Signal Control Program Environment Development

Interface Design Document

Version 1.8

06-FH1 Phase II

Advanced Technologies, Incorporated
5703 Red Bug Lake Road, Suite 320
Winter Springs, Florida 32708

Contact Number DTRT57-09-C-10005

Table of Contents

Revision	Date	Description
1.0	Sept 17, 2009	Initial Version
1.1	Nov 21, 2009	Modified Version for More Data between Pri-Sec
1.2	Feb 22, 2010	Added GUI / Primary Interface
1.3	Jun 7, 2010	Added Interface to the TEXAS Model Traffic Simulator
1.4	July 12, 2010	Add ability to select different intersection types.
1.5	April, 2011	Add interface for actuated control.
1.6	June, 2011	Add better debugging control.
1.7	Nov, 2011	Add Pedestrian Interface and Output Types
1.8	Jul 22, 2012	Add Data Recording Interface (Pages 29-41)

Table 1: Revision History

Introduction

This document contains the requirements and definitions for the interface between the Ada95 Primary Signal Intersection Control Program and the C++ Secondary Intersection Control program.

Primary / Secondary Command Names and Values

Name	Value	Definition	Direction
Heartbeat	0	Secondary is Alive	Secondary to Primary
Send Results	1	Request Secondary State	Primary to Secondary
Secondary State	2	Secondary Intersection State	Secondary to Primary
Preempt	3	Stop and Start New State	Primary to Secondary
Min Green Time	4	Set a Minimum Green Time	Primary to Secondary
Max Green Original Value	5	Set a Maximum Green Time initial value	Primary to Secondary
True Max Time	6	Set the True Max Green Value	Primary to Secondary
Extension Time	7	Set the initial Extension Time	Primary to Secondary
Consecutive Fails	8	Set the Consecutive Failure Constant Number	Primary to Secondary
Change Clear	9	Set a Clear Value	Primary to Secondary
Change Split	10	Modify a Split Time	Primary to Secondary
Change Mode	11	Switch Processing Mode	Primary to Secondary
Sending Time	12	Current Time	Primary to Secondary
Adjustment	13	Set the Max Green Adjustment Value	Primary to Secondary
Actuated Trigger	14	Extension Green Processing	Primary to Secondary
Actuated Mode	15	Set Actuated Processing Mode	Primary to Secondary
Gap Times	16	Actuated Mode - Set Gap Times)	Primary to Secondary
Time Before Reduction	17	Actuated Mode - Set Time Before Reduction)	Primary to Secondary
Time To Reduce	18	Actuated Mode - Set Time To Reduce)	Primary to Secondary
Min Gap Times	19	Actuated - Set Minimum Gaps	Primary to Secondary
Sync Message	20	Start of New Green Phase	Primary to Secondary
Debug Level	21	Logging Level (0 off)	Primary to Secondary

Name	Value	Definition	Direction
Phase Pedestrian Walk Time	22	Set Pedestrian Walk Time	Primary to Secondary
Phase Pedestrian Clearance Time	23	Set the Pedestrian Clearance “Flashing Don't Walk” Time	Primary to Secondary
Phase Pedestrian Omit	24	Command to not run pedestrian intervals on this phase	Primary to Secondary
Pedestrian Detector Phase Assignment	25	Assign a Pedestrian Detector to a Phase	Primary to Secondary
Pedestrian Detector Call	26	A Call on a Pedestrian Detector	Primary to Secondary

Table 2: Commands

Data sent from the Primary to the Secondary

Sending Time

Description: Time is sent to the secondary from the primary at the beginning of the main processing loop. It is used to keep the secondary software running in lock-step with the primary software. Seconds are seconds from midnight UTC of January 1, 1970 (epoch).

Size: 72 bits (command + data)

Structure:

Bits	Description	Possible Value(s)	Data Type
0 .. 7	Command	12	8 bit unsigned
8 .. 43	Seconds	0 .. Max Unsigned	32 bit unsigned
44 .. 71	Microseconds	0 .. Max Unsigned	32 bit unsigned

Table 3: Sending Time Structure

Frequency of Message: Sent once every 100 milliseconds.

