spaces
7…	**ASCII** intersection number (most significant digit first), where: 0 requests a region status message 1-64999 requests an intersection status message for the nominated intersection 65535 requests intersection status messages for ALL intersections
n	***a carriage return character terminates the message***

INTERSECTION STATUS

This message is a snapshot of an intersection showing the operating status. It is sent when an alarm state or an operating state changes. It contains flags showing items such as:

- operating **mode** (0-7) where:

0=	Isolated	**4=**	Police Off
1=	Flexilink	**5=**	Police Manual
2=	Masterlink	**6=**	Police Red
3=	Hurry call	**7=**	Maintenance

- LM# is set if the operating mode (link mode) is locked (M#, F# or I#)
- running phase (1=A, 7=G)
- phase step where:

0=	**LST**	Late Start	**4=**	**ECG**	Early cut-off Green
1=	**MIN**	Minimum Green	**5=**	**YEL**	Yellow
2=	**VIG**	Variable Initial Green	**6=**	**RED**	Red
3=	**GRN**	Rest	**7=**	**R/Y**	Rd/Yellow

- active Special Facilities: Z- Z+ Y- Y+
- dwell state
- intersection alarms, vehicle detector alarms (1-24) and pedestrian push button alarms (1-8)
- running intersection plan (IP) 0-8 where 0=plan 0 trim - 15=plan 0 lock (PL0#)
- running offset plan (PP) 0-4 where 0=PP0 trim - 7=PP 0 lock (PP0#)
- subsystem locks is set if a plan locks (PM#) and/or a cycle lock (CL^) is active
- subsystem alarms: HID=High Density, FBK=Fallback
- FTP shows Fixed Time Plans are active
- clock alarm (0=none) - error range is +/-59 seconds - (‘80’x means LARGE error)

Byte	DATA	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1	Header	201 octal							
2	Byte count	16 decimal							
3	Intersection	intersection number *(low byte)*							
4	number	intersection number *(high byte)*							
5	Plans	SS locks	PP			IP			
6	Mode	mode *(see above)*				Z-	Z+	Y-	Y+
7	Phase step	v4.xx	v5.xx	phase		phase step			
8	Next phase	Dwell	HID	FTP	FBK	LM#	phase called		
9		PK	ST	SC	LC	OD	BO	WD	PF
10	Alarms		NC	FY	FL	UD	DA	BD	CK
11		CE	PE	SY	IV	NF	CF	GW	SF
12		DZ	IH	XU	SI	LF	GT	IR	
13	Detector	detector alarms *(1-8) bit 0=1*							
14	alarms 1-16	detector alarms *(9-16) bit 0=9*							
15	PB 1-8	push button alarms *(1-8) bit 0=1*							
16	CA+/-59	clock error *(in secs)* (‘80’x = CA**)							
17	DA 17-24	detector alarms *(17-24) bit 0=17*							
18	CL	cycle length							

19	Checksum	checksum *(negated sum of bytes 1-18)*

MAP MODE

The **map mode** message is sent to the map.

Byte	DATA	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1	*message header*				201 octal				
2	*byte count*				2 decimal				
3	*mode*	0 = normal display 1 = detector alarm display 17 = lamp test				200 = printers are OK 201 = Event log printer faulty 202 = Fault log printer faulty			
4					377				
5	*Negated sum of bytes 1-4*				Checksum				

TIME

The **time request** is an **ASCII** message which is received **from the map**.

Byte	Data
1-4	4 byte ASCII string **"TIME"**
5	a **carriage return** terminates the message

TIME REPLY MESSAGE FORMAT

The reply is sent to the map.

Byte	DATA	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1	*message header*				201 octal				
2	*byte count*				8 decimal				
3	*1 word message code*				22 octal				
4					377 octal				
5					year (since 1900)				
6					month				
7	*date and time*				day				
8					hours				
9					minutes				
10					seconds				
11	*Negated sum of bytes 1-10*				Checksum				

COMMUNICATIONS STATUS

This message the status of the communications link to a region..

