Mars Orbiter Mission AOCC Software –
Development, Testing and Mission Aspects

S Sudhakar-Division Head, COED/CEG, ISAC/ISRO

Email: sudhakar@isac.gov.in

FSW-2014

California Institute of Technology, Pasadena, USA

December 18, 2014
Agenda

- Mars Orbiter Mission Flight Software Elements
- MOM AOCC Functions
- MOM AOCC Flight Software Architecture
- Typical Software Design
- Testing Aspects
- Mission Aspects
- Experiences
# MOM Flight Software Elements

**AOCC**
- Attitude Control, Sensor, Actuator Interfaces
- Solar Array Slewing
- Orbit Control Maneuvers, Target Control Maneuvers, Trans Mars Injection, Mars Orbit Insertion
- Autonomy

**Star Sensor**
- Star Sensor
- Inertial Attitude Determination

**TCP**
- Telecommand Processor
- Telecommand, Autonomy

**IRAP**
- Inertial Reference and Accelerometer Processing Unit
- Incremental Angle
- Incremental Velocity

**SSR**
- Solid State Recorder for Payload Data Storage Management
AOCC Configuration

- 31750 Processor Based System
- ASICs for Digital Logics
- Modular Package
- Customised IO Bus
- Interface with Power Sensors Actuators Solar Array Drive

- MIL STD 1553B Interface for IRAP Star Sensors Data Handling TC TM
AOCC Functions

- Attitude and Orbit Control, Controllers, Filters, Estimators
- Sensor Interface - Star Sensors, Coarse Analog Sun Sensor, Solar Panel Sun Sensor, Inertial Reference Unit + Accelerometer
- Actuator Interface – RCS Thrusters, Liquid Engine
- Accelerometer Processing and Thruster Cut off logics
- Onboard Time Reference
- Attitude Reference generation – Model / Profile Based for Pointing, Imaging
- Orbit Reference generation Model Based / Profile Based
- TC – Telecommands, Autonomy Events Interface
- TM – House Keeping TM, Autonomy TM Data Interface
- Mil Std 1553B Bus Control
- Solar Array Drive Mechanism – SPDM logics
- Safety Logics, Fault Tolerant Features, Safe Mode Logics
- Operational Autonomy – Launch Phase Sequencer, LEB sequencer, Payload Sequencer
- Fault Tolerant Autonomy – FDIR logics for Sensors, Actuators I/F
Software Design Challenges

Mars Orbiter Mission-AOCC Software

Schedule

- Longer time for System Reqmts
- Subsequently S/W requirement delayed
- Product release tied to events (Launch window)

Solutions:

- Staggered release of S/W
- Rapid proto-tying
- S/W reusability
- Usage of tools

Multiple Groups

- Requirements from different groups
- S/W size is growing
- Larger Development teams

Solutions:

- Common methodology and process
- Common language – Standard template
- Adhering to the standards religiously
- Disjoint Components as much possible

Quality

- Increasing complexity and Quality Reqmts

Solutions:

- Stringent requirements for Process & Product

December 18, 2014
### Software Design Challenges contd..

#### Needs

<table>
<thead>
<tr>
<th>Needs</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reduced Development Schedule</td>
<td>Methodology, Tools</td>
</tr>
<tr>
<td>Quality</td>
<td>Methodology, Process &amp; Technology</td>
</tr>
<tr>
<td>Multi Group Involvement</td>
<td>Process, Seamless interaction, Common language</td>
</tr>
</tbody>
</table>

- **UML for Design**
- **Customised IEEE 12207 Process Standard**

### Diagrams

- Logical View
- Implementation View
- Use Case View
- Process View
- Deployment View

#### Diagram Elements

- **Class Diagrams**
- **State Diagrams**
- **Scenario Diagrams**
- **Component Diagrams**
- **Model**
- **Deployment Diagram**
- **Use Case Diagrams**
Software Architecture

Round-robin is presently used

- **Advantage**
  - Simplicity, Deterministic

- **Shortcomings**
  - Scalability
  - Fragile for code changes
  - Manual partitioning
  - Manual scheduling
  - Multiple cycles – 8ms/16ms/32ms/64ms/128ms/512ms

