A PROPOSAL OF AN EXTENSION FOR AADL TO MODEL IMA ARCHITECTURES INTERCONNECTED WITH TTETHERNET

AADL STANDARD MEETING, ORLANDO
Introduction
Objectives
Time-Triggered Ethernet
AADL Extension mechanisms
The Metamodel of the proposed extension
Prototype Implementation of the extension
An Illustrative Example: A model of a subsystem of the Flight Management System (FMS)
Introduction

- Project VerlTTAS
  - ÉTS, École Polytechnique de Montreal, Bombardier and CMC Electronics.
- Incremental integration of avionics functionalities on IMA architectures with TTEthernet
- Developing new technique for modeling Integrated modular avionics (IMA) deployed on Time-Triggered Ethernet platform
  - Applying analysis and verification techniques
  - Safety and performance properties of system
Introduction

- Approach: Model-based Engineering
  - Cope with the complexity of the systems
  - Higher level of abstraction
  - Apply diverse analysis techniques at the model level
  - Verification of safety and performance properties

- Selected modeling language: AADL

- Extension mechanism
  - ARINC 653, Error modeling and Behavioral annex
Objectives

- Presenting our proposal to extend AADL to model IMA architectures interconnected using TTEthernet
- Getting your feedback and orientation
- Improving our work
- Eventual collaboration with a working group to define an annex for AADL for TTEthernet
Time-Triggered Ethernet (TTEthernet)

- Ethernet is used in real-time embedded system
  - IEEE 802.3
  - The flexible and highly scalable protocol
- No reliable temporal performance bound
- Not predictable
- Extension:
  - AFDX
  - Time-Triggered Ethernet extension
TTEthernet

- Communication systems with mixed time criticality application, sharing a single physical network
- Based on Time-Triggered Architecture (TTA)
  - Defines clusters and nodes and establish a fault tolerant global time for clusters

Figure 1: TTA structure [Hermann Kopetz]
TTEthernet

- Characteristics of TTA
  - Implementation of the global time
    - Algorithm for clock synchronisation
  - Off-line computed schedule for the network
  - Consequence: Error detection and fault isolation
  - Uses Time-Triggered Protocol (TTP)
TTEthernet

- TTEthernet standard (SAE AS6802)
  - Defining the algorithms to allow scalable fault-tolerance and self-stabilization mechanisms and clock synchronization
  - Partitioning for mixed-Criticality system
    - Different types of traffics

Figure 2: TTEthernet Standard s

IEEE 802.3 (Switched Ethernet)
Traffic with different timing requirements

- TT traffic: requires limited latency and jitter
- RC traffic: requires limited latency
  - ARINC 664, in particular ARINC 664 part 7
- BE traffic

Applications can use different protocol services and QoS enhancements for communication.
AADL Extension mechanisms

- AADL (SAE Standard AS5506) is an extensible language
- Ability to use external languages
  - Property set
  - Annex extension mechanism
The Metamodel of the proposed extension

- Based on main concepts and characteristics of TTEthernet standard
- Enables the design and analysis of IMA architecture using TTEthernet
- Using Eclipse Modeling Framework (EMF)
  - EMF used for AADL metamodel core
  - Allows integration of dependencies and Java API
  - Eases the expression of the domain concepts dependencies and navigation between
The Metamodel of the proposed extension

- Model of dependencies
The Metamodel of the proposed extension

- TTEthernetAnnex concept
Composed of:

- Synchronization element
- Schedulable resource
- Processing resource
- Physical connection, logical links and channel
Synchronization domain

- Supports system-of-systems communication
- Introduces independent TTEthernet systems
  - Respecting their synchronization
- Two components with different synchronization domain in one TTEthernet network must not be synchronized together (AS6802)
  - The communication between these two component: non time-triggered traffic classes (e.g. RC, BE)
Schedulable element

- **Scheduler**
  - Description of the temporal communication behavior applied to specific synchronization domain

- Defines required scheduling constraints for TTEthernet network
  - In order to produce valid schedule for all TTEthernet traffic
Processing resource
Processing resources

- Processing resources
  - Computing resource: module
  - Networking resource: switch
  - Synchronisation master / compression master: clock synchronisation

- Cluster
  - Several processing resources aggregated into logical group
  - Associated with one synchronisation domain and one synchronisation priority
Connections

- **Connection**
  - Between two physical port
  - Unidirectional or bidirectional

- **Channel**
  - Within the scope of cluster/multicluster
  - Redundancy concept implied by fault-tolerant system
  - Does not share device and communication link with other channel within cluster/multicluster

- **Virtual link**
  - ARINC 664 part 7 (AFDX)
Schedulable resource
Schedulable resource

- Schedulable resource of network not system
  - Partitions
  - Frames
  - Virtual links
    - Synchronized VL, reserved for TT traffic
  - Channels
    - In case of cluster/multicluster
frame

- Protocol Control
- Time-Triggered
- Rate-constrained
- Best-Effort
Textual Syntax

- Limitation of the graphical representation
  - Difficulty to formalize
  - Leakage of precision
- Using grammar
  - Set of rules defining the composition of a language
- Lexer, parser, semantical analyzer
- Emitter for transformation from model to text
- Using the grammar ↔ meta-model mapping definition
Prototype Implementation of the extension
Xtext framework

- Implementing the textual syntax according to an extended BNF
- AADL-TTEthernet metamodel concepts is mapped to a Java implementation
  - TTEthernet objects names are used as class names
  - Attributes are implemented as private fields
  - Classes support the Visitor pattern
    - To traverse the abstract syntax along the composition relationships
Xtext grammar overview

```
grammar org.osate.ttethernet.xtext.Aadltte
import "http://ca.estml.aadl2/aadltte/1.0"
import "http://aadl.info/AADL/2.0" as aadl2
import "http://www.eclipse.org/emf/2002/Ecore" as.ecore

// Partition declaration
Partition returns Partition:
  'Partition' name = ID
  'frames' ':frames += Frame*'
  'end' ID ';

// Frame declaration
Frame:
  RateConstrainedFrame | TimeTriggeredFrame | BestEffortFrame
  | ProtocolControlFrame

// Synchronization Priority declaration
SynchronizationPriority returns SynchronizationPriority:
  'Synchronization Priority' name = ID
  'level' level = Integer ';
  'end' ID ';
```
AADL TTEthernet extension plug-in integrated to OSATE

- OSATE2 four extension point to support
  - Parsing, unparsing, name resolution/semantic checking, instantiation of annex model

- Compiler plug-in
  - Parser/lexer and analyzer

- Integration of plug-in
  - Linking the AADL-TTEthernet plug-in to the OSATE2 annex plug-in
  - Registering parser in the OSATE2 annex registry
AADL TTEthernet extension plug-in integrated to OSATE
An Illustrative Example: A model of a subsystem of FMS

<table>
<thead>
<tr>
<th>Virtual Link</th>
<th>Source</th>
<th>Destination</th>
<th>BAG</th>
<th>Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>VL_1</td>
<td>KU_1</td>
<td>FM_1, FM_2</td>
<td>32</td>
<td>(S_1, S_2), (S_1, S_3)</td>
</tr>
<tr>
<td>VL_2</td>
<td>KU_2</td>
<td>FM_1, FM_2</td>
<td>32</td>
<td>(S_1, S_2), (S_1, S_3)</td>
</tr>
<tr>
<td>VL_3</td>
<td>FM_1</td>
<td>MFD_1</td>
<td>8</td>
<td>(S_2, S_1)</td>
</tr>
</tbody>
</table>

Table 1. Virtual Links details
Demo of FMS
Thanks for attention
Question??