Byte	DATA	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1	*message header*				**201** *octal*				
2	*byte count*				**9** *decimal*				
3	*1 word message code*				**20** *octal*				
4					**377** *octal*				
5-10	*region name:*				***ASCII*** *with trailing nulls*				
11	*Communication status flags*						**L**	**W**	**C**
12	*Negated sum of bytes 1-11*				**Checksum**				

The communications status flags in byte 11 are:

Bit	meaning when set	
0	**C**	**Carrier OK**
1	**W**	**Watchdog OK**
2	**L**	**Communications link OK**

SUMMARY OF ALARM ON A REGION

Byte	*Data*	*Bit 7*	*Bit 6*	*Bit 5*	*Bit 4*	*Bit 3*	*Bit 2*	*Bit 1*	*Bit 0*
1	Header	201 octal							
2	Byte count	10 decimal							
3		21 octal							
4		377 octal							
5-10	Region name	ASCII with trailing nulls							
11	Status	**SC**	**LA**	**MAJ**	**LC**	**DET**	**TIM**	**BSY**	**UD**
12	Flags	**DWL**	**TRM**	**TMP**	**CL^**	**PL#**	**INF**	**HID**	**FBK**
13	Checksum	checksum *(negated sum of bytes 1-12)*							

STATUS FLAGS

Bytes 11 and 12 of the record showing a summary of regions alarms contain flags with the following meaning.

flag byte 11	*bit 0*	**UD**	*Failure to update RAM at an intersection*
	bit 1	**BSY**	*Disk write in progress (SYS.LX or INTFIL.DAT)*
	bit 2	**TIM**	*System time is incorrect*
	bit 3	**DET**	*An intersection has a detector alarm*
	bit 4	**LC**	*An intersection has a long Clearance alarm*
	bit 5	**MAJ**	*An intersection has a major alarm*
	bit 6	**LA**	*An intersection has a Lamp Failure*
	bit 7	**SC**	*An intersection Short Clearance*
flag byte 12	*bit 0*	**FBK**	*A subsystem is on fallback*
	bit 1	**HID**	*A subsystem has a high density alarm*
	bit 2	**INF**	*A subsystem has an increment failure alarm*
	bit 3	**PL#**	*A plan lock is present*
	bit 4	**CL^**	*A subsystem has a cycle time locked*
	bit 5	**TMP**	*CPU over-temperature warning*
	bit 6	**TRM**	*An intersection plan has been trimmed*
	bit 7	**DWL**	*An intersection has been dwelled*

CMCS INTERFACE

This section provides details on messages between SCATS and the Central Management Computer System (CMCS) in the RTA Traffic Management Centre (TMC).

MESSAGE FORMATS - SCATS TO CMCS

This is a description of the messages formats between SCATS and the CMCS.

- Messages will mostly be of variable length and will not exceed 1k bytes.
- They will be sent via a mailbox.
- Depending on the message type, a six character region name may be present to indicate the source of a message
- One or more sets of data specific to the message type will then follow.
- The number of sets of data will be determined from the total number of bytes in the message retrieved from the mailbox.

Where SCATS already has existing messages containing the necessary information, those messages will be passed on from the region to the CMCS process without modification.

MESSAGES FROM CMCS TO SCATS

Scats processes will assign to CMCS_MAIL (the logical name for the VAX), then send a message consisting of:

Byte	Meaning
1	*Message type 255*
2	*Mail box unit number for CMCS request (low byte)*
3	*Mail box unit number for CMCS request (high byte)*
4	*First message types accepted by the mailbox*

This message is sent by each SCATS VAX process to the CMCS interface process, notifying the CMCS interface of the mailbox ID to which the CMCS should send requests of the type(s) specified from byte 4 onwards.

The CMCS interface is to have one input mailbox which must have a logical name of **CMCS_MAIL**.

If the CMCS interface terminates, it must delete the logical name CMCS_MAIL then deassign the mailbox before the program exits.

It must assign to the mailbox of each VAX process from which it requires data.

CMCS INTERFACE PROGRAM SHUTDOWN

The program must delete the logical name CMCS_MAIL before deassigning the mailbox and terminating, to guarantee that other processes do not keep an assignment to an unused mailbox.

READING AN ENTRY FROM A TC FILE

MESSAGE TYPE 9

This message is to be sent by the CMCS to request that the set of commands defined by the TC number in bytes 8 and 9 be read from the SYS.TC file at the nominated SCATS region.

As confirmation that the command has been received by the region, the command will be echoed to the CMCS by the region.

FORMAT

Byte	Meaning
1	*Message type 9*
2-7	*Region name (ASCII)*
8	*TC number (low byte)*
9	*TC number (high byte)*

VOLUME AND DS FOR STRATEGIC APPROACHES

MESSAGE TYPE 11

This message consists of a message type, regional computer source and a starting SA number, followed by sets of volume and degree of saturation (DS) data. The SA number in byte 8 defines the SA source for the first set of volume and DS data. Subsequent sets of data are from SAs numbered contiguously from that SA. A regional computer can have up to 250 SAs. This data will be sent by each region at a regular interval.