---

**Mission & Control Laws**

**Data Selection & Distribution**

**Data Proc & Filter**

**Data Acq Layer**

**Interface Layer**

**Real Time Executive**

**OBC Hardware Layer**

**O/P Data Proc & Distribution**

**Data Delivery Layer**

**Fault Tolerant Autonomy**

**Operational Autonomy**
AOCS Software

- Round Robin type scheduling
- Language Ada with Safe Subset
- Design
  - 64ms Major Cycle (AOCS, Sensors)
  - 8ms Minor Cycle (PWPFM etc.)
  - 512ms Super Cycle (Orbit Model, Autonomy)
- NMI for WDT action
- Provision for Remote Programming
- Review:
  - System Reqmts
  - Software Reqmts
  - Software Design
  - CWT
  - Test Readiness Review
  - Test Results Review Board
## Development of AOCC Software

### Mars Orbiter Mission-AOCC Software

<table>
<thead>
<tr>
<th>Category</th>
<th>Total</th>
<th>Old modules</th>
<th>Modified modules</th>
<th>New modules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Acquisition + TC</td>
<td>19</td>
<td>5</td>
<td>12</td>
<td>2</td>
</tr>
<tr>
<td>Sensor Processing</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>OnBoardTime, Ref Gen</td>
<td>11</td>
<td>0</td>
<td>9</td>
<td>2</td>
</tr>
<tr>
<td>AOCS Modes</td>
<td>11</td>
<td>2</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>AOCS Controllers &amp; MD</td>
<td>6</td>
<td>4</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>Sequencers</td>
<td>7</td>
<td>1</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Actuator I/f and TM</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>FDIR + Autonomy</td>
<td>20</td>
<td>3</td>
<td>6</td>
<td>11</td>
</tr>
<tr>
<td>Initialisation + Main Pgm</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>83</strong></td>
<td><strong>19</strong></td>
<td><strong>42</strong></td>
<td><strong>22</strong></td>
</tr>
</tbody>
</table>

- Reuse of Modules from GEOSAT IRNSS, IRS
- Modification for MOM
- New Interfaces –
- Autonomy and Safety Logics – New Development
- 25% Old
- 25% New
- 50% Modified - Reuse
Software Development Phases

Mars Orbiter Mission-AOCC Software

Requirement Analysis
• Stake Holder Meetings – Joint Requirement Development sessions
• Requirement Listing
• Goals
• Use cases – All Stakeholder’s view
• Functional and Extra Functional Reqmts separation.

Design Phase
• Design for Quick Prototyping
• Component Decomposition
• Function definition
• Interfaces including testing
• Reuse study
• Design inputs, errors, rules

Preliminary Design
• Architecture, Components

Detailed Design
• Component Spec
• Accuracy, Precision
• Call Sequence
• Timing Analysis etc.

Design Phase

Coding

Design Review – SRS, SDD, CWT

Testing

Testing Review – TRR, TRB, CMRB

December 18, 2014
Typical Design - MIL STD 1553B Interface

- Uses Redt Bus Topology.
- Selected AOCC is BC.
- Non selected AOCC is RT
- Message Table for Each Message
- Grouping of Messages of Each RT.
- Each RT is a component
- Messages divided as
  - Time Critical
  - Periodic
  - Asynchronous Messages
  - Event Driven Messages
  - Bulk Messages
- Fixed Scheduling
- Bulk messages dynamically scheduled in fixed slots
Typical Design- Mil Std 1553B I/F contd.

- Maximize Efficiency of Data transfer. - whenever bus is used for data transfer, the data should be utilized.
  
  Data is grouped - Function wise – AOCS, HK, Handshaking
  
  Schedule wise - Time critical, Periodic, Data which require delay
  
  Delay times are effectively used – because of grouping
  
  Low volume Handshake I/F (on 1553) for High volume message Data

- Maximize the usage of the bus for all Interface requirements to avoid other discrete interfaces.
  
  - Schemes for All data transfer Reqmt, Time sync, 1553 Protocol, Minimum Latency I/F

- Minimize the Dynamic Scheduling of data - To simplify design and employs a fixed, determinant, reliable scheduling scheme.

