UVOT Autonomous Operations

H.E.Huckle and P.J.Smith
Mullard Space Science Laboratory, University College London,
Holmbury St. Mary, Dorking, Surrey, RH5 6NT, UK

ABSTRACT

The SWIFT/UVOT has a requirement for on-board autonomous control of exposures, health and safety. It is anticipated that the optimal form of control may not emerge until after launch and may change during the course of the mission. A flexible and readily re-configurable system is therefore required. Two schemes have been adopted. As well as the more usual approach of tables of experimental configurations, action tables mapping command sequences to key events have been implemented. The command sequences, consisting of a series of command words located in EEPROM, are executed using a stack-based software 'virtual CPU'. Each command word, analogous to hardware CPU assembler instructions, results in the execution of well-checked Ada code fragments. As well as implementing the UVOT commands, the code includes functionality such as delaying a specified time, awaiting action completion, 'subroutine' calls and simple flow control. These permit the construction of complex control sequences. A C-like language is used to describe the required sequences. A translator converts them to the required command word sequence that is then validated on a simulator. Reloading the command sequence or the tables referring to it alters the autonomous behaviour of the instrument.

1. INTRODUCTION

1.1 OVERVIEW

An overview of the UVOT instrument is given elsewhere in these proceedings[1]. This text describes the software components resident in the Instrument Control Unit (ICU) that are responsible for the autonomous control of exposures and for maintaining the health and safety of the instrument. All ICU resident code described here was written in Ada.

Figure 1 summarizes the electronic architecture of the UVOT. The ICU controls and monitors the telescope module (TM) via the instrument control bus (ICB). The TM contains the Telescope Module Power Supply (TMPSU) that controls (a) a filter wheel, the position of which is monitored by LED illuminated sensors, (b) the heaters and (c) a beam deflector that switches the optical path between the prime and redundant halves of the instrument. The Detector Processing Electronics, known as the BPE, acquires and forwards photon events to the Digital Processing Unit (DPU) in the Digital Electronics Module (DEM). The BPE also (1) controls the high voltage unit attached to the image intensifier, (2) activates a calibration flood LEDs, (3) powers to the filter wheel sensors and (4) monitors the effect of the heaters via thermistors. The DPU processes the photon events into lists, images and a parameterized finding chart that are then forwarded directly to the spacecraft.

1.2 SCIENTIFIC GOALS

In order to achieve the scientific goals of the UVOT, the ICU must autonomously control the instrument to perform two types of exposure sequences: automated and planned.

Automated: These occur when Gamma Ray Bursts (GRBs) are detected by the Burst Alert Telescope (BAT) or a Target of Opportunity (ToO) is detected by other spacecraft and ground based sources. On the first observation of the source an event list is gathered while the spacecraft is still settling on the target. An exposure is then made from which a finding chart is constructed to permit optical counterpart identification. All subsequent exposures on this source for this and later slews to the same source will involve obtaining a sequence of images, event lists or combinations thereof in differing filters with increasing exposure lengths. The active area used on the detector (referred to as the window) will

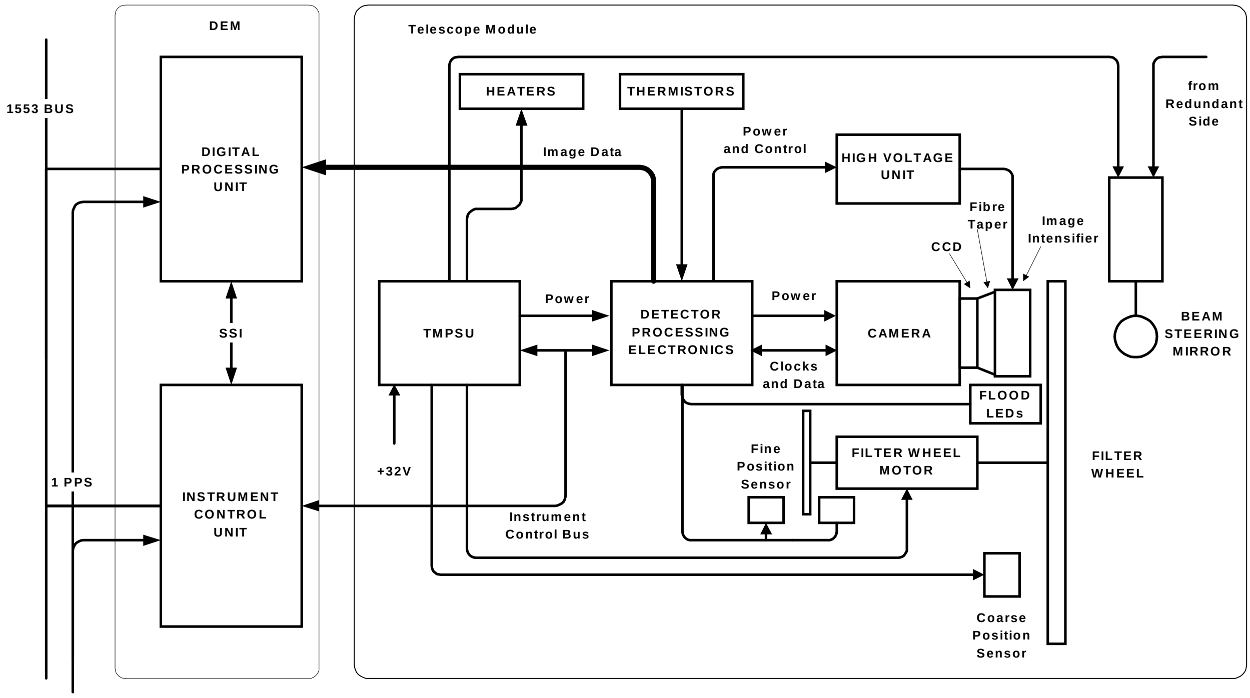

Figure 1: UVOT Electronic Architecture

be refined if the X Ray Telescope (XRT) is able to supply an improved position before the end of the finding chart exposure.

