IEEE 1451.4 Standard Working Group
Telephone Meeting, Nov. 9, 2000
Meeting Minutes, issued 11-09-00, Approved 08-23-01

Chair: Torben Licht
Secretary: Paul Hufnagel

Attendance:
Allen Broad, Crossbow, abroad@xbow.com
Mark Buckner, ORNL, bucknerma@ornl.gov
Steven Chen, Aeptec, schen@aeptec.com
Geraldine Couffignal, Airbus, Gilles.Freaud@airbus.aeromatra.com
Mike Dunbar, Crossbow, mdunbar@xbow.com
Gilles Freaud, EADS Airbus, Gilles.Freaud@airbus.aeromatra.com
Fernando Gen Kuong, Endevco, fernando@endevco.com
Brett Gidge, Wilcoxon, brett@wilcoxon.com
Paul Hufnagel, Kistler Instrument Corp, phufnagel@kistler.com
Charles Jones, Edwards AFB, Charles.Jones@edwards.af.mil
Torben Licht, B&K, trlicht@bk.dk
Carlos Lopez-Reyna, Scitefair, clopez@ee.upenn.edu
John Mark, Kistler Instrument Corporation, jmark@kistler.com
Joe Que, Endevco, jque@endevco.com
Bruce Swanson, Endevco, bruce @endevco.com

1) T-Block: C. Lopez-Reyna
    a) Communication from CLR:
    ****

Chapter 8: T-Block

8 T-Block Introduction

This chapter defines the Transducer Block (T-Block) that represents transducers compliant with the 1451.4 standard in accordance to the 1451.1 object model.

The 1451.4 T-block provides a common object interface between transducers and application software. It includes methods for extracting and encoding1451.4 TEDS data, configure Signal Conditioner and NCAP parameters and properties to access the extracted transducer information. {Consideration if T-Block can exist in non 1451.4 NCAPs}

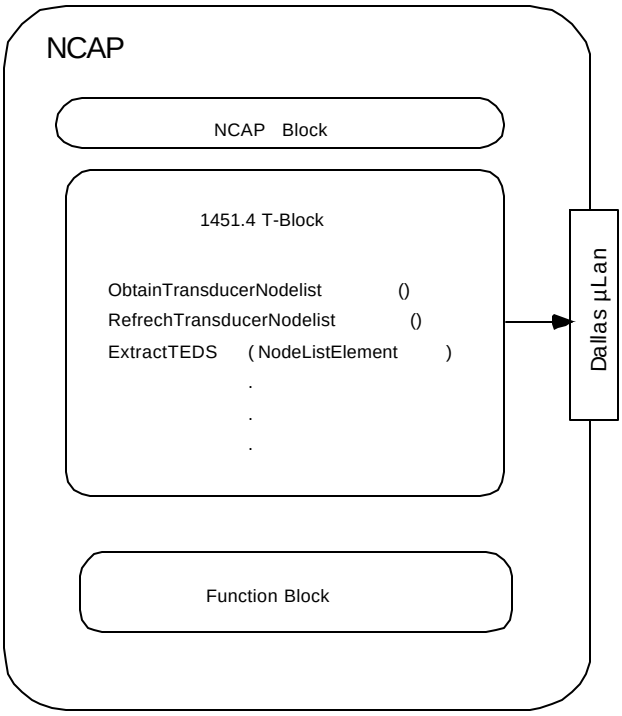


Figure 1. NCAP and T-Block object mapping to 1451.1 object model

8.1 T-BlockClass definition

Class:IEEE1451_Dot4TransducerBlock
Parent Class:IEEE1451_TransducerBlock
Class ID: X.X.X.X.X.X
Description: The T-block class (IEEE1451_Dot4TransducerBlock) provides a common object interface for transducers compliant with the 1451.4 standard.