Send Results

Description: A command sent from the primary program to the secondary program that instructs the secondary program to send its intersection state values back to the primary. This command is sent when the primary finishes its main processing loop.

Size: 8 bits (command)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	1	8 bit unsigned

Table 4: Send Results Structure

Frequency of Message: Sent once every 100 milliseconds.

Change Clear Time

Description: A command sent from the primary program to the secondary program that instructs the secondary program to modify a Yellow or Red clear time. This command is sent when the primary receives a command to change a clear time.

Size: 32 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	9	8 bit unsigned
8 .. 15	Clear_Time_To_Change	0 – Red 1 - Yellow	8 bit unsigned
16 .. 31	New_Time	0 - 200	16 bit unsigned

Table 5: Change Clear Time Structure

Frequency of Message: Sent on Event.

Change Split

Description: A command sent from the primary program to the secondary program that instructs the secondary program to modify a split time. This command is sent when the primary receives a command to change a split time.

Size: 32 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	10	8 bit unsigned
8 .. 15	Split_Time_To_Change	0 - 15	8 bit unsigned
16 .. 31	New_Time	0 - 200	16 bit unsigned

Table 6: Change Split Time Structure

Frequency of Message: Sent on Event.

Change Mode

Description: A command sent from the primary program to the secondary program that instructs the secondary program to change mode. This command is sent when the primary receives a command to change its processing type (i.e., pretimed to CICAS).

Size: 32 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	11	8 bit unsigned
8 .. 31	New_Mode	1000 – Pretimed 1001 – Actuated 1002 – CICAS 1003 - Low Level 1004 - Adaptive	24 bit unsigned

Table 7: Change Mode Structure

Frequency of Message: Sent on Event.

Preempt

Description: A command sent from the primary program to the secondary program that instructs the secondary program to stop processing and change state. There are several variants. If the immediate indicator is set, the phases immediately transition to the new phases set in the message. If the delay indicator is set, the phases wait for the amount of time specified before changing to the new phase. If the when_done indicator is set, the current phases should complete before changing to the new specified phases.

Size: 64 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	3	8 bit unsigned
8 .. 15	Indicator	0001 – Immediate 0002 – Delayed 0003 – When Phase Done	8 bit unsigned
16 .. 47	Delay Time	0.0 – 300.0 seconds	float
48 .. 55	New_Phase_0	1 - 16	8 bit unsigned
56 .. 63	New_Phase_1	1 - 16	8 bit unsigned

Table 8: Preempt Command Structure

Frequency of Message: Sent on Event.

Min Green Time

Description: A command sent from the primary program to the secondary program that instructs the secondary program to modify a minimum green time. This command is only used in actuated mode. It is sent when the primary changes a minimum green time.

Size: 32 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	4	8 bit unsigned
8 .. 15	Min_Green_Time_To_ Change	0 - 15	8 bit unsigned
16 .. 31	New_Time	0 - 200	16 bit unsigned

Table 9: Set Minimum Green Time Structure

Frequency of Message: Sent on Event.

Max Green Original Value

Description: A command sent from the primary program to the secondary program that instructs the secondary program to modify a maximum green time. This command is only used in actuated mode. It is sent when the primary changes a maximum green time.

Size: 32 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	5	8 bit unsigned
8 .. 15	Max_Green_Value_To_ Change	0 - 15	8 bit unsigned
16 .. 31	New_Value	0 - 200	16 bit unsigned

Table 10: Max Green Original Value Structure

Frequency of Message: Sent on Event.

True Max Time

Description: A command sent from the primary program to the secondary program that instructs the secondary program to modify a the true max green time. This command is only used in actuated mode. It is sent when the primary changes a true max green time. The "true max" green time is actually the greatest allowed value for the maximum green time. This maximum green time gets adjusted by an "adjustment" value up to the true max time.

Size: 32 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	6	8 bit unsigned
8 ..15	True Max Time to Change	0 - 15	8 bit unsigned
16 .. 31	New_Value	0 - 200	16 bit unsigned

Table 11: True Max Time Structure

Frequency of Message: Sent on Event.

Extension Time

Description: A command sent from the primary program to the secondary program that instructs the secondary program to modify an extension time. This command is only used in actuated mode. It is sent when the primary changes an extension time. The extension time is used to extend the minimum green value.