Planned: These are a series of exposures that obtain images, event lists or combinations thereof in various instrument configurations according to an up-linked plan.

1.3 ENGINEERING GOALS

The ICU must respond to pointing constraints imposed by the Sun, Earth, Moon and planets.

Whilst observing a source the ICU needs to avoid damage to the instrument from known bright sources in, or close to, the field of view (FOV). These can cause irreparable damage to the UVOT instrument by depressing the gain of the Micro-Channel Plates (MCPs) or by damaging the photo-cathode.

Additional hardware protection against unknown bright sources is provided by a safety circuit that is described elsewhere in these proceedings[2].

The ICU needs to monitor critical parameters for out-of–limits conditions and take an appropriate recovery action. It needs to protect the instrument during any loss of spacecraft attitude. It must recover from any command failures. It should respond quickly to an emergency shutdown warning from the spacecraft.

It is a project requirement that the code will need to run continuously for at least 72 hours without ground intervention.

The optimal form of control of the UVOT for both science exposures and for handling, or recovering from, conditions that involve detector safety issues may not emerge until late in the development and test program. It may also change during the course of the mission as we accommodate changes in ideas and circumstances during the lifetime of the instrument. Therefore a flexible and readily re-configurable system is required.

1.4 CONSTRAINTS ON OBSERVING

There are a number of reasons why it is impossible to perform the above sequence of exposures continuously on either automated or planned targets.

There are pointing constraints imposed by the Sun, Moon and Earth. Each of these very bright, and therefore potentially damaging, sources has an avoidance angle constraint to which the satellite must adhere. The Swift Observatory will be in a low Earth orbit with an orbital period of approximately 96 minutes. Assuming no additional pointing constraints imposed by the current position of the Sun or Moon, the Earth's avoidance angle implies that it will not be possible to observe any source for longer than about 45 minutes, at which time the satellite must slew to another source. For sources at high elevation above the satellite orbital plane, this maximum observing time is decreased, falling to zero at 84 degrees.

The detector is susceptible to damage from sources such as stars or planets brighter than about 8^{th} magnitude, depending on their colour. The ICU must monitor long exposures and ensure that the total accumulated counts on detector locations associated with a given source do not exceed a damage limit.

Because of the particle radiation present, the instrument may have to be protected during passages through the South Atlantic Anomaly (SAA). These interruptions will occur two or three times in any twenty-four hour period and each may last up to ten minutes. This protection is achieved by ramping down the high voltages controlling the image intensifier.

As a consequence of the above constraints, the ICU design must:

- Allow for automated and planned observations to be interleaved.
- Expect either type of exposure to be interrupted by pointing and SAA constraints.
- Make decisions on exit from interruptions on how to reconfigure the instrument and restart the exposure.
- Permit the curtailing of all types of exposures to prevent detector damage.

1.5 AVAILABLE INFORMATION

The following information is supplied to the ICU to enable it to determine its actions.

The spacecraft supplies an attitude control system message at a frequency of 5 Hz. This contains an identifier uniquely specifying the current observation. It also supplies Boolean flags indicating whether the spacecraft has settled or is within 10 arc minutes of its final position and whether we are currently inside the SAA. It also includes a flag that, if true, signals that the spacecraft may be about to remove instrument power. Positional information, specifically the current right ascension and declination of the source and the satellite's latitude and longitude are supplied. Timing information, in the form of the current spacecraft clock setting, is contained within the same record.

The spacecraft supplies a hardware driven 1 Hz timing pulse, referred to as the 1PPS. Between these pulses, a message that supplies the spacecraft clock and UTC values at the next pulse is sent.

Telecommands are sent by the spacecraft to notify the instrument when a slew is about to commence or if a signalled slew has been abandoned.

Prior to slewing to the next source, referred to here as the target, the Figure of Merit Processor (FOM) sends out a record detailing the next observation. It is referred to as the FONEXTOBSINFO message. This includes an identifier uniquely specifying the next observation and a flag identifying it as an automated or planned target. For an automated target the time since the source was detected and whether this is the first visit to that source are also given. The target right ascension, declination and roll are supplied, together with the anticipated maximum observing time on the source before the next interruption, allowing for anticipated interruptions by the SAA. For all types of observations it further supplies a value known as the UVOT mode that the ICU uses to select the sequence of exposures to run on the target.

The BAT supplies a message detailing burst brightness information.

Finally, the XRT may supply a refined position for a GRB whilst the UVOT is performing the finding chart exposure. This impacts on the area of detector that needs to be processed to ensure the inclusion of the source.