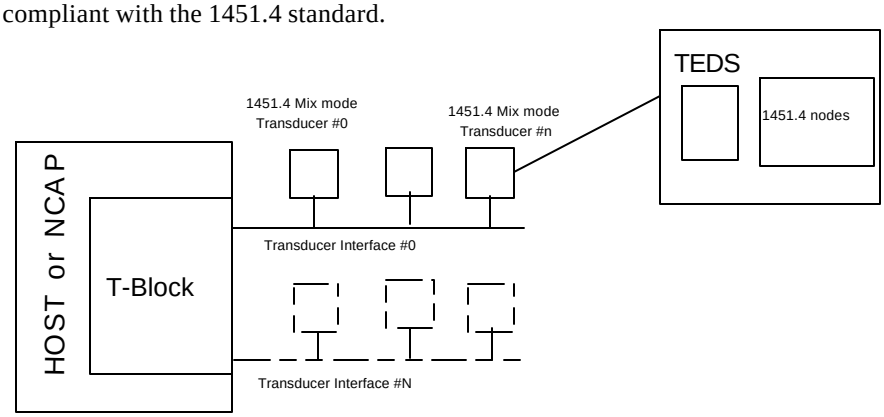


Figure 2: 1451.4 component list

The T-block is responsible for mapping the encoded TEDS data from the transducer into a set of previously defined properties using the relationships contained in the TDL template. {*The T-block keeps and manages a node list with at least one 1451.4 transducer. The elements of the node list are represented by node class objects owned by a 1451.4 transducer interface class. Transducer properties are mapped directly to the node object properties and accessible through T-block methods.}* Figure 2 present the 1451.4 T-Block class diagram using standard UML[1] notation.

[1] Universal Modeling Language (UML)

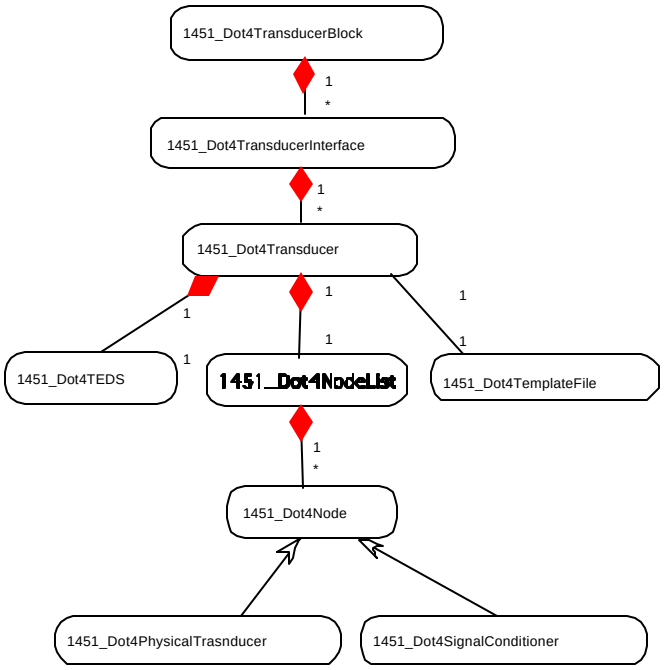


Figure 3:1451.4 Transducer Block (T-Block) UML class diagram

The T-Block is composed of one or more 1451.4 transducer interfaces. Each 1451.4 transducer interface corresponds to the physical interface described in chapter 4 of the standard. The T-Block is responsible for the creation and destruction of all its transducer interface members (strong composition). {Discuss strong composition, association and aggregation}

The 1451.4 transducer interface is strongly composed of one or more 1451.4 transducers. A 1451.4 transducer is an entity containing the TEDS and at least one node that meets the 1451.4 specification. See Chapter 4.1

Each 1451.4 transducer is strongly composed of a single 1451.4 compliant TEDS, an optional Template file and a node list with at least one node. Strong composition excludes sharing ownership of 1451.4 transducers across 1451.4 transducer interfaces. Grouping of 1451.4 transducers is allowed within the same 1451.4 transducer interface. {Comment of composition across transducer interfaces using the T-Block, additional directives to group transducers and nodes in DL needed, change wording in 4.2}

The 1451.4 node list is an indexed collection of 1451.4 nodes. The 1451_Do4Node class is a generalization of the 1451_Dot4PhysicalTrasducer and 1451_Dot4SignalConditioner classes. The 1451_Dot4PhysicalTrasducer class represents the sensing elements (accelerometers, microphones, etc) while the 1451_Dot4SignalConditioner deals with signal conditioning elements (preamplifiers, multiplexors, etc). {Discuss multiple inheritance to build transducer with built in signal conditioners}

