Vehicle-Bus Interface for Data Collection

David C. Pheanis and Jeffrey A. Tenney
Computer Science and Engineering
Arizona State University
Tempe, Arizona 85287-5406
David.Pheanis@asu.edu

Abstract

Nearly every vehicle that General Motors (GM) manufactures today has a vehicle bus known as the *Class-2 Bus*. The Class-2 Bus is a single-wire, ground-referenced, voltage-driven implementation of the SAE J1850 bus, and it links all of a vehicle's various processors or *nodes* into a vehicle network. The Class-2 Bus uses Variable Pulse Width (VPW) modulation and implements Carrier Sense Multiple Access with Collision Resolution (CSMA/CR). Because of the VPW protocol, the Class-2 Bus is a variable-speed bus, and it has an average transmission rate of 10.4 KBS (10,400 bits per second).

The primary purpose of the Class-2 Bus is to improve reliability while simultaneously lowering cost by reducing the numbers of wires, connections, and special-purpose circuits in a vehicle. The Class-2 Bus also promotes synergism among the vehicle nodes to provide new features that would simply not be feasible without the bus.

We have developed a Vehicle-Bus Interface (VBI) that allows us to collect test data with a General-Purpose Data-Acquisition System (GPDAS) directly from the vehicle's nodes via the Class-2 Bus. The VBI facilitates vehicle testing by greatly reducing the need for external sensors and transducers.

1 Introduction

Microcontrollers are changing the automotive industry in two related areas. The first area concerns the computer control built into today's vehicles. Microcontrollers are controlling vehicle components such as the engine, the antilock braking system, the transmission, the instrument panel, and the entertainment center, to name just a few. A microcontroller does more than just making sure its own component works correctly — it also participates in one of possibly several networks in the vehicle. These vehicle networks exist to allow the sharing of information among components and to reduce the numbers of wires and sensors required in the vehicle.

The second area in which microcontrollers are changing the automotive industry concerns the way we interact with our vehicles. Computers are now infiltrating the passenger area, and consumers can now interact with their vehicles in new ways. The first phase of this change merely allows users to run already-familiar programs in the vehicle — email, GPS navigation, calendar management, etc. The second phase of this migration allows us to interact with the vehicle's on-board controllers, gathering information, controlling the vehicle directly, and aiding in the diagnosis of problems.

Until now, automotive testing by vehicle manufacturers has required the expensive and time-consuming installation of myriad sensors and transducers to collect data from the vehicle. Now, however, we can interact with the vehicle bus to collect a wealth of data from sensors and controllers that already exist on the vehicle. This paper discusses a Vehicle-Bus Interface (VBI) that allows GM's General-Purpose Data-Acquisition System (GPDAS) to collect data by interacting directly with the nodes on the Class-2 Bus that exists on GM vehicles.

2 The Class-2 Bus Protocol

The SAE J1850 Vehicle Standard [1] specifies a bus for use in vehicles, and General Motors implements its version of that bus as the Class-2 Bus on GM vehicles. The Class-2 Bus achieves a single-wire implementation by using the vehicle chassis as a ground reference. A single-wire bus can connect all of the nodes in a vehicle with less than 130 feet of wire, so it provides a tremendous reduction in cost while greatly improving reliability. As a point of reference, the 1993 Corvette, a sophisticated vehicle without a Class-2 Bus, contains more than three *miles* of wire weighing over 200 pounds and has more than *fifty* wires running from the driver's door to the rest of the vehicle! Every wire is a wire that can break, and every connection is a connection that can fail. Wires add cost to the vehicle, and the sheer weight of the wire impacts fuel economy.

The Class-2 Bus uses Variable Pulse Width (VPW) signals, so an active state (7 volts) can represent either a one bit or a zero bit while a passive state (ground, not driven) can also represent either a one or a zero. A transmitting node actively drives the bus to 7 volts to achieve the active state, and a transmitting node simply does nothing to "transmit" a passive state. If multiple nodes attempt to transmit simultaneously, therefore, the active state predominates.

An active pulse with a width of 128 microseconds represents a zero bit, and an active pulse with a width of 64 microseconds represents a one bit. Conversely, a passive period with a width of 128 microseconds represents a one while a passive period with a width of 64 microseconds represents a zero. Active pulses and passive periods with various other widths provide framing information to delimit data packets on the bus, and the bus remains in the passive state when the bus is idle.

Note that a zero bit is wider than a one bit in the active state while a zero bit is narrower than a one bit in the passive state. Therefore, zero bits predominate over one bits at the beginning of a message if multiple nodes begin to transmit simultaneously. The Class-2 protocol takes advantage of this situation by starting each message with a priority field followed by a target-node address and a source-node address. Lower priority numbers and addresses take precedence over higher priority numbers and addresses, so priority messages to or from nodes with priority addresses automatically take precedence over other messages.