FORMAT

Byte	Meaning
1	*Message Type **11***
2-7	*Region name - ASCII with trailing blanks*
8	*The SA number providing the first set of data*
9	*Sum of volumes from all active lanes in the SA as a flow in 90 seconds*
10	*VO/VK expressed as a % from the lane with the highest DS*
11	*Highest lane DS*
12	*Volume for the next contiguously numbered SA (flow in 90 seconds)*
13	*VO/VK as a %*
14	*DS*
15…	*Additional sets of Volumes and DS values - The number of sets of SA data is calculated by taking the total number of bytes in the message, subtracting 8 for the message type, Region name and starting SA number, then dividing by 3 - eg: if the message had 212 bytes, there would be (212-8)/3 or 68 sets of data. If the first SA was number 125 (ie in byte 8), the message would contain Volume, VO/VK ratio and DS values for SA 125 to SA 192 inclusive.*
n-2	*Last Volume of this set (flow in 90 seconds)*
n-1	*last VO/VK as a %*
n	*Last DS*

REGION STATUS

MESSAGE TYPE 100

The region status message shows the operating condition of a regional computer. This message will be sent to the CMCS interface at a regular interval or when a condition changes.

FORMAT

Byte	Meaning							
1	*Message type **100***							
2-7	*ASCII Region name (with trailing blanks)*							
8						*Comms*	*Traffic*	*Carrier*

where:

Comms	**Bit2**	*clear if communication has been lost with the region*
Traffic	**Bit1**	*clear if the "TRAFFIC" program on the region is not operating*
Carrier	**Bit0**	*clear if the carrier has been lost between the region and the CMS*

STATUS REQUEST

The status for all regions can be requested by the CMCS by sending a single byte containing a message type of 100.

Byte	Meaning
1	*Message type **100** - request status of all regions*

FAULT DATA FOR SCATS MONITORING SITES

MESSAGE TYPE 101

Byte 1 is the message type. Subsequent data will be grouped in sets of 3 bytes. The first two bytes of a set contain the intersection number. The last byte of the set contains the alarm flags for the fault conditions.

Data will be sent to the CMCS interface for those intersections where a change of state occurs. This is a variable length message.

FORMAT

Byte	Meaning							
1	*Message type **101** decimal (65Hex)*							
2	*First Intersection number (low byte)*							
3	*First Intersection number (high byte)*							
4	***Dwell***	***Trim***	***FBK***	***Mode***	***Major***	***Warn***	***FY/BO***	***DA***
5								
6	*Second Intersection number (low byte)*							
7	*Second Intersection number (high byte)*							
8	***Dwell***	***Trim***	***FBK***	***Mode***	***Major***	***Warn***	***FY/BO***	***DA***
9								
…	*Additional data - the number of sets of intersections in the message is the total number of bytes divided by 4 - eg if the message contained 112 bytes, there are 112/4 or 28.intesections in the message. Note that the second data byte is currently unused.*							
n-3	*Last Intersection number (low byte)*							
n-2	*Last Intersection number (high byte)*							
n-1	***Dwell***	***Trim***	***FBK***	***Special***	***Major***	***Warn***	***FY/BO***	***DA***
n								

where a bit set in the first data byte means:

Dwell	**Bit7**	*the intersection has an active stage dwell (ie manual control)*
Trim	**Bit6**	*the intersection or its subsystem has a manual lock ie LM# PL# PP# SP#, LP# or CL^*
FBK	**Bit5**	*the intersection has been forced to fallback mode by its subsystem*
Special	**Bit4**	*the intersection is operating under a special mode - ie. Police manual, Police Red, police OFF, Hurry call or Maintenance Interrupt*
Major	**Bit3**	*the intersection has a major alarm: NC, FY, FL, BD, CK, ST, BO, WD, PF, DZ, GT, IR, PE, CF, GW, SF*
Warn	**Bit2**	*the intersection has a minor alarm: UD, SC, LC, OD, IH, XU, LF*
FY/BO	**Bit1**	*the lamps are either Flashing Yellow or Blacked Out*

DA	**Bit0**	*the intersection has a DA or SI alarm*

FAULT DATA REQUEST

The fault data for all intersections can be requested by the CMCS by sending a single byte containing a message type of 101.

Byte	Meaning
1	*Message type 101 - request fault data for all intersections*

CONGESTION ALERTS

MESSAGE TYPE 103

This message shows the state of non-recurrent congestion for each Strategic Approach at the nominated SCATS region. A message will be sent for each region at one minute intervals.

Byte	Meaning
1	*Message type 103*
2-7	*Region name (ASCII)*
8-39	*32 bytes of congestion alert for SA 1-250 by bit position - ie bit 0 is set in byte 8 if SA 1 has a congestion alert, Bit1 is set if SA 2 has a congestion alert etc.*

TRAVEL TIME REQUESTS

MESSAGE TYPE 104

The CMCS may request travel times between a start and an end junction.