8.1.1 Class Summary

8.1.1.1 Operation specifications: Network Visible operations

List operations that are visible from the NCAP network interface

8.1.1.2 Operation specifications: Inherited operations

List operations that are inherited from 1451.1 transducer block class

8.1.1.3 Operation specifications: Local Operations

M.M T-Block class behavior specification

Description

The IEEE1451_dot4TransducerBlock class and its members observe different behaviors[2] that uniquely identify the state of the system. T-Block Membership classes recognize digital and analog states {a mix analog/digital state is possible and open for discussion, This is extra wire/wireless case}

M.M.2.1 IEEE1451_dot4TransducerBlock Behavioral model

The IEEE1451_dot4TransducerBlock is the base class of the T-Block case. When first bound the IEEE1451_dot4TransducerBlock defaults to it's **Initialize** state (See figure x). The **Initialize** state creates the T-Block object hierarchy and sets all member objects to their default start-up state. When initialization is complete the IEEE1451_dot4TransducerBlock transitions to the **IDLE** state and awaits messages from the host. {More to come}

Defined behaviors

DestroyTBlock: When the IEEE1451_dot4TransducerBlock transitions to the DestroyTBlock state it stops listening to host messages, finishes it's last received message and broadcasts a shutdown message to all it's member objects. Once all member objects have been destroyed the IEEE1451_dot4TransducerBlock is unbound.

ConfigureTransducerInterface:
More states to be defined

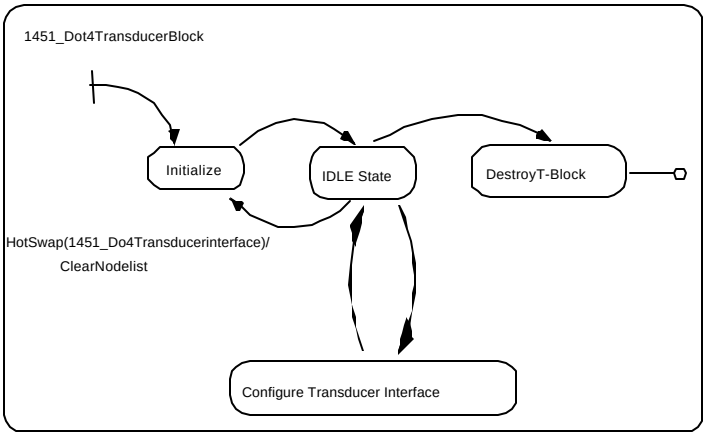


figure 4: 1451_Dot4TransducerBlock state chart

M.M.2.1.1 1451_Dot4Transducerinteface

The 1451_Dot4Transducerinteface class determines the operation of the communication link between the IEEE1451_dot4TransducerBlock and the 1451.4 mix mode transducers. Two basic states have been defined for the 1451_Dot4Transducerinteface: Analog and Digital modes. The NCAP or host can switch

[2] .Behaviors for the 1451.4 T-Block are presented using standard UML state charts .

the state of the 1451_Dot4Transducer interface using the methods provided by the IEEE1451_dot4TransducerBlock. The 1451_dot4TransducerBlock is not responsible for detecting legacy transducers in the 1451_Dot4Transducer interface.

