A Model Based Toolset for Supporting Rapid Integration and Verification of Spacecraft Electronics

Anand S Madhusoodanan
Dept. Of Electrical and Computer Engineering
Utah State University
anandsmadhu@cc.usu.edu

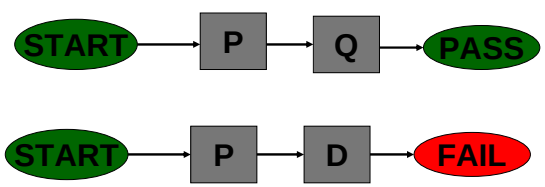
Dr. Brandon Eames
Dept. Of Electrical and Computer Engineering
Utah State University
beames@engineering.usu.edu

Introduction

The goal of the project is to develop tool-based support for rapid spacecraft system integration and verification. Specifically, we focus on the rapid integration of electronic devices using plug-and-play standards (i.e. USB), with the goal of supporting automatic middleware generation for performing I/O operations on these devices. Furthermore, our tool supports the synthesis of a power-on-self-test for the integrated system, allowing users to compare system as assembled through plug-and-play hardware against a previously modeled system specification.

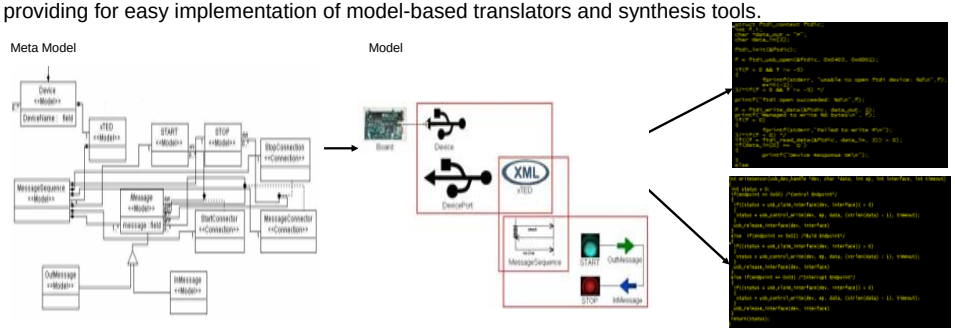
Self Test

A code generator tool, called a model interpreter, parses visual system specification, captured in a GUI tool, and produces both the domain specific middleware as well as a self test for the system. The self test consists of a sequence of messages that a host computer exchanges with each device that is plugged into the USB bus. Valid message sequences are defined as part of the specification of the device, and can be included in the device xTED. The generated self test software ensures that all integrated devices pass their message sequencing test as the system boots.

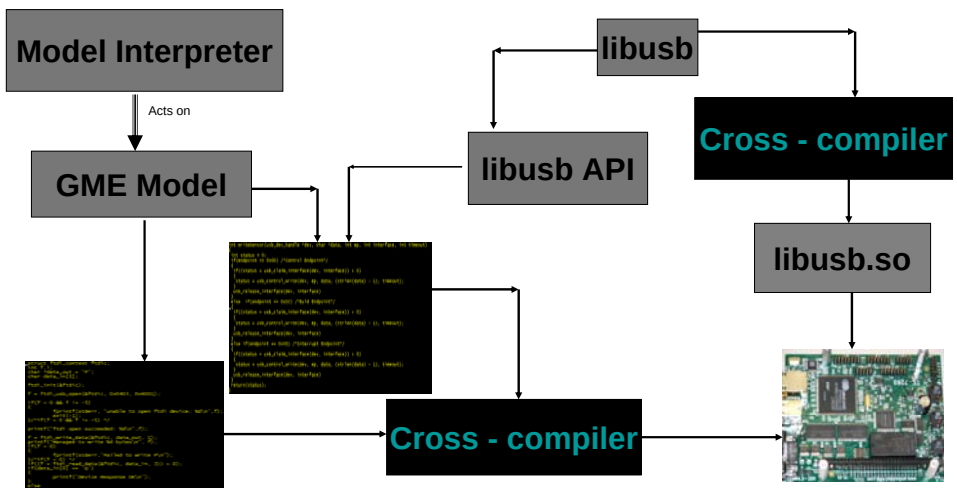


Flexible Tool Framework

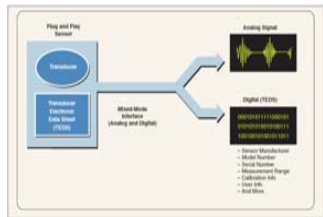
The Generic Modeling Environment (GME) is a customizable modeling framework that allows the creation of domain-specific models and modeling languages, with support for constructing tools for program synthesis and rapid code generation. GME facilitates the creation of a UML-based Meta model, or a model of a domain-specific modeling language. The GME-supplied Meta model translator allows the automatic creation of GME-based modeling tool which conforms to that Meta model. GME also supports an API for acquiring and manipulating information captured in the GUI environment, providing for easy implementation of model-based translators and synthesis tools.



Process Flow



TED



A Transducer Electronic Data sheet (TED) is a document that provides sufficient information about a device so as to allow rapid system integration. In our project, we augment the standard interfacing information contained in the TED with a set of message sequences, which define a protocol for determining the health of the device.
Fig. courtesy Smart plug and Play Sensors, by David Potter IEEE Instrumentation and Measurement Magazine March 2002 pg. 28

DLP Temp G

This is a sensor that was purchased from DLP Design. It consists of a USB–to serial interface. It also has an on–board PIC 12F683 micro–controller that can be reprogrammed. The USB–to serial interface is implemented by the FT232RL USB–UART IC from FTDI.

Phidgets

The Phidget devices used in the project were low cost easy to use sensors. They used the USB interface. They were every simple devices and could not be reprogrammed. Some of the devices were connected to an ADC

TS – 7260

This is a commercially available single board computer purchased from Technologic Systems. It has an ARM–based Cirrus Logic EP 9302 processor. It runs a custom version of Linux.