SpaceWire-PnP: A Quick Refresher

Peter Mendham

22 October 2010
Agenda

› Requirements and aims
› SpaceWire-PnP services
› Extensibility and capabilities
› Known issues
SpaceWire-PnP Aims

› Protocol aims
  › Interoperability and reuse
  › Standard mechanisms for standard features
  › Support device/network discovery as required by SOIS

› Document aims
  › A complete solution
  › A starting point for discussion
Perspective

› PnP views the network like the SpaceWire standard
  › Links
  › Nodes
  › Routers

› Devices

› No topology restrictions
› Both nodes and routers have links
  › Nodes have 1 or more links
  › Routers have 2 or more links
› Every device on the network has a port zero
  › This is the target for PnP transactions
Levels of Support

- **Managed Networks**
  - Important role for system designer
  - Competition during discovery process removed by design
  - Competition for configuration of devices removed by design
  - Simplest case

- **Open Networks**
  - Network handles all competition issues
  - Deals with networks where design is **not** known *a priori*
  - More flexible but more complicated
What is Standardised?

- A set of parameters on the target
  - This is a standardised RMAP address space
- An interface of primitives at the initiator
  - Satisfying the requirements for SOIS
- A description of how the initiator and target will both behave
Core Services

Four core services defined

- **Device Identification**
  - Read-only, constant fields
  - A few, mirrored, read-only dynamic fields

- **Network Management**

- **Link Configuration**
  - All devices

- **Router Configuration**
  - Routers only

Optionally, there is also a time-code source

Basic discovery
Satisfies SOIS
Necessary for SpaceWire-specific configuration
SpaceWire-PnP Extensibility

› SpaceWire-PnP is a convenient mechanism for detecting and configuring
› Can it be used as a “gateway” to more functionality?
› Devices can define their **capabilities**
  › Identifiable feature set
  › Supported by a SpaceWire-PnP service
    › Parameters
    › Primitives
    › Permits identification and configuration of the capability
Capabilities

- Device can provide a list of *capabilities*
- Capabilities based on protocol ID
  - A protocol which is supported
  - Optionally “transported” over another protocol
  - Supports nesting of “transports”
- Examples
  - CPTP over SpaceWire-(R)T
  - A standardised address space “transported” over RMAP
Describing RMAP Address Spaces

- SpaceWire-PnP document proposes a method for describing RMAP address spaces
- Capability services allow the description of:
  - Memory regions which exist to receive data: **data sinks** (e.g. actuators)
  - Memory regions which permit access to generated data: **data sources** (e.g. sensors)
- Also permits non-trivial access mechanisms
  - Delayed response reads and writes
  - Initiated reads and writes
Summarising SpaceWire-PnP

- Protocol utilising RMAP
- UoD document available: SpaceWire-PnP v2.1
  - Since February 2010
- Defines
  - Target parameters
  - Initiator primitives (service interface)
  - Behaviours (algorithms) where necessary
- Simple
- Does not require extra feature support
- Flexible and extensible
  - Can use capability services to extend support
Where next?

› Feedback on the document from the community
  › Just level 1 (?)
› Turn feedback into proposed revisions

› Bread-boarding/prototyping
  › Already some work done by SciSys, Aeroflex Gaisler and others
Known Issues (1/3)

› Need for clarification and further investigation
  › Capabilities
  › All of level 2

› Points added pending changes/clarifications to SpaceWire standard
  › Time-code sources
  › Interrupt handling
Known Issues (2/3)

- Standard way to handle not-implemented parameters/fields
  - Not clear at the moment
- Defined way to handle vendor-specific additions

- What about “dead space” in the memory map
  - Is it safe to read and ignore this?
  - Should there always be no side-effects on read?
    - This also relates to the ability to retry
Known Issues (3/3)

› Issues with particular fields, e.g.
  › Mirrored fields
  › Region field
  › Port types
  › Link errors
  › Link state … and more

› Terminology
  › Links, ports, nodes, routers, services
  › Needs to align with updates to standard and to SOIS