2. DESIGN PHILOSOPHY

In order to achieve the flexibility required, a design based primarily around three types of tables stored in electrically erasable programmable read only memory (EEPROM) was chosen. All these tables have an associated Cyclic Redundancy Check (CRC) value that is used to validate each table as it is loaded into random access memory (RAM) prior to use. It is intended that most, if not all, proposed changes to the behaviour of the system can be achieved by modifying one or more of these tables.

The first type of table is a dataset. These will be configuration description values that may change as a result of testing or observations.

The second type is an action table. It defines actions that are to take place when an event or combination of on-board values occurs.

The third type of table contains the action details. It is a sequence of command words derived from text files containing scripts that are known as Relative Time Sequences (RTSs).

In addition, to determine if there are known bright sources close to or in the field of view, planetary position calculations are performed and an on-board star catalogue is checked.

3. THE RTS SYSTEM

3.1 OVERVIEW

RTSs provide a separate layer of “software” on top of the Ada code that deals only with the higher-level aspects of UVOT control. As the Integrated Test and Operations System (ITOS) is used as the satellite control and monitoring system for the Swift mission, the syntax of the RTSs is, by design, similar to ITOS procedures (which are somewhat like Unix scripts). RTSs therefore act as an on-board UVOT command facility into which changes in ideas and circumstances can more readily be incorporated.

Only one (top-level) RTS may run at any time, although it may call other RTSs as subroutines. Each RTS is assigned a priority, which is also used by any RTSs called. If a commanded RTS has a higher priority than the one currently running, the latter will be shut down and the new RTS run in its place. Priorities are set based firstly on safety and then on science considerations.

3.2 RTS STATEMENTS

A RTS may contain any of the following statements:

Any UVOT telecommand: The arguments supplied may be constants, references to the contents of standard memory locations or arguments given to the calling RTS. A secondary table, directly derived from the ITOS database, contains the formatting information for constructing the command on-board.

An RT-to-RT telecommand: These are system-to-system commands. An example of this is when the UVOT, after receiving a request to slew, acknowledges to the spacecraft that it is ready to slew.

A call to another RTS: This is analogous to a subroutine call. The arguments supplied may be constants, references to the contents of standard memory locations or arguments given to the calling RTS.

A delay statement: This allows a RTS to delay its next action for a number of clock-ticks, where a clock-tick is 0.2 seconds. The delay may be a constant, a reference to the contents of standard memory locations or an argument given to the calling RTS.

A wait for event statement: This causes the RTS to wait for a specified number of seconds – the timeout – for a significant event to happen. An example of such an event is the successful completion of a filter wheel rotation to a commanded filter position. Should that event fail to occur by the end of the timeout, the RTS could, for example, shut down and replace itself with another RTS.

A chain statement: This is effectively an unconditional 'wait for event' with zero as the value for the timeout. It is used to shut down the current RTS and run a complete replacement.

A flow control statement: An "`if … then … else`" construct is supported. Blocks of statements are conditionally executed depending on the values of certain Boolean flags maintained by the underlying Ada code. For example, it is possible to respond differently if the instrument is in the SAA or when the 'wait for event' statement described above exceeds its time limit.

Internal calls: These are commands to procedures supplied by the underlying Ada code. They therefore act as calls to built-in functions. An example of this is a call to the procedure that loads the selected exposure configuration into RAM from EEPROM.

Return: A top-level RTS stops executing. A lower-level RTS returns.

Exit: The current RTS, and any RTSs calling it, stops executing.

Messages: A facility similar in concept to the Unix `echo` command is provided. This allows appropriate diagnostic or error messages to be sent to the ground.

Symbolic Constants and Comments: These are provided to make the scripts more human-readable.

3.3 THE RTS CONVERTER

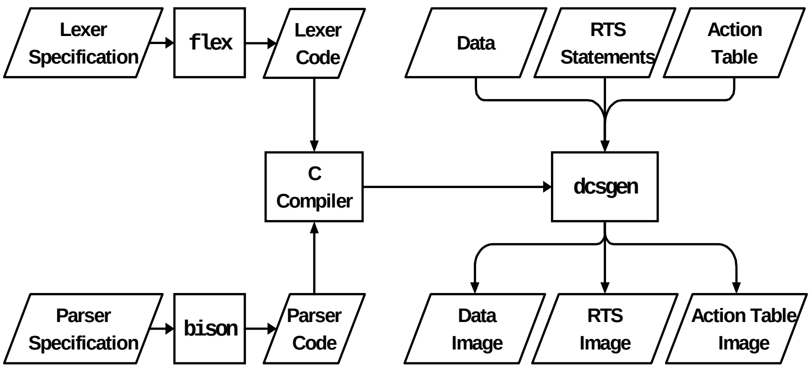

Figure 2: RTS Converter and Image File Generation

The above RTS statements must be converted to command words understood by an on-board command interpreter. This is performed on the ground with a utility generated using the GNU tools `flex` and `bison`. `Flex` generates the lexical analyzer (or 'lexer') layer while `bison` generates the parser layer of the code. The lexical analyzer converts a character stream into a syntactically meaningful string of text tokens. For example the stream of characters "`if true then`" is analyzed and split up into the meaningful keywords "`if`", "`true`" and "`then`". These are then sent to the parser which recognizes the combination of those keywords in that order to be the start of an "`if … then else`" construct. It then generates the appropriate branching instructions. The instructions are written into a file in a format suitable for uploading into EEPROM via ITOS. This type of file is referred to as an "image" file.