- Minimize Host Processor interactions

- Reliable data transfer
  
  - Data Transfer verification – Protocol wise, Data wise with Checksum, Validation schemes

- Provision of Fault tolerant features
  
  - Bus and BC Changeover Logics
Typical Design: Autonomy Design

- Multiple Levels
  - Subsystem Level, Interface Level, Fault Handling, Operational
- Experience
  - Mission Experience of IRS, IRNSS, GEO
- Identification of Realistic/Perceived requirements
- Validation of Design
- Requirement of the logics at different phases of Mission
- Failure effects of Autonomy logics

Driving Factors:
- Non availability of Contact, Range delays
- Fail Safe Operations.
- Easiness of Operations.

Level 0
- Within Subsystem
- WDT
- Long Pulse Detection
- Reset Handling

Level 1
- Deals with Interfaces
- FDIR for Sensors, Actuators

Level 2
- Acts on Faults if detected
- Safety Logics
- Corrective action on operations

Level 3
- Operational Autonomy
- Safety features
Fault Tolerant Features –

- Fault Tolerant features
  - Hardware
    - NMI, WDT
    - Redundancy
    - EDAC
    - Hardware Filters
    - Thruster Shut Down Timer LPD
  - Software
    - Filters
    - Wild Sample Remover
    - Data Validation, Consistency Checks
    - Remote Programming
    - EEPROM
    - Memory Scrubbing
  - Actions
    - Shutdown
    - Abort
    - Retry
    - Reconfiguration

- Interface F T Features:
  - Data Transfer Consistency
  - Functional Consistency
  - Communication Failure

- Means:
  - Wild sample remover (For all sensors)
  - Validation of samples at least 3 consecutive before taking critical action
  - Checksum in Packet Transfers
  - Validation with Other sensors
  - Valid / Invalid bit provision in I/f
  - In case of Communication failure – Reconfiguration of Bus
Software Fault Tolerance and Autonomy

- **Autonomy features like**
  - Time tagged mode,
  - Event based commanding
  - Configurable command blocks
  - FDIR logics for sensors, Actuators,
  - Launch Phase, Acquisition Sequencer
  - Payload Operation Sequencer
  - LEB Sequencer
  - Master Recovery Sequencer
  - AOCS Safe Mode, Power Safe Mode

- **Software Fault Tolerant Features**
  - Remote Programming
  - Provision to Modify Software
  - AOCS mode changes
  - Modification of Gains
  - Inclusion of new Commands
  - Full Software Main program change
  - Minor Cycle Program Change
  - Majority Voting of critical Parameters
  - Reset Handling logics with EEPROM for Handling inadvertent Resets
  - Event Based Commanding Features
Typical Feature: AOCE Reset Handling Logic

Power On Initialization, Reset logic En → RTE Loop → AOCE Reset Logic

Read & Execute from EEPROM/RAM: POI, Patch, Command Table

S/C configuration after reset (Susp/SafeMode/ChangeOver/No Action)

Event 60 to TCP to indicate completion of POI

Event 61/62

After 1 Sec

TCP

Event 60
Operational Autonomy

- LEB Sequencer for EBN, TMI, TCM, MOI
- Requirements from
  - AOCS, Downlink, Power, Mission, Contingency Plans

### Diagram

**Safety & Contingency Features:**

1. MRS for LEB orientation
2. MRS for Normal Orientation
3. MRS for Fuel saving
4. AOCE Reset Handle
5. Thrust Augmentation Scheme
6. LEB Coil Changeover scheme
Testing Aspects – Different Phases

- Hardware Tests
- Software Unit Level Tests
- Functional checks Input, output, Logical, computational Checks
- Software –HW Integrated Tests
- Open Loop Tests – Static
- Open Loop – Dynamic - SIP – Tests – Simulated Input Profile Tests
- Specific Interface Tests (New I/F)
- OILS – On Board Computer In loop simulation Tests
- HILS- Hardware In Loop Simulation Tests
- Mission Scenario Tests
- Environmental Tests (AOCC)
- Integrated Spacecraft Tests
- Operational Validation Tests
Flight Software Design For Testing

- **Onboard Diagnostics**
  - 1553 Cycle Completion Error Checks and Diagnostic counters
  - Interface Fail Error Counters
  - Mil 1553 Bus Error checks and Fail counters
  - Real Time Executive - Scheduler – Time overrun Indicators
  - Message Status, Bus Status of Mil 1553B
- Time Sync Markers for Time Tag Verification
- Error Codes / Path indicators
- Diagnostic Telemetry like Block verification for upload/ EEPROM loading
- Programmable Telemetry System to Capture Mode dependant TM
- PROM and RAM Checksum Computations
- Onboard Checksum for Command
- Uplink Validation
Ground Test System Design