Byte	Meaning
1	*Message type 104 - Travel time request*
2	*First intersection - low byte*
3	*First intersection - high byte*
4	*Last intersection - low byte*
5	*Last intersection - high byte*

TRAVEL TIME REPLY

MESSAGE TYPE 105

Byte	Meaning
1	*Message type 104 - Travel time reply*
2	*First intersection - low byte*
3	*First intersection - high byte*
4	*Last intersection - low byte*
5	*Last intersection - high byte*
6	*Travel time in seconds - low byte*
7	*Travel time in seconds - high byte*

CONTROL OF TIDAL FLOW OPERATION

To be advised

CONTROL OF DRIVER ADVISORY SYSTEMS

To be advised

TIME SYNCHRONISATION

The system time can be obtained from the VAX using the system service **SYS$NUMTIM**.

HEARTBEAT

To be determined.

FORMAT OF INTERSECTION GRAPHICS FILE

File name: PCINTGRF.TXT

LOCATION RECORD - Record type 0

Maximum record size is 42 bytes.

All characters in this record are ASCII.

Byte	Item	Comments
1-5	5 digit intersection number	right justified
6	Record type 0	
7…	Suburb name	20 characters maximum plus a % delimiter
…	RTA Division	4 characters maximum plus a % delimiter
…	UBD map reference	8 characters maximum plus a % delimiter
…	Change flag	! = data has changed, blank otherwise

COORDINATES RECORD - Record type 1

Record size 30 bytes

NB - items shown with an asterisk after the byte location are numeric data in sixel format. All other values are in ASCII.

Byte	Item	Comments
1-5	5 digit intersection number	right justified
6	Record type 1	
7-8*	Scale	
9-10*	X coordinate of centre of intersection	X + 1024
11-12*	Y coordinate of centre of picture	Y + 1024
13-14*	X coordinate of first signal group display box	
15-16*	Y coordinate of first signal group display box	
17-18*	X coordinate of second signal group display box	
19-20*	Y coordinate of second signal group display box	
21-25*	X coordinate of intersection location	
26-30*	Y coordinate of intersection location	

ARM RECORD - Record type 2

Record size 38 bytes.

NB - items shown with an asterisk after the byte location are numeric data in sixel format. All other values are in ASCII.

Byte	Item	Comments
1-5	5 digit intersection number	right justified
6	Record type 2	
7*	Arm number	
8*	Street name index + 32	Index to a street name in a STREET RECORD (record type 5)
9-10*	Angle of arm	12 bit value - 0 is 9 o'clock
11-12*	Offset	Offset of arm from centre of intersection in logical units. An intersection picture is 512 x 512 logical units.
13-14*	Lane configuration	Lanes left of centre line + (lanes right of centre lane * 16) + (Slip lanes * 256) (standing in the arm, looking towards the centre of the intersection)
15-16	Pedestrian type	
17	Median type	
18	Break	
19-20*	Side 1	
21-22*	Side 2	
23-24*	Entry	
25-26*	Exit	
27*	Exit arm	
28-29	Slip lane pedestrian crossing	
30-31*	Pedestrian crossing position	
32-35*	Lane arrows	Allows for 8 sets of 3 bits (ie for 8 lanes in order from the centre line to the kerb) where: bit 0 = Left bit 1 = Through bit 2 = Right
36-38*	Next intersection along arm	0 = none, -1 = stub (no next intersection) 1-63999=next intersection number 64000-64999=exception points 65000…=unused

SIGNAL GROUP RECORD - Record type 3

Maximum record size is 30 bytes.

NB - items shown with an asterisk after the byte location are numeric data in sixel format. All other values are in ASCII.

Byte	Item	Comments
1-5	5 digit intersection number	right justified
6	Record type 3	
7*	Signal group number	
8-9*	Group direction and type	Entry arm number + (exit arm number * 16) + (type * 256) Type (4 bit field): Top bit = display in Box 2 (else Box 1) Lower 3 bits = type: 1 = Through 2 = Right 3 = Left 4 = Pedestrian
10*	Next signal group number	
11-12	Group direction and type	Entry + (exit * 16) + (type * 256)
…	*additional groups* *maximum of 8 groups per record* *groups are numbered from 1 to 16*	
…*	Last signal group number in record	
…*	Group direction and type	Entry + (exit * 16) + (type * 256)

PHASE RECORD - Record type 4

Maximum record size is 50 bytes

NB - items shown with an asterisk after the byte location are numeric data in sixel format. All other values are in ASCII.