It was found convenient to use this generator – referred to as `dcsgen` – to produce all EEPROM loadable tables. This was for two reasons. First, even for those tables that made no reference to a RTS, the lexical and parsing capabilities described above, together with the symbolic constants and comments, were useful in checking and generating the loadable files from a human readable text source. Secondly, those tables that made reference to RTSs require knowledge of their code numbers, which are only readily available within the converter.

The above process is summarized in Figure 2. A detailed description of `flex` and `bison` may be found in reference 3.

3.4 THE VIRTUAL CPU

A standard area of EEPROM contains the generated set of command words. These are best considered as analogous to assembler instructions encountered by a simple hardware CPU. The on-board code is a virtual cpu that, from within a simple loop, reads a command word from EEPROM and then calls a routine that performs the instruction corresponding to that word. It thus 'executes' the command word, analogous to a hardware CPU executing an assembler instruction. For example, if the command word means 'switch on detector', then a routine that performs the necessary instructions to switch on the detector would be called. The virtual CPU normally obtains the next instruction from the next location in EEPROM. However, branch instructions modify that value according to a true or false value found at the top of its internal stack.

Each of the routines called is straightforward in what it does. This enables them to be thoroughly tested prior to launch

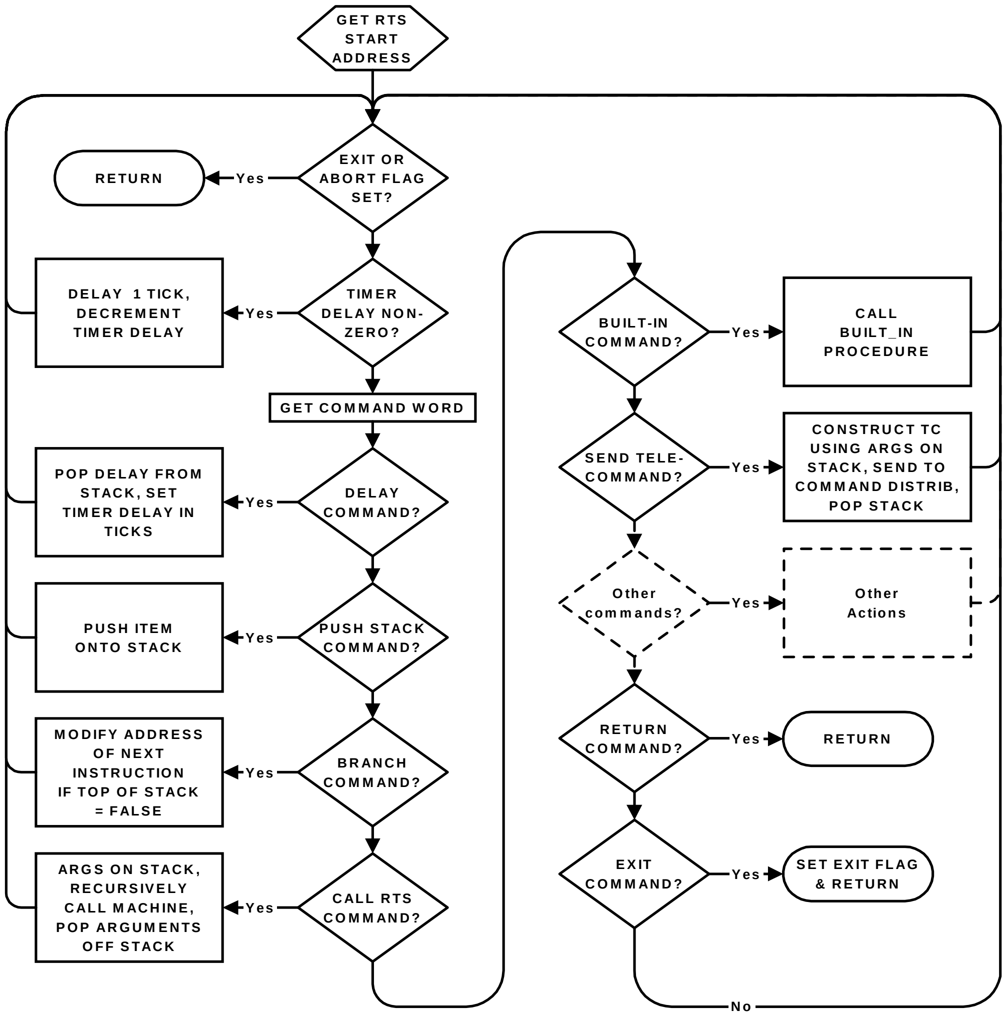

Figure 3: The Virtual CPU Flow Diagram

and not require modification on orbit.

The design uses a stack to hold all temporary values. This is primarily to exploit the nested nature of the RTS language, for example, in allowing the passing of parameters to potentially deeply nested calls to RTSs. It also permits a simpler code design. All commands that produce a result place the values on the top of the stack and commands requiring results of prior operations always find them on the top of the same stack. In order to fully exploit this structure, there are ‘push value onto stack’ or ‘push number at a RAM location onto stack’ instructions available to the virtual CPU.

The flow diagram for Figure 3 illustrates the simple structure of the virtual CPU. For clarity, not all possible command codes are shown.

4. EEPROM LOCATED TABLES

As stated earlier, there are 3 types of table images stored in EEPROM, RTSs, Data and Action Tables.