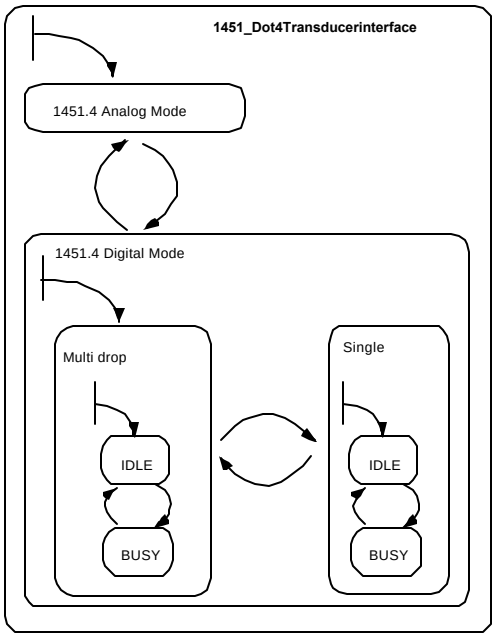


Figure 5:1451.4 Transducer interface

M.M.2.1.1.1 **Defined behaviors**

Analog mode: This is the default and start-up mode of the 1451_Dot4Transducer interface. While in this mode no digital operations are allowed. When the 1451_Dot4Transducerinteface is in analog state it rejects all messages from the **IEEE1451_dot4TransducerBlock** that require digital operations on the transducer interface and triggers an error message (operation not complete) to the T-Block. Mixing legacy transducers with 1451_Dot4Transducer is possible in this mode.

Digital mode: The digital mode supports two sub-states: Multi drop and Single. In digital mode the 1451_Dot4Transducerinteface allows all operations pertaining to the Dallas command library. While the transducer interface is carrying a digital operations it transitions to the busy mode otherwise remains on the idle state.

• **Single sub-state**: In this state the 1451_Dot4Trasnducerinterface expects a single 1451_Dot4Transducer. Analog and digital sub states are allowed.
• **Multi drop sub-state**: Multiple 1451_Dot4Transducers are allowed on the 1451_Dot4Transducerinteface. Only 1451_Dot4Transducers with analog switching capability are allowed in a multi drop configuration.

{Multi wire mode: Analog and digital operations are allowed simultaneously on the interface. Mix Analog/Digital:???? {Need to discuss the extra wire case.)}

M.M.2.1.1.2 Control of the transducer interface

The NCAP or host can request the T-Block to release a transducer interface to handle the analog signal itself. Before the transducer interface is released the T-Block will finish all digital communications and switch the transducer interface to analog mode. Alternatively the T-Block can use its analog monitoring methods to digitize (if supported by hardware) the analog signal (See T-Block analog monitoring method).

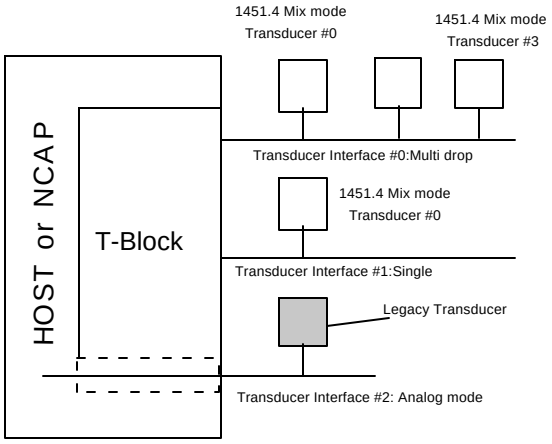


figure 6: 1451.4 Transducer interface example

M.M.2.1.2 1451_Dot4Transducer

The 1451_Dot4Transducer class observes two basic behaviors: analog and digital. The default start up state for a 1451_Dot4Transducer is in analog. The behavior of the 1451_Dot4Transducer is entirely determined by the T-block and the transducer interface.