Byte	Item	Comments
1-5	5 digit intersection number	right justified
6	Record type 4	
7-8	Phase name	eg A or B or F1 etc
9…*	Signal groups associated with this phase.	Up to 8 group numbers may be present, with each group number being stored as (group number + 32).
…	Phase group delimiter	% delimiter
	A record may contain data for up to 4 phases.	
…	Name of next last phase	eg A or B or F1 etc
…*	Signal groups associated with this phase.	Up to 8 group numbers may be present, with each group number being stored as (group number + 32).
…	Phase group delimiter	% delimiter

STREET RECORD - Record type 5

Maximum record size is 50 bytes

NB - items shown with an asterisk after the byte location are numeric data in sixel format. All other values are in ASCII.

Byte	Item	Comments
1-5	5 digit intersection number	right justified
6	Record type 5	
7*	Index number + 32	Corresponds to the index to a street name in an ARM RECORD (record type 2)
8…	First street name	% delimited
…*	Index number + 32	Corresponds to the index to a street name in an ARM RECORD (record type 2)
…	Second street name	% delimited

DETECTOR RECORD - Record type 6

Maximum record size is 38 bytes

NB - items shown with an asterisk after the byte location are numeric data in sixel format. All other values are in ASCII.

Byte	Item	Comments
1-5	5 digit intersection number	right justified
6	Record type 6	
7*	Detector number	4 bytes per detector
8*	Arm number	
9*	Lane number	Lane number + 32
10*	Detector position	Position + 4 + (slip lane flag * 32)
up to 8 detectors per record		

ALARM AND EVENT FILES

When a regional computer is connected to a **CMS** system, all alarms and events that occur at the region are sent to the CMS for storing.

The **CMS** creates *daily* alarm and event files. These files are in directory **SYS$ALARMS**. The files are named after the date and have a **.ALM** extension for alarms and a **.EVT** extension for events. For example, the alarm file for the 23rd October 1998 is called **981023.ALM** and the event file is **981023.EVT**.

ALARM FILE

The alarm file is a fixed length record file where each record has **32** bytes. The file contains:

- Intersection alarm records
- regional computer faults
- CMS mailbox status records
- Event file pointer records - pointers to records which have more than 32 bytes

EVENT FILE

The event file contains:

- operator commands
- text messages
- CMS mailbox status records
- Diagnostic messages

ALARM FILE FORMATS

Intersection status

Byte	DATA	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1	Year	-	Year since 1900						
2	Month	-	-	-	-	Month			
3	Day	-	-	-	Day				
4	Hour	-	-	-	Hour				
5	Minutes	-	-	Minutes					
6	ID	-	-	-	^N character				
7	Intersection	intersection number *(low byte)*							
8-13	Region	Region name							
14	Intersection	intersection number *(high byte)*							
15	Plans	SS locks	PP			IP			
16	Mode	mode *(see below)*				Z-	Z+	Y-	Y+
17	Phase step	v4.xx	v5.xx	phase		phase step			
18	Next phase	Dwell	HID	FTP	FBK	LM#	phase called		
19		PK	ST	SC	LC	OD	BO	WD	PF
20	Alarms		NC	FY	FL	UD	DA	BD	CK
21		CE	PE	SY	IV	NF	CF	GW	SF
22		DZ	IH	XU	SI	LF	GT	IR	
23	Detector	detector alarms *(1-8) bit 0=detector 1*							
24	alarms 1-16	detector alarms *(9-16) bit 0=detector 9*							
25	PB 1-8	push button alarms *(1-8) bit 0=push button 1*							
26	CA+/-59	clock error *(in secs)* ('80'x = CA**)							
27	DA 17-24	detector alarms *(17-24) bit 0=detector 17*							
28	CL	cycle length							
29	spare	-							
30	spare	-							
31	spare	-							
32	Seconds	-	-	Seconds					

MODE

SCATS v4.x

Byte 16 (bits 6-7) contain the operating mode:

bit 7	Masterlink	bit 6	Flexilink	neither set	Isolated

SCATS v5.x

Byte 16 (bits 5-7) contain the operating mode as a 3 bit value:

0	Isolated	1	Flexilink	2	Masterlink	3	Hurry call
4	Police off	5	Police manual	6	Police red	7	Maintenance interrupt

REGION ALARM SUMMARY

This record provides a summary of all alarms on a regional computer.