Data Tables: These contain sets of numbers that may need to change in the course of the mission. These include calibration data for the on-board high voltage ramping and heater control algorithms. Two further tables contain the AT and PT exposure configurations. Count rate and avoidance angle tables are also present and are more fully described in the On-Board Catalogue section.

RTS Images: These contain the translated versions of the RTS scripts and an index into them.

Action tables: These declare a RTS to be run if conditions match entries in the table. They consist of:

a) A state change table, the usage of which is described in the UVOT State Transitions section.

b) A limit-checking table that states which RTS is to be run when a limit failure for a particular engineering item occurs. The items selected are critical to instrument safety, such as voltages or temperatures. For each one, the expected high and low limits, the frequency with which they should be polled and the maximum number of consecutive times the limit may be exceeded are also given.

c) An errors action table that states which RTS is to be run to perform an error recovery action. All telemetry from the ICU and DPU is internally monitored for error messages. When one is detected, the table is then consulted and the RTS run. The design allows for up to 256 such messages.

5. THE BRIGHT SOURCE SYSTEM

5.1 ON-BOARD CATALOGUE

The ICU software uses an on-board star catalogue to determine the magnitude and colour of stars in, or close to, the target field of view. This information is used in two ways to protect the detector from bright source damage:

1. If the star is within an EEPROM tabulated angle of avoidance around the field of view for that magnitude, observation of that field of view is prohibited. This prevents stray light (for example, reflected off the baffle) entering the optics.

2. If the star is in the field of view and does not violate any angle of avoidance criteria, its colour is used to index into a table giving theoretical count rate as a function of filter and then scaled by the catalogued magnitude. The count rate for each star thus calculated is used to decide how long, if at all, an observation at a particular pointing, in that filter, can safely continue. The colour index is a B-V magnitude.

The catalogue is stored in EEPROM and is divided into three contiguous sections: the main catalogue, the addendum and the pointer table

The main catalogue: This contains stars down to 12th magnitude. As the detector is more sensitive to blue than to red, sources that are too red to affect the detector are filtered out. Over 200 000 stars are stored, derived from the Tycho II, GCVS III, NGC and Yale Bright Star catalogues. The catalogue is split into 2524 sky areas of approximately equal solid angle, in 44 declination bands. Within each area the position of each source is stored to +/- half an arc minute accuracy in the right ascension and declination axes, relative to the origin of the area, along with each source's associated magnitude and colour information. Each sky area is followed by a CRC value for memory corruption checking.

The addendum: This is an area left blank in case any sources were omitted from the catalogue prior to launch. At minimum it consists of a marker to show that there are no more sources stored, and the remaining memory zero filled, except for the last word, which is a CRC value. If a source is added to the addendum then it will be put at the beginning of the area, and followed by the marker.

The pointer table: This allows quick access to catalogue data. It holds a set of pointers that, via two levels of direction, point to each sky area. It is CRC protected.

5.2 PLANETARY POSITIONS

In order to protect the instrument from moving celestial object damage, the ICU calculates the positions of the Sun, Earth and Moon as a backup to the protection already provided by the spacecraft. In addition, similar calculations are performed for Venus, Mars, Jupiter, Saturn, Uranus and Neptune, as the spacecraft does not provide this information. All target fields of view are compared against these positions and an EEPROM located table of angles of avoidance. Any violation of those angles prevents the exposure except in the cases of the fainter planets Uranus and Neptune, which are considered as stars and a maximum observing time deduced instead.

The positions are calculated using formulae, algorithms and data given in references 4, 5, 6 and 7.

6. UVOT STATE TRANSITIONS

In order to achieve its goals, the ICU must successfully transition the UVOT between several instrument configuration states. The possible transitions are shown in Figure 4.

Each state transition is performed by a RTS. A look-up table located in EEPROM contains a list of those RTSs against the requested transition. It also contains the expected state of internal ICU flags for that particular transition to be allowed to take place. This table is used in two ways.

On command: A telecommand requesting a particular transition is issued from the ground or, alternatively, from a running RTS. The table is scanned for a match against both the current state and the requested state. If a match is found, the relevant RTS is selected and executed. If no match is found, the command is rejected.

Autonomously: A number of events will cause the ICU to select the required state transition itself by comparing the table of internal ICU flags against the expected values. If a match is found, the RTS is selected and executed. If no match is found, the request is ignored. The events that trigger such a response are

1. The settled flag becoming false
2. The reception of the FONEXTOBSINFO message
3. The 'within 10 arc minutes of target' flag being set to true.
4. Entry or exit from the SAA.

The purpose and activities of each state are summarized below.

Safe: As its name implies, in this state the instrument is configured to be least susceptible to damage. The filter wheel is in a blocked position and the high voltages (HVs) are set to zero.