We use a Motorola MC68HC58 Data Link Controller (DLC) to interface between a processor and the Class-2 Bus. The DLC can provide either a synchronous serial interface or an eight-bit parallel interface to the processor. Each node on the bus has its own DLC, so the bus interface is uniform from node to node.

Each node that transmits on the bus also listens on the bus at the same time. If a node's DLC receives an active state when the node is "transmitting" a passive state, that DLC loses arbitration and immediately ceases its transmission without affecting the dominant transmission that is on the bus. Eventually, one node wins arbitration as all others drop out, and the winning node's message is the only message that appears on the bus. This mechanism implements Carrier Sense Multiple Access with Collision Resolution (CSMA/CR). The appropriate message drives the bus, and we resolve collisions without damaging that message on the bus. Thus there is no need to retransmit the dominant message following a collision.

The DLC automatically appends an eight-bit CRC (Cyclical Redundancy Check) to each message that it transmits, and the DLC validates the CRC when receiving a message. Since each node receives all messages, including the messages that it transmits itself, a node's DLC automatically retransmits any message that doesn't transmit properly the first time. This scheme makes the bus reliable even in the electrically noisy environment of the vehicle.

Since a one bit or a zero bit can each require either 64 or 128 microseconds, the Class-2 Bus is a nondeterministic bus in the sense that the bus speed varies according to the particular patterns of bit values in the messages. On average, each bit requires 96 microseconds, so the average speed of the Class-2 Bus is 10.4 KBS (10,400 bits per second). A message on the Class-2 Bus can contain a maximum of twelve bytes, including the CRC byte, and the average length of a bus message is approximately seven bytes. The bus can therefore typically handle on the order of 100–200 messages per second. Note that a single message can contain as many as six different data values, so the data rate on the bus is higher than it might initially appear to be.

3 Bus Synergism

Besides reducing cost and improving reliability, the vehicle bus also allows various vehicle nodes to work together to provide features that simply wouldn't be feasible without a bus. For example, the Powertrain Control Module (PCM) can broadcast the vehicle speed on the bus, and the node that controls the entertainment center can adjust the volume of the sound system automatically according to vehicle speed to compensate for increased or decreased road noise. The Antilock Braking System (ABS) can enhance vehicle control and safety by reacting to data values that the PCM broadcasts. The cell phone can broadcast its status, so an incoming or outgoing call on the cell phone can automatically mute the radio.

As another example, using the remote keyless entry system to unlock the doors can automatically activate the interior lights and adjust the power seat and entertainment center to the settings for the driver who unlocked the doors. The interior lights can go off automatically after the engine starts, and the vehicle can automatically lock its doors after reaching a speed of 15 MPH. Later, removing the key from the ignition after stopping the vehicle can automatically unlock the doors. Opening the door to leave the vehicle can then turn off the sound system, and the convenience lights can go off at a programmed interval after the driver and passengers leave the vehicle and close the doors.

We are only beginning to reap the advantages that are possible with the availability of communication among the various vehicle nodes. The vehicle bus will soon foster features that we can't even imagine today.

4 GPDAS Overview

General Motors has developed and now uses a General-Purpose Data-Acquisition System (GPDAS) [2] to collect data from vehicles under test. The GPDAS is a real-time, embedded, distributed-processing system that collects, displays, and stores data to help engineers design better cars and trucks. The GPDAS consists of a twelve-slot chassis, a laptop computer known as the *controller*, and five types of plug-in cards.

The first plug-in card is the Main Chassis Processor (MCP) card. The MCP card is the master-control card for the chassis, so the chassis always contains one MCP card. The other four plug-in cards are signal-conditioning cards that interface to transducers and other data sources, and the chassis can contain any suitable mix of as many as eleven signal-conditioning cards. The four types of signal-conditioning cards are: (1) temperature card, (2) strain-gauge card, (3) analog card, and (4) vehicle-bus interface (VBI) card.

5 VBI Card

The VBI card is a new card that provides an interface to the Class-2 Bus, thus allowing the GPDAS to obtain data from vehicle nodes and from sensors that are already part of the vehicle.

CARB, the California Air Resources Board, requires vehicle manufacturers to equip each vehicle with a port that allows an emissions tester to connect an instrument to the vehicle to obtain and validate the vehicle's emissions history and performance. GM meets this requirement by providing a connector to the Class-2 Bus below the steering column. Serendipitously, we use this same connector to attach the GPDAS's VBI card to the vehicle's Class-2 Bus, so preparing to collect Class-2 data from a vehicle is as simple as plugging a cable into an easily accessible connector. Compare this happy circumstance to traditional testing, which typically requires several days of preparation to install transducers and wiring on a vehicle, often damaging the vehicle and making it unsalable in the process.

The VBI card contains a Motorola 68376 microcontroller and a Motorola MC68HC58 DLC. The DLC interfaces to the Class-2 Bus, and the VBI card acts as a test node on the bus. As a test node, the VBI card can request real-time data values from other nodes on the bus, and it can also command other nodes on the bus to perform various actions. Potentially, the VBI card could even flash new data tables or programs into vehicle nodes to change the characteristics of a vehicle for test purposes.