Size: 32 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	7	8 bit unsigned
8 .. 15	Extension Time to Change	0 - 15	8 bit unsigned
16 .. 31	New_Value	0 - 200	16 bit unsigned

Table 12: Extension Time Structure

Frequency of Message: Sent on Event.

Consecutive Failure Constant

Description: A command sent from the primary program to the secondary program that instructs the secondary program to modify the consecutive failure's constant. This command is only used in actuated mode. It is sent when the primary changes the consecutive failures constant.

Size: 32 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	8	8 bit unsigned
8.. 15	Number of Failures before Action	0 - 15	8 bit unsigned
16 .. 31	Spare	N/A	16 bit unsigned

Table 13: Consecutive Failure Constant Structure

Frequency of Message: Sent on Event.

Adjustment

Description: A command sent from the primary program to the secondary program that instructs the secondary program to modify the minimum green adjustment constant. This command is only used in actuated mode. It is sent when the primary changes the minimum green adjustment constant.

Size: 32 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	13	8 bit unsigned
8 .. 15	Spare	N/A	16 bit unsigned
16 .. 31	Spare	N/A	16 bit unsigned

Table 14: Adjustment Structure

Frequency of Message: Sent on Event.

Actuated Trigger

Description: A command sent from the primary program to the secondary program that instructs the secondary program to perform extension processing for minimum green.

Size: 32 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	14	8 bit unsigned
8 .. 15	Actuated Trigger_Element	0 - 15	8 bit unsigned
16 .. 31	New_Value	0 The channel has no call – and there has been no change in this status since this frame was last transmitted (no call – no change). 1 The channel has a call - and there has been no change in this status since this frame was last transmitted (constant call - no change). 2 The channel has no call - and there has been a change in this status since this frame was last transmitted (call has gone away). 3 The channel has a call - and there has been a change in this status since this frame was last transmitted (new call).	16 bit unsigned

Table 15: Actuated Trigger structure

Frequency of Message: Sent on Event.

Actuated Mode

Description: A command sent from the primary program to the secondary program that instructs the secondary program to perform actuated processing based on the mode sent.

Size: 32 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	15	8 bit unsigned
8 .. 15	Phase	0 - 15	8 bit unsigned
16 .. 31	New_Value	Presence (0) Recall Min (1) Recall Max (2), Max Out (3), Gap Out (4) , Pedestrian Recall(5)	16 bit unsigned

Table 16: Recall Mode structure

Frequency of Message: Sent on Event.

Gap Time

Description: A command sent from the primary program to the secondary program that instructs the secondary program to set a gap time

Size: 64 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	16	8 bit unsigned
8 .. 15	Phase	0 - 15	8 bit unsigned
16 .. 31	Spare	N/A	16 bit unsigned
32 .. 63	Gap Time	2.0 – 5.0	Float

Table 17: Gap Time structure

Frequency of Message: Sent on Event.

Time Before Reduction

Description: A command sent from the primary program to the secondary program that instructs

the secondary program to set a reduction wait time

Size: 64 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	17	8 bit unsigned
8 .. 15	Phase	0 - 15	8 bit unsigned
16 .. 31	Spare	N/A	16 bit unsigned
32 .. 63	Time Before Reduction	0.0 – 200.0	Float

Table 18: Time Before Reduction structure

Frequency of Message: Sent on Event.

Time To Reduce

Description: A command sent from the primary program to the secondary program that instructs the secondary program to set a reduction time

Size: 64 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	18	8 bit unsigned
8 .. 15	Phase	0 - 15	8 bit unsigned
16 .. 31	Spare	N/A	16 bit unsigned
32 .. 63	Time To Reduce	0.0 – 200.0	Float

Table 19: Time To Reduce structure

Frequency of Message: Sent on Event.

Minimum Gap Time

Description: A command sent from the primary program to the secondary program that instructs

the secondary program to set a min gap time

Size: 64 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	19	8 bit unsigned
8 .. 15	Phase	0 - 15	8 bit unsigned
16 .. 31	Spare	N/A	16 bit unsigned
32 .. 63	Minimum Gap Time	0.0 – 200.0	Float

Table 20: Minimum Gap Time structure

Frequency of Message: Sent on Event.