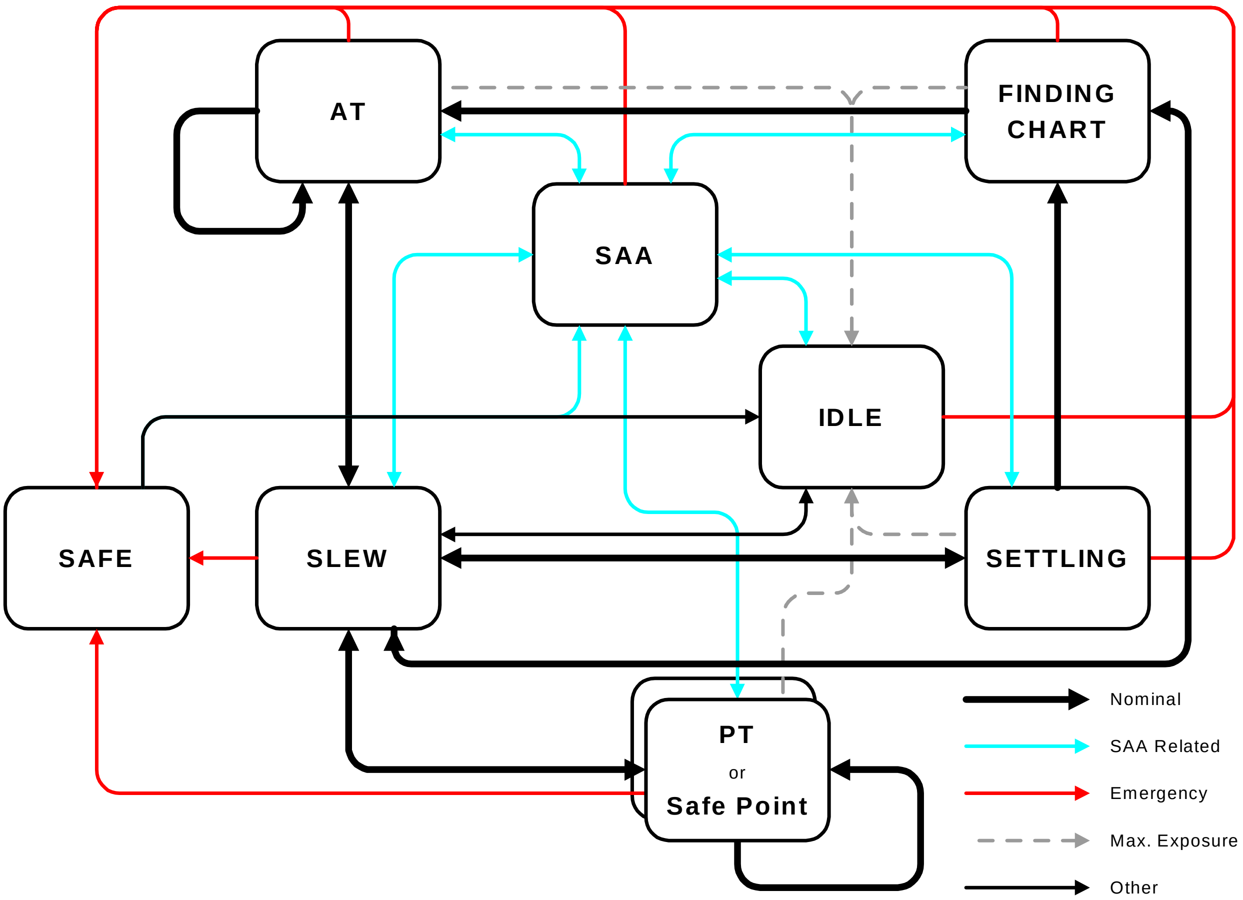

Figure 4: UVOT State Diagram

Idle: The UVOT is configured to be ready to observe but is awaiting the next slew. As a safety precaution the cathode voltage is held down at zero and the filter wheel may be placed in blocked position until observations commence.

Slewing: In this state, the ICU must (a) cleanly shut down any current observation, (b) determine the target position of the slew and the type of observation to be performed when the slew has settled, (c) verify, by calculation, that there are no bright planets or, by examining an on-board catalogue, that there are no bright stars at that location that might damage the detector and then (d) configure the instrument for that observation. If there are bright sources at that position, it must ensure that the exposure does not take place and take appropriate corrective action, for example by reducing the gain of the image intensifier and placing the UVOT into idle state.

Settling: After a new GRB has entered the UVOT's field of view but before the spacecraft has settled, it is observed by collecting an event list. As the target will be moving rapidly it is not possible to collect an image.

Finding Chart: If the target is a new GRB, then once the spacecraft has settled, a 100 second exposure is made in a standard filter to produce a finding chart that will be sent to the ground. In many cases, by the end of the exposure, the XRT will have reported an improved position for the source. The ICU uses that information, and the GRB brightness information supplied by the BAT, when it configures for the subsequent Automated Target exposures.

Automated Target (*AT*): Whilst the GRB has no pointing constraints – this period of time being referred to as a snapshot – the ICU will configure and run a series of exposures. The precise configurations used, for example which filter, the area of the detector used and whether an image, event list or a combination should be acquired is tabulated in a list stored in EEPROM using the UVOT Mode parameter supplied by the Figure of Merit (FOM) in the FONEXTOBSINFO record. They are also a function of whether a refined target position was received from XRT and the intensity of the GRB

Planned Target (*PT*): The ICU will configure this type of exposure when the FOM informs the ICU, via the FONEXTOBSINFO record, that the spacecraft is performing a planned observation. The precise configuration is selected from a large possible selection stored in EEPROM, using the UVOT Mode parameter supplied.

Safe Pointing (*SP*): The ICU will configure this when informed that the spacecraft is slewing to a safe pointing. This type of exposure occurs when no automated or planned targets are available. They are similar to PT exposures. However, an important difference is that it is not possible to perform the safety calculations that are normally carried out in the slew state, as the target pointing information is not available until the spacecraft is settled.