The VBI card allows the GPDAS user to record as many as 96 channels of information at sample rates as fast as 40 Hz. A *channel* is a single data value such as engine RPM, throttle position, coolant temperature, lateral acceleration, etc. We must limit the amount of information that a user can record from the Class-2 Bus because of the limited bandwidth of the bus. Remember that the vehicle itself uses some of the bus bandwidth (usually 20% or less) for its own operation, and we must take care to avoid affecting the very performance that we want to measure.

6 VBI Data-Collection Features

Collecting data from the Class-2 Bus presents a special challenge because of the nondeterministic nature of the bus. Not only is the raw bus speed data dependent at the bit level, but the bus is also nondeterministic in a much greater sense because a variable number of nodes with differing performance characteristics and unpredictable agendas share the bus.

The design of the VBI software carefully considers the fact that we can't depend on receiving data according to the delivery schedule that the test engineer has requested. We must tolerate late responses and missing responses, and we must also accommodate bus traffic that isn't intended for us. We classify some problems as harmless problems and make them transparent to the GPDAS user, and we report serious problems while taking special steps to recover from them automatically.

Scheduling Data Requests. When we schedule outgoing messages to request data values from vehicle nodes, we use *Anticipative Message Staggering*. We want to transmit a request for data far enough ahead of the scheduled data-delivery time to have a good chance of receiving the data before it is due, but we don't want to request it so far ahead that we end up with stale data. We assess our data-delivery schedule as it compares to our own usage of the Class-2 Bus and expected response times. This analysis tells us how far in advance to request data values.

We need to avoid flooding the Class-2 Bus with too much traffic at any one time because the bus has limited capacity and because we must share that capacity with the vehicle's normal operating traffic. Unfortunately, the natural tendency is to flood the bus with requests for data shortly before the time that the test engineer has specified for the collection of data values from several interrelated sources. We alleviate this problem by staggering message transmissions for channels due at the same time instead of simply requesting all data values at the same time.

Preventing Bus Monopolization. We allow the GPDAS user to set the priorities of outgoing messages from the VBI card to the vehicle. A user who chooses priority zero (the highest priority) could potentially dominate the bus with high-priority messages from the VBI card, preventing other nodes from transmitting the normal messages that operate the vehicle. The VBI software eliminates this problem by implementing unique message timing that provides a small but sufficient window of opportunity for another node to transmit after each message that the VBI card transmits. Thus the vehicle, which typically needs only 20% of the bus bandwidth, can always get up to 50% of the bus bandwidth regardless of the priorities that the test engineer specifies for test messages. At the same time, the VBI card automatically absorbs as much of the bandwidth as it needs if the vehicle doesn't consume the bandwidth.

Persistent Acquisition. The VBI card keeps track of all the identified nodes in the vehicle. We monitor each node's transmissions of State-of-Health (SOH) messages. We also detect a problem when a node has terminated diagnostic operation while we are using a collection scheme that requires the node to be in diagnostic mode. If we ever notice a loss of health or a premature termination of diagnostic mode, we immediately take corrective action. In many cases we can restore the node to the desired mode of operation automatically without user awareness. In other cases, we must report a loss of data due to the faulty node.

Dynamic Rejection Recovery. All data collection requires successful transmission of messages from the VBI card and successful processing of those messages by the vehicle nodes. When a node cannot successfully process a VBI message, the node sends a rejection message to the VBI. Usually, this rejection affects data collection by voiding one or more channels. Sometimes, however, the rejection also affects subsequent messages, both in content and in scheduling. The VBI dynamically and automatically alters message contents and schedules to recover as gracefully as possible.

7 Conclusion

The addition of the VBI card allows GPDAS users to collect low-speed test data directly from a vehicle by simply connecting a cable to an easily accessible connector that is already on the vehicle. This convenience greatly reduces the need for the time-consuming and expensive installation of external transducers for data collection.

Additionally, the VBI card provides access to internal data values that simply aren't obtainable with external transducers. With the VBI card, the tester can determine not just what the vehicle is *doing* but also what the vehicle is *"thinking"*.

The VBI card has limitations, of course, since it provides relatively low-speed data and can provide only data that is available through the Class-2 bus, so the VBI card can't entirely replace the other GPDAS signal-conditioning cards. However, the VBI card does supplement the other cards and makes the GPDAS more versatile and more economical to use.

References

[1] SAE Surface Vehicle Standard J1850, "Class B Data Communications Network Interface," revised May, 1994, SAE International, Warrendale, PA.

[2] David C. Pheanis and Lon D. Gowen, "General-Purpose Data-Acquisition System," ISMM International Conference on Microcomputer Applications, December 14–16, 1989, *Proceedings*, pp. 71-74.