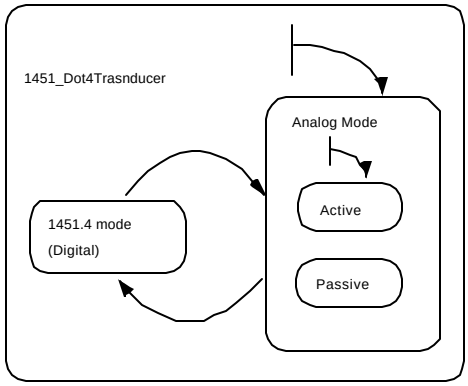


Figure 7: 1451_Dot4Transducer behavioral diagram

Defined behaviors

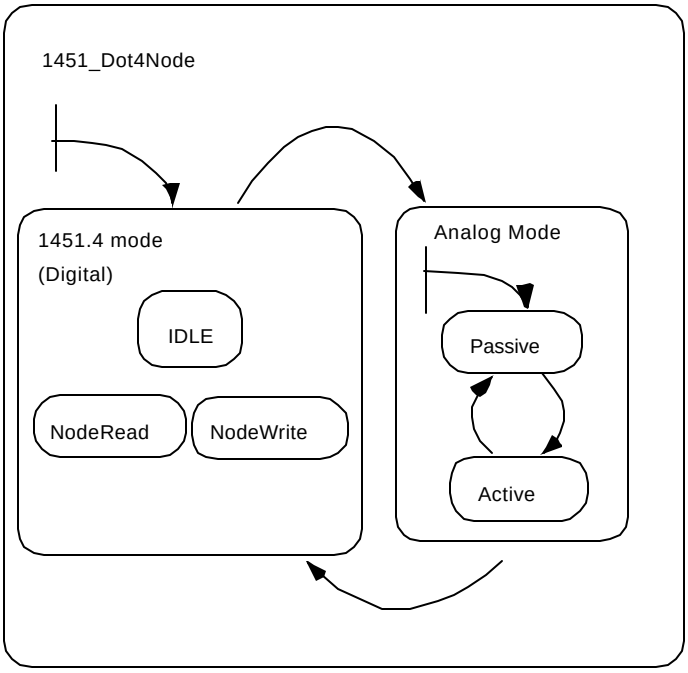
Analog mode: This is the default start-up state for 1451_Dot4Transducers. Two sub states are defined in this mode.

Active sub state: Internal electronics of the 1451_Dot4Ttransducer allow analog sensor voltages to flow into the transducer interface. {If this is the default start up state then there is a potential conflict with multi drop configuration if the transducer interface starts in legacy analog mode. Solution is to set the 1451_dot4Transducer interface always in digital mode when in Multi-drop configuration}

Passive sub state: Internal electronics of the 1451_Dot4Transducer block analog sensor voltages to flow into the transducer interface. {If this is the default start up state there could be a problem with legacy systems not seeing the transducer}

Digital mode: Supported Dallas 1-wire digital operations are valid in this mode.

M.M.2.1.2.1 1451_Dot4Node



M.M.2.2 New Transducer detection

M.M.2.3 User access (calibration, user, etc)
M.M.2.4 Physical Units

M.M.2.5 Shut down

****

b) The T-Block always leaves the Transducer Interface in the analog mode. There are no indeterminate states.
c) Request for examples of node lists from any members who have used them.
d) Typo in sec M.M.2.1.2, Analog Mode, Active Sub-state, Reference to flow of voltage. Should refer to flow of current. CLR to revise.
e) A section has been added to allow digital-only nodes.

2) Electrical Specifications:

a) Question on source of remarks on relationship of speed, cable length and drive current.
b) Remarks a paraphrased from an application note by Dallas Semiconductor and are quite general and not mathematically rigorous. The low data rates presently in use probably require no more than this.
c) Solicitation by Airbus for data on 1451.4 receiver/transmitter designs.
d) Airbus will forward implementation questions to T. Licht, who will dispatch them to the Working Group.

3) Other Sensor (Actuator) Types: J. Que, B. Swanson

a) Request for membership to forward opinions on concessions accommodating other sensor and actuator types in Standard 1451.4. To date all manufacturers on the Working Group have had an interest in PE applications, and manufacturers of other types have not shown interest in 1451.4 TEDS.
b) One concession is the addition of a dedicated TEDS wire, which is mentioned in the Draft. Only one paper has been received (From Somat), discussing the implementation of TEDS for bridge sensors, and the need for a separate connection for TEDS.
c) The Standard cannot specify all commonly used connectors.

d) Transducer technologies are electrically incompatible by their natures, and will require separate definitions in the Standard, and separate TEDS Templates. The T-Block can accommodate any sensor type, but some sensor types may not be compatible with 1451.4 implementation.
e) It may be possible to place TEDS in parallel with a bridge element.

4) Next Face-to-Face meeting:
a) Action: K. Lee to forward a list of hotels to Working Group.
b) **Action: S. Chen, T. Licht and K. Lee will plan an agenda, to be forwarded to IEEE.**

5) Next Telcon, 11-16-00: Adjourn: 12:01pm EST