Send Sync

Description: A command sent from the primary program to the secondary program that tells the secondary to sync frame processing with the primary.

Size: 32 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	20	8 bit unsigned
8 .. 15	Ring	1 .. 2	8 bit unsigned

Table 21: Send Sync Structure

Frequency of Message: Sent on Event.

Debug Level

Description: A command sent from the primary program to the secondary program that tells the secondary at what debug level to run at (0 – none). The higher the level, the more information that is logged.

Size: 32 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	21	8 bit unsigned
8 .. 15	Debug Level	0 .. 3	8 bit unsigned

Table 22: Debug Level Structure

Frequency of Message: Sent on Event

Phase Pedestrian Walk Time

Description: A command sent from the primary program to the secondary program that instructs the secondary program to set the pedestrian walk time

Size: 64 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	22	8 bit unsigned
8 .. 15	Phase	0 - 15	8 bit unsigned
16 .. 31	Pedestrian Walk Time	0-255 Seconds	16 bit unsigned

Table 23: Pedestrian Walk Time structure

Frequency of Message: Sent on Event.

Phase Pedestrian Clearance Time

Description: A command sent from the primary program to the secondary program that instructs the secondary program to set the pedestrian clearance time

Size: 64 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	23	8 bit unsigned
8 .. 15	Phase	0 - 15	8 bit unsigned
16 .. 31	Pedestrian Clearance Time	0-255 Seconds	16 bit unsigned

Table 24: Pedestrian Clearance Time structure

Frequency of Message: Sent on Event.

Phase Pedestrian Omit

Description: A command sent from the primary program to the secondary program that instructs the secondary program to set the pedestrian omit state.

Size: 64 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	24	8 bit unsigned
8 .. 15	Phase	0 - 15	8 bit unsigned
16 .. 31	Pedestrian Omit	1=Pedestrian Omit, 0=Pedestrian Allow	16 bit unsigned

Table 25: Pedestrian Walk Time structure

Frequency of Message: Sent on Event.

Pedestrian Detector Phase Assignment

Description: A command sent from the primary program to the secondary program that instructs the secondary program to set the pedestrian walk time

Size: 64 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	25	8 bit unsigned
8 .. 15	Pedestrian Detector	0 - 8	8 bit unsigned
16 .. 31	Assigned Phase	0-8	16 bit unsigned

Table 26: Pedestrian Detector Phase Assignment structure

Pedestrian Detector Call

Description: A command sent from the primary program to the secondary program that instructs the secondary program to change the pedestrian call state.

Size: 64 bits (command + data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	26	8 bit unsigned
8 .. 15	Pedestrian Detector	0 - 8	8 bit unsigned
16 .. 31	Pedestrian Call	1=Pedestrian Call, 0=Pedestrian Clear	16 bit unsigned

Table 27: Pedestrian Detector Call structure

Frequency of Message: Sent on Event.

Data sent from the Secondary to the Primary

Heartbeat

Description: An alternating value (0,1) sent to the primary from the secondary to let the primary know the secondary is still executing. When the primary software receives the secondary heartbeat, it makes a call to an internal watchdog timer. If this timer is not called at least every second, a problem is declared.

Size: 16 bits (command+data)

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	Command	0	8 bit unsigned
8 .. 15	Data	0,1	8 bit unsigned

Table 28: Heartbeat Structure

Frequency of Message: Sent once every 900 milliseconds.

Secondary State

Description: Sent from the primary program to the secondary program when the primary requests the secondary's state data. The primary program then uses this data to insure both programs are computing the same values.

Size: 8 bits (command)
1280 bits (data)
Total – 1288 bits.