South Atlantic Anomaly (SAA): In this configuration, the instrument is configured to be safe whilst passing through the SAA. In particular the MCP bias voltage is held at 70% of its nominal value and the cathode voltage is at zero.

7. OVERALL DATA FLOW

Figure 5 illustrates how data flows between the software modules dependent on the software components described in this text.

Command Distributor: This module receives not only all the commands and messages sent on the spacecraft bus, but also all commands internally generated by the RTS Manager. It then distributes them to the appropriate software module.

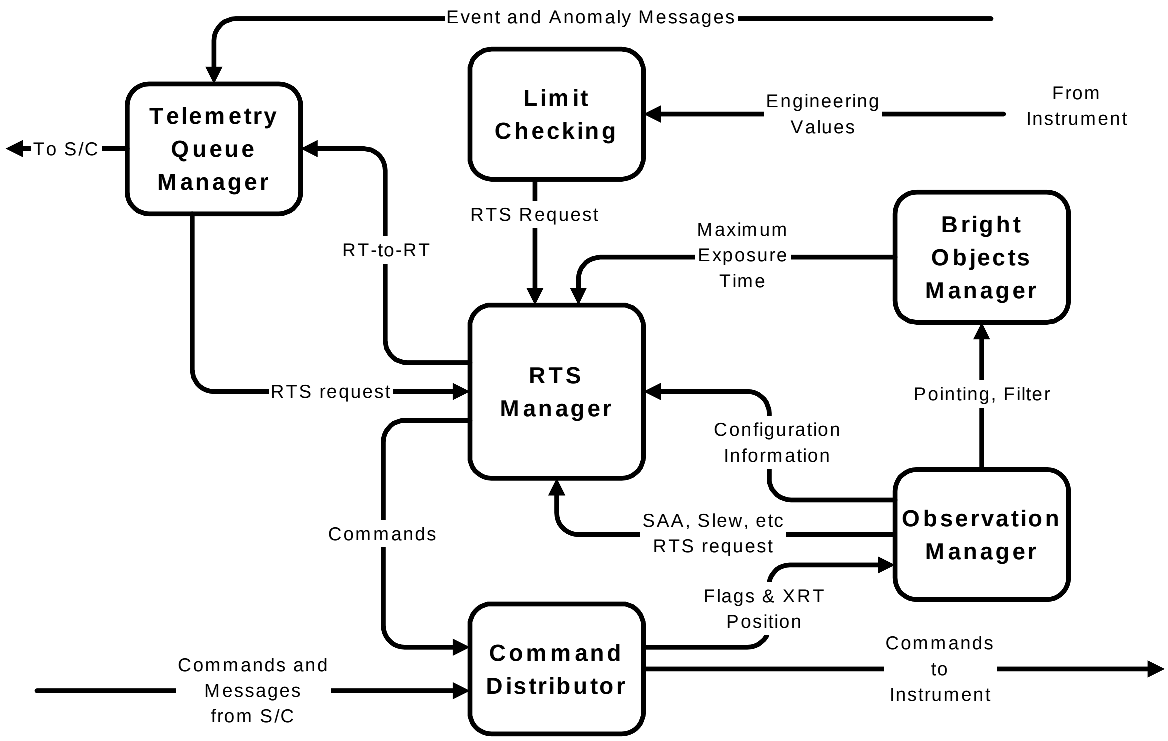

Figure 5: Overview of Autonomous System.

Observation Manager: This module monitors the spacecraft and FOM messages and maintains a record of the status of the ICU. It uses this information to determine when it is appropriate to issue a suitable RTS command. It accesses the state management tables as well as the AT and PT configuration tables described above as part of this process.

Bright Objects Manager: This returns information about planets and stars near or in the target field of view, using the star catalogue and avoidance angle tables described above.

Limit Checking: This monitors critical engineering values. Using the table described above, it issues requests, if necessary, for a RTS to perform recovery actions from a limit failure.

Telemetry Queue Manager: This monitors all outgoing telemetry for error messages and, using the table described above, issues a request to run a recovery RTS if required.

RTS Manager: This module executes the RTS. It consists of the virtual CPU code to execute the RTS, together with software to scan the RTS index image to permit rapid location of a given RTS.

8. TESTING AND VALIDATION

The ICU code must function as part of a co-operating system of processors. A high degree of fidelity to the flight system is therefore required at the development stage. Figure 6 illustrates the in-house environment used to test and validate the ICU software.

The ICU used is the qualification model (QM) ICU. The DPU simulator is a DOS-based PC with a SSI interface card.