Byte	DATA	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1	Year	-	Year since 1900						
2	Month	-	-	-	-	Month			
3	Day	-	-	-	Day				
4	Hour	-	-	-	Hour				
5	Minutes	-	-	Minutes					
6	ID	-	-	-	1E hexadecimal				
7	Alarms	SC	OD	MAJ	LC	DET	TIM	BSY	UD
8-13	Region	Region name							
14	Alarms	DWL	TRM	TMP	CL#	PL#	INF	HID	FBK
…	unused	*bytes 15 to 31 are not used in this record*							
32	Seconds	-	-	Seconds					

CMS MAILBOX STATUS

This is a message from the VAX computer reporting on the status of the mailbox to a PDP11 CMS computer.

Byte	DATA	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1	Year	-	Year since 1900						
2	Month	-	-	-	-	Month			
3	Day	-	-	-	Day				
4	Hour	-	-	-	Hour				
5	Minutes	-	-	Minutes					
6	ID	-	-	-	^Z (1A hex)				
7	State								**state**
…	spare	*bytes 8 to 31 are not used in this record*							
32	Seconds	-	-	Seconds					

MAILBOX STATE

Byte 7 contains the mailbox state in bit 0.

Bit 0	CMS Mailbox State
0	**FAULTY**
1	**OK**

REGION STATUS RECORD

This record shows the operational status of a regional computer.

Byte	DATA	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1	Year	-	Year since 1900						
2	Month	-	-	-	-	Month			
3	Day	-	-	-	Day				
4	Hour	-	-	-	Hour				
5	Minutes	-	-	Minutes					
6	ID	-	-	-	1F hexadecimal				
7	Status	Source			Error code				status
8-13	Region	Region name							
…	*spare*	*bytes 14 to 30 are not used in this record*							
31	zero ID	-			**0**				
32	Seconds	-	-	Seconds					

STATUS

Byte 7 contains three items:

Status

This is a 1 bit status code, where:

Value	Status
0	Faulty
1	OK

Error code

This is a 4 bit code where:

Octal value	Fault type
0	Carrier
1	Computer
2	Communications
10	Temperature

Source

The source shows the sender of the record. This is a 3 bit value where:

Value	Source
0	CMS
1…6	ADC 1…6
7	VAX

EVENT FILE RFA POINTER

To maintain a relationship between intersection alarms and events, it has been necessary to use a form of record synchronisation between the intersection status records on the .ALM file and events on the .EVT file. This is done by keeping a pointer in the alarm file to all records in the event file. This allows the alarm file to be read sequentially and still be able to extract records from the event file in the order that the conditions occurred. The reverse is not considered necessary.

Therefore, every time the CMS writes a record to the event file, it also writes a record to the alarm file which contains the disk block and byte offset of the location of the event file record.

Knowing the RFA (Record File Address) allows a record on the event file to be directly read without the need to read sequentially from the start of the file.

Byte	DATA	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1	Year	-	Year since 1900						
2	Month	-	-	-	-	Month			
3	Day	-	-	-	Day				
4	Hour	-	-	-	Hour				
5	Minutes	-	-	Minutes					
6	**Type**	-	-	-	^C, ^P or 1F character				
7	**Flag**	ID of record type on event file							
8-13	Region	Region name							
14-17	Disk block	RFA disk block (integer*4)							
18-19	Byte offset	RFA byte offset (integer*2)							
20-30	spare	*bytes 20-30 are not used in this record*							
31	RFA ID	1F hexadecimal - denotes that this is an RFA pointer record							
32	Seconds	-	-	Seconds					

TYPE POINTER

Byte 6 contains the type of record being pointed to on the event file, where:

Type	Meaning
^C	Operator command
^P	Text message
1F	Diagnostic record

FLAG

Byte 7 contains an identifier for the record in the event file.

Flag	Meaning
0	unknown .EVT file record
80x	LF record

EVENT FILE RECORD FORMAT

The event file is a sequential file containing variable length records up to a maximum of 128 bytes.

This file contains operator commands, text messages, diagnostic records and region status records.

DIAGNOSTIC RECORD

Byte	DATA	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1	Type	-	-	-	1F hexadecimal				
2-7	Sender	ASCII sender name with trailing nulls							
8	Date	<< Month (4 bits)			Day (5 bits)				
9	Date	Year since 1900 (7 bits)							Month
10	Time	<< Minutes (6 bits)			Seconds (halved) - (5 bits)				
11	Time	Hours (5 bits)					Minutes (continued) >>		
12	Fault	Source			Fault code				State
13	Flags	Flags			Extended fault code				
14…	*Text*	*optional text follows*							

FAULT CODES

Byte 12 contains three items.