- Test System Provides
  - All Electrical I/f Power I/f, Loads
  - Measuring Instruments
  - Supports Open Loop Testing
  - Supports OILS Tests
  - Interface verification
  - Timing Verification

  Behaviour as understood simulated

  Short fall – Tested with Actual I/f Tests

- Software for
  - Interface Simulation
  - Sensor Models
  - Actuator Models
  - Orbit Models
  - Dynamics Simulation
  - Automatic Testing
  - Data Logging and Retrieval

Fig 2.0 OILS (On Board In Loop Simulation)
Major causes of error Design /Testing

- Major causes for Failures/observations
- Design Level
  - Requirements inconsistent/not complete/ambiguous
  - Design Errors/Insufficient margins
  - HW/SW Interface Mismatch
  - Accuracy/ Precision requirement mismatch
  - Time synchronization
  - Inexperience & Experience – Overconfidence
  - Error due to heritage assumption

Solutions
- Analysis, Review and Process Implementation
- Test case generation from requirements /Test Review Boards/Test procedure review/ Automation of testing / Test Results Review
- Observations tracking and guidelines/check list generations

Test Level
- Test coverage is not complete/insufficient test cases
- Actuator I/f clearance with Diagnostic TM rather than actual actuator output.
- Insufficient Test – especially less critical ones like TM
- Independent Tests – Interrelated effects missing
- Test Limitations
- Simulation Errors
Software Verification for Mission Operations

Mars Orbiter Mission-AOCC Software

Fig. 1 MOM AOCE –TC Simulator SETUP

AOCE TEST System
- Raw Bus Simulation
- Thruster Load Simulation
- Industrial PC with PCI based Add-On Cards for the following simulations
  - 1553
  - Digital
  - Analog
  - Thruster
  - SADA
  - Potentiometer

Host PC

AOCE TM

Spacecraft Simulator

AOCE TM

AOCE(R)

AOCE(M)

TC Capture

TC Simulator

S/c TM

Mil 1553Bus

TCP/IP interface
Verification for critical Mission Phases

Software Verification in All Mission Phases

- Launch Phase
- EBN, TMI Phase
  - Rehearsal of Operations
  - Profile Validation
  - Time Tagging, Burn Start Time
  - Verification of Modes, OILS
  - FDIR Logics Verification
- Cruise Phase and TCM
  - FDIR, Safe Mode Logics
  - Accuracy Verification
  - Backup Plans Verification
- MOI
  - EBN Verifications +
  - Backup Plan Verifications, Switching of Plans
  - Contingency Plan Verification
- Martian Phase Payload Operations Verification
Typical Mission Operations Verification

TCM Mission Operations Verification

- Identification of Sources of Error
  - Software Sources (Specific Tests to Verify)
  - Command Resolution
  - Cycle Time – Cut off Revisit Time
  - Attitude Reference Selections (Inertial / Profile)
  - Actuator: Thrust Cut off delays, latencies
  - FDIR Logics
  - Validation of Uplink Procedures,
  - Backup Plan for any eventuality

- Hardware Sources
  - Interface
  - AOCS reqmts after cutoff
  - Failure Modes
  - Tail off thrust
  - Verification of Cut off latency –
Experiences- Test As You Fly-

- Safe Mode – Attitude Loss if Rate Exceeds 2.8deg/s.
- Safe Mode Scenario Test revealed
  - On SM First Time Gyro Used for Recovery
  - After AOCC Changeover, Mil 1553 BC reconfiguration is delayed by Design
  - Results in Sync Not sent to Gyro
  - Gyro continues accumulation and saturates incremental Angle
  - **Gyro Logic is**
    
    *If Incremental Angle is above Linear Range => Saturate Incr Angle with Polarity of Angle*

    **Enter Hysteresis Loop**

    *Hysteresis Loop is exit only if Incremental angle magnitude is below 1deg/s*
• Gyro Design uses Incremental Angle polarity for Saturated angle polarity and not actual rate polarity

• If Incremental angle accumulation is overflown, wrong polarity can come.

• Correction: Use Analog Rate info when Incr angle saturates.

• Analog rate parameters were sampled at faster rate for FDIR and available with AOCC.