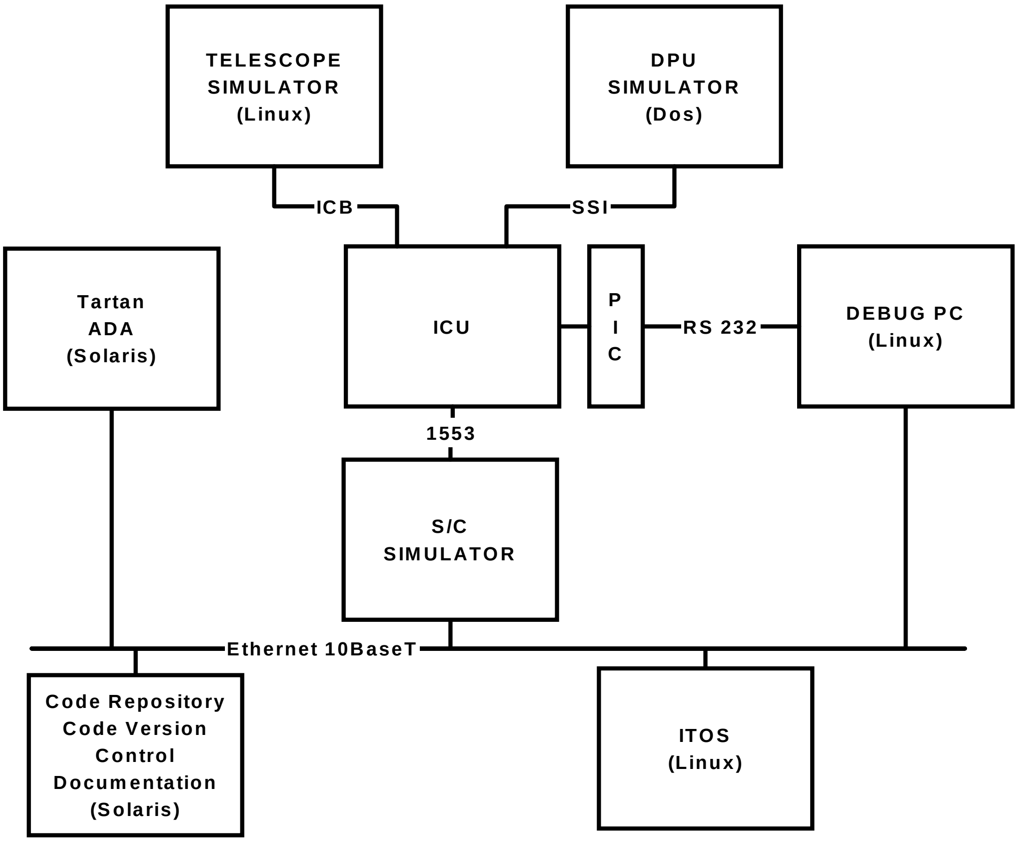

Figure 6: Overview of Testing Environment.

This runs a C-based program that monitors the SSI interface and reacts to any DPU commands received from the ICU with an appropriate response. The spacecraft is simulated either via a Southwest Research Institute or Spectrum Astro supplied Spacecraft Simulator, subject to their availability. The UVOT telescope module is emulated by C-based code running on a Linux based PC containing an interface representative of the ICB. This code was adapted from software developed for a similar purpose for the UVOT heritage instrument, the XMM-Newton Optical Monitor. ITOS is run on a Linux based PC and is used to control tests, display housekeeping and maintain logs. All code is maintained in a Concurrent Version System (CVS) controlled repository on a Solaris based Sun. A debug PC, interfaced to the ICU via a PIC based interface, is used for rapid code load direct into memory and to monitor debug locations.

A comprehensive test plan was devised. As well as checking that all nominal Settling, Finding, Safe Pointing, AT and PT exposures were correctly performed, additional emphasis was placed on the following:

1. Correct early termination, or prevention, of all types of exposures due to the presence of catalogued bright sources or planets.

2. Appropriate interruption of any type of exposure when entering, and a correct recovery on leaving, the SAA.

3. Correct recovery action in the event of a limit being exceeded, an emergency shutdown warning from the spacecraft, any violation of avoidance angle criteria, a loss of spacecraft attitude and internal error messages.

CONCLUSION

The SWIFT/UVOT requirement for on-board autonomous control of exposures and health and safety, together with the anticipated need for flexibility of control during the course of the mission, presents a challenging design problem. The on-board placing of high-level scripts executed by a simply constructed virtual CPU, in circumstances defined by readily reconfigurable tables, has proved highly adaptable in dealing with problems encountered during pre-launch testing. This flexibility is achieved without the need to modify and reload the main Ada code and is further re-enforced by the short time that a reload of a particular table takes. It is anticipated that this adaptability will continue into the post-launch phase, allowing potentially wide-ranging changes to the control of the instrument in order to achieve future scientific goals.

ACKNOWLEDGEMENTS

The authors would like to thank L.K.Gilbert for provision of the star catalogue generation software and the algorithm to search it, E. Auden and A.Breeveld for the preparation of the catalogue datasets and P.Broos and S. Hunsberger for help in establishing the state diagram. Many useful discussions with B.K.Hancock and H.Kawakami concerning the functioning of the instrument are gratefully acknowledged.

REFERENCES

1. Roming, P.W.A, *et al*., “SWIFT Ultraviolet/Optical Telescope”, *Proc. SPIE* **5165**, in press.
2. Hancock, B.K., *et al*., “UVOT Bright Source Safing System”, *Proc. SPIE* **5165**, in press.
3. Levin, J.R., *et al., lex and yacc – 2nd Edition*, O’Reilly and Associates, Inc., Cambridge, 1995.
4. U. S. Naval Observatory, *The Astronomical Almanac*, Government Printing Office, 2001.
5. P. Kenneth Seidelmann (Editor), “*The Explanatory Supplement to the Astronomical Almanac*”, University Science Books, 1992.
6. Peter Duffet-Smith, *Practical Astronomy with Your Calculator - 3rd Edition*, Cambridge University Press, Cambridge, 1979.
7. Jean Meeus, Astronomical Formulae for Calculators, Willmann-Bell, 1985.