STATUS

Bit 0 is a 1 bit status code, where:

Value	Status
0	Faulty
1	OK

Fault code

Bits 1 to 4 contain the fault code where:

Value	Fault type
0	**CARRIER**
1	**REGIONAL COMPUTER** if the source is CMS or ADCn -or- **COUNTING STATION n.m UPLOAD** if source is VAX where bytes 13 to 16 contain the 4 byte station number (n.m) stored as n*1000 + m
2	**COMMUNICATIONS** if source is CMS or ADCn -or- **ADP TIMED DISPLAY MESSAGE** where byte 13 contains the display period in seconds and byte 14 onwards contains the message to be displayed
3	**LAN MESSAGE** where byte 13 contains the device type and byte 14 onwards contains the device name. The byte 13 device type is: **1 TERMINAL SERVER** **2 TRANSLAN BRIDGE** **3 BATCH/PRINT QUEUE** **4 PROCESS**
8	**POWER CONDITIONER**
9	**AIR CONDITIONER**
10	**TEMPERATURE ALARM**
11	**FIRE ALARM**
12	**VAX WATCHDOG**
13	**CMS WATCHDOG** (PDP11)
15	**SPECIAL** for EXTENDED STATUS MESSAGE follows in byte 13

Extended Fault codes

If the fault code in byte 12 contains a value of 15, byte 13 contains a 5 bit extended error code in bits 0 to 4, where:

Value	Fault type
1	COMMUNICATIONS STATUS of the computer name from bits 5, 6 and 7 where: **0 = CMS, 1=ADC1 … 6=ADC6, 7=VAX**
2	**STRATEGIC ROAD NETWORK** - (bits 5, 6,7 ignored) The following byte 14 contains the SRN number, where: 0 = **SRNIO - LINE ACCESS** 1-127 = **SRNIO - SRN No. n ACCESS** if bit 7 is set, append **AFTER CALL-IN FAILURE**
3	**ADC MAP WATCHDOG** (bits 5, 6, 7 ignored)
4	**TRANSLAN BRIDGE** (bits 5, 6, 7 ignored). The name of the bridge follows from byte 14 onwards.
5	**LOGGING PRINTERS** where bit 5=**EVENT** printer and bit 6=**FAULT** printer

Source

The source shows the sender of the record. This is a 3 bit field in bits 5, 6 and 7, where:

Value	Source
0	CMS
1…6	ADC1…ADC6
7	VAX

If the first character of the sender name (byte 2) is a blank, the sender name is the name of the computer that wrote the message to the log. This may not be the name of the computer which initiated the message. Use the source code from bits 5-7 in byte 12 as the computer reporting the fault.

If the first character of the sender name is not blank, it is a region name whose status is being reported.

OPERATOR COMMANDS

Operator commands are sent whenever a change is made to SCATS data or traffic operation.

Byte	DATA	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1	Type	-	-	-	^C (03 hex)				
2-7	Region	ASCII region name with trailing nulls							
8	Date	<< Month (4 bits)			Day (5 bits)				
9	Date	Year since 1900 (7 bits)							Month
10	Time	<< Minutes (6 bits)			Seconds (halved) - (5 bits)				
11	Time	Hours (5 bits)					Minutes (continued) >>		
12	Commands	Command byte no.1							
…	*Commands*	*additional command bytes*							
n	*Commands*	*last command byte (record size is variable - maximum size is 128 bytes)*							

TEXT MESSAGES

Text messages include:

- Route Preemption (RPC) messages
- TRAFF STOPPED / TRAFF RUNNING
- Security key LOGON/LOGOFF messages
- Data CHANGE messages
- TC DATA ERROR messages
- Messages from TI operator terminals

Byte	DATA	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1	Type	-	-	-	^P (10 hex)				
2-7	Region	ASCII region name with trailing nulls							
8	Date	<< Month (4 bits)			Day (5 bits)				
9	Date	Year since 1900 (7 bits)							Month
10	Time	<< Minutes (6 bits)			Seconds (halved) - (5 bits)				
11	Time	Hours (5 bits)					Minutes (continued) >>		
12	Message	Message byte no.1							
…	*Message*	*additional message bytes*							
n	*Message*	*last message byte (record size is variable - maximum size is 128 bytes)*							

NOTES

If message byte no.1 has bit 7 set, it is a flags byte, not part of the message text. When it is a flags byte:

- if bit 6 is also set, this is a warning message for display on the Alarm Display Panels (ADPs)
- if bit 4 is also set, the next byte (no.13) is an ID for the ADP

Thus, if bit 7 is set in message byte 1:

- when bit 7 and bit 4 are both set, the message starts in byte 14
- when bit 7 is set but bit 4 is clear, the message starts in byte 13
- otherwise, the message start in byte 12

VAX VOLUME STORE DETECTOR FILE FORMAT

When a regional computer is connected to a CMS system, detector volumes can be collected on the VAX management computer. The format of records on this VS file is as follows:

COMMENT RECORDS

The file starts with a comment record which describes the data collection. This record shows the start and finish dates, daily collection periods, intersection numbers and detector numbers.