**CORRECTION:**

If Incr Angle Saturates

Read Analog Rate

Saturate Analog Rate

Convert Analog Rate to Equivalent Incr angle

Incr Angle = Incr Analog angle

End If
Experience From Chandrayaan-1

- Time Tag commands sent through Link2 will not execute when Main Computer is selected for Control at the time of Time maturity
- Due to wrong Mask Pattern usage in internal Time Tag command execution.
- Detected in Ground Checkout

- Initially Scenario was not envisaged
- Whereas during critical Mission ops, requirement was put for JPL to use Link1 frequency for ranging
  ISRO for commanding through link2 frequency with Main computer for control.
- Based on Test Results, this was prohibited and avoided.
Usage of Software Features - EBC

Event Based Commanding provides function

- To Define An Event
- Which is “Any Parameter” to Satisfy Logical Condition >, <, =, Bit wise conditions for consecutive “n” Number of times
- Issue Action – Which is execution of Commands

Extensively Used to Handle critical Autonomy functions which were called for during Mission operations

- Bring in Fail Safe Logics –
- To switch to Redundant Coil in case of Anomaly – during LEB Burn
- To Bring in Safe Mode with Additional Checks felt Necessary after MOI

Variable Address or TM channel Information
Type, Condition, Limits

Event Based commanding
Action-> Event to Execute a command/ A Group of commands
Fault Tolerant Feature for LEB Operations

- To take care of Electrical failure of LEB coil during critical operations plan was put to
  Excite Both Main and Redt coils of LEB engine
- Mission Plan included progressive verification of TMI, MOI procedures during Earth Bound orbits
  - All Features were enabled one by one
  - All FDIR logics were exercised for verifying correctness and No wrong trigger
  - Two Coil excitation was planned for EBN4
- Two Coil Operation – did not operate LE Engine – due to opposing effect of coils due
to wiring configuration
  - Time out / Fail safe logics of LEB sequencer tested – during this operation.
- Alternate option to switch LEB coils worked out with
  Event Based commanding Feature of Software – using delta –V achievement in first few seconds.
MOI Contingency Management

- Battery Energy Balance Might Deplete in case of underperformance leading to longer firing.

- Nominally Battery Safe Mode planned to be enabled after completion of DeltaV achievement

- Battery Safe Mode Checks for Bat Voltage < 32V to declare Safe Mode

- New Logic: Concern was raised, if Battery is already charging no need to declare Safe Mode even if Battery is below 32V.

- EBC introduced for the function.

- Check if Bat Charging is below Threshold, Enable Power Safe Mode
• Simple, Deterministic Proven Software Architecture-Used for MOM
• SW Modular design approach/Base-lining and Reusability
• Defined Software Process
• Use of Experience Self & Others
• Prototyping & Testing : As much close to final configuration as possible
• Extensive Testing and Validation for Mission operations
• Review process for all Stages of Development and Testing

MOM AOCC Software Successfully supported all Phases of Mission.
Acknowledgements are Due to

• Dr. K. Radhakrishnan, Chairman ISRO
• Dr. S K Shivakumar, Director ISAC
• Dr. A S Kiran Kumar, Director SAC
• Dr. M. Annadurai, Programme Director, IRS and SSS ISAC
• Mr. E. Vasantha, DD, Control and Digital Area, ISAC
• Dr. V. Kesavaraju, Mission Director, MOM
• Mr. S. Arunan, Project Director, MOM
• Mr. K. Parameswaran, Satish Dhawan Professor, ISAC
• Mr. Subramanya Udupa, GD, CEG, ISAC
• Mr. Jagannath Das, Division Head, Control Onboard Software Division, CEG
• Colleagues and Team members of
  Control Electronics Group, Control Dynamics and Simulation Group,
  Mission Development Group, Flight Dynamics Group
  Digital Systems Group, Systems Reliability Group
  Power, Mechanism, Fabrication, Integration, Checkout, Project Teams -ISAC,
  IISU, LEOS, LPSC, ISTRAC ISRO Teams

Sincere thanks to
• Dr Allen D Unell, Chairman FSW-2014
• Mr. Subodh Harmalkar, FSW-2014, APL
Mars Orbiter Mission-AOCC Software

Thank You
sudhakar@isac.gov.in

Good Luck, MOM!
From Your JPL Family!
Mars Orbit Insertion, 24th Sept 2014