The first 7 records on a VS file are comment records. Each comment record is 128 bytes.

Byte	DATA	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1-4	ID	A comment record has a zero in the first 4 bytes							
5	Type	^V for VS data (^S for SM data)							
6-11	Region	Region name in ASCII							
12	Year	Start year (since 1900)							
13	Month	Start month							
14	Day	Start day							
15	Year	End year (since 1900)							
16	Month	End month							
17	Day	End day							
18	Hour	Daily period 1 - start hour							
19	Minutes	Daily period 1 - start minutes							
20	Hour	Daily period 1 - end hour							
21	Minutes	Daily period 1 - end minutes							
22	Hour	Daily period 2 - start hour							
23	Minutes	Daily period 2 - start minutes							
24	Hour	Daily period 2 - end hour							
25	Minutes	Daily period 2 - end minutes							
26	Intersection	First Intersection number (low byte)							
27	Intersection	First Intersection number (high byte)							
28	Detectors	Bit mask for detectors 1 to 8							
29	Detectors	Bit mask for detectors 9 to 16							
30	Detectors	Bit mask for detectors 17 to 24							
31	Detectors	Bit mask for detectors 25 to 32							
…	*Additional Intersection / Detectors*	*Additional intersection numbers and detector bit masks. Six bytes per intersection. Stops at an intersection number of zero.*							
128	No.	Comment number for VS records (starts from no.1)							

Each comment record has 17 six byte fields consisting of a 2 byte intersection number and a 4 byte detector bitmask.

Comment records that contain at least one intersection/ detector bitmask are numbered sequentially in byte 128 starting with a value of 1. Bytes 1 to 25 will also be duplicated.

An intersection number of zero terminates the specification of intersection/detector bitmasks.

Comment records that do not contain any intersection/ detector bitmasks have all bytes set to zero!

DATE/TIME STAMP RECORD

A date/time stamp record is generated every 15 minutes (or every 5 minutes if RK=FV). It is generated 2 minutes after the period, so the minutes need to be decremented by 2 to normalise it (ie the time stamp for 12:15 shows 12:17).

A date/time stamp record always has seven bytes.

If VS data is collected **locally** on a SCATS 6 region, each record will be preceded by a two byte record length (low byte first). If the record length is **odd**, the record will contain a trailing null character as padding to make the overall record length **even**!

Byte	DATA	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1-2	ID	A time stamp record has a zero in the first 2 bytes							
3	Year		Year (since 1900)						
4	Month					Month			
5	Day				Day				
6	Hour				Hour				
7	Minutes	5 min		Minutes + 2					

- The date and time is for the period ending at that time.
- Note that midnight is sent as 00:02 of the next day.
- If bit 7 is set in the minutes byte, the volume data that follows is for a 5 minute period (ie RK=FV!) otherwise the volume data is for a fifteen minute period.

INTERSECTION DETECTOR VOLUME RECORD

Each intersection for which detectors volumes are collected have their own record. To differentiate a volume record from the rest, the first two bytes contain the intersection number which is never zero (a comment record has the first 4 bytes set to zero and a time stamp has the first 2 bytes set to zero).

If VS data is collected **locally** on a SCATS 6 region, each record will be preceded by a two byte record length (low byte first). If the record length is **odd**, the record will contain a trailing null character as padding to make the overall record length **even**!

Byte	DATA	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1-2	Intersection	Intersection number (low byte = byte 1, high byte = byte 2)							
3	volume	11 bit volume (ie plus first 3 bits of the next byte)							
4	detector	5 bit detector number					Volume (continued)		
…	volume/ detector	Additional volume detector number data							
n-1	volume	11 bit volume (ie plus first 3 bits of the next byte)							
n	detector	5 bit detector number					Volume (continued)		

- A volume record can contain a minimum of 4 bytes (2 byte intersection number plus 2 byte volume/detector value) up to a maximum of 50 bytes (2 byte intersection number plus 48 bytes of volume/detector values for detectors 1 to 24).
- A volume where all 11 bits are set (a value of 2047) signifies that the detector had a DA alarm.
- A five minute volume of 255 (the controller sends back volumes every 5 minutes in a single byte) would represent a BAD value (possibly an oscillating detector). Note that it would be unlikely the controller would count over 150 vehicles in one lane in 5 minutes as this would represent one vehicle every two seconds for the entire 5 minute period.

Notes

Notes