Command Structure:

Word Number	Size in Bytes	Size in Bits	Field Name	Possible Value(s)	Data Type
0	1	8	Command	2	unsigned

Data Structure:

Word Number	Size in Bytes	Field Name	Possible Value(s)	Data Type
0	64	Phase	1 .. 8	8 by 8 matrix of unsigned
16	64	Splits	0 – 300 (in seconds)	16 element array of float
32	64	New_Splits	0 – 300 (in seconds)	16 element array of float
48	4	Red Clear	0 – 300 (in seconds)	float
49	4	New Red Clear	0 – 300 (in seconds)	float
50	4	Yellow Change	0 – 300 (in seconds)	float
51	4	New Yellow Change	0 – 300 (in seconds)	float
52	4	Speed	0 – 50 (in seconds?)	float
53	2	Current Phase	1 through 8 (each element)	2 element array of Unsigned Byte
53	2	Control Mode	1002 - CICAS 1000 - Pretimed 1004 - Adaptive 1003 - Low	Short integer
54	2	New Control Mode	1002 - CICAS 1000 - Pretimed 1004 - Adaptive 1003 - Low	Short Integer
54	2	Current Color	Red (0), Yellow (1), Green (2)	2 element array of Unsigned Byte
55	1	Status	Initialized (10)	Unsigned Byte
55	3	Spare	N/A	3 Unsigned Bytes

Table 29: Secondary State Structure

Frequency of Message: Sent on command by primary to secondary

Data sent from the Primary to the GUI

Display Data

Description: Data displayed on the test GUI. This data is sent by the primary to the GUI over a TCP/IP channel. The data is sent once the main processing loop has finished a complete iteration.

Size: 1664 bytes
Structure:

Byte	Field Name	Possible Value(s)	Data Type
1	Display_Pri_Sec_Validation	0 (False), 1 (True)	8 bit Boolean
2	Display_Secondary_Status	0 (False), 1 (True)	8 bit Boolean
3	Display_Status	0 - 255	Byte
4	Display_Detector_Value	0 – 64 (0 = do nothing)	Byte
5	Display_Split_Counter	0 .. Max Float	Float
9	Display_Control_Mode	Pretimed (1000), Actuated (1001), CICAS (1002) Low_Level (1003) Adaptive (1004)	Short_Integer
11	Display_New_Control_ Mode	Pretimed (1000) Actuated (1001) CICAS (1002) Low_Level (1003) Adaptive (1004)	Short_Integer
13	Display_Rc	0 .. Max Float	Float
17	Display_New_Rc	0 .. Max Float	Float
21	Display_Yc	0 .. Max Float	Float
25	Display_New_Yc	0 .. Max Float	Float
29	Display_Current_Phase	Each byte ranges 0 - 7	2 Element Array of Byte
31	Display_Current_Color	0,1,2 (Red, Yellow, Green)	2 Element Array of Byte
33	Display_Speed	0 .. Max Float	Float
37	Display_Splits Display Transition Times*	0 .. Max Float	16 Element Array of Float
101	Display_New_Splits Display_Current_Max_ Green_Times*	0 .. Max Float	16 Element Array of Float

Byte	Field Name	Possible Value(s)	Data Type
165	Pedestrian Crossing On	1(Pedestrian Phase) 0 (Normal Phase)	Byte

Table 30: GUI Display Data Structure

Frequency of Message: Sent once every major processing iteration.

Notes:

* In actuated mode, the current transition times are displayed instead of split times. Transition times are equal to maximum green + yellow clear + red clear. In addition, the current value of max green is displayed instead of New Splits.

Data sent from the GUI to the Primary

GUI to Primary

Description: Commands and Data sent from the User GUI to the Primary Ada95 SCOPE program.

Size: 697 Bytes

Structure:

Byte	Field Name	Possible Value(s)	Data Type
1	Simulate_Secondary	0 (False), 1 (True)	8 bit Boolean
2	Forever	0 (False), 1 (True)	8 bit Boolean
3	Stop	0 (False), 1 (True)	8 bit Boolean
4	Output_Interface_Data	0 (False), 1 (True)	8 bit Boolean
5	Time_To_Stop	0 .. Max_Float	32 bit Float
9	Debug_Level	0 .. 3 (None,Low,Med,High)	16 bit Short Integer
11	Traffic Simulation	0 (False), 1 (True)	8 bit Boolean
12	Start	0 (False), 1 (True)	8 bit Boolean
13	Intersection Type	0 (4 Leg), 1 (3 Leg), 2 (Diamond)	Byte
14	Output Type	0 (None), 1 (NEMA TS2), 2 (ITS)	Byte
15	Trace On	1 (On), 0 (Off)	Byte
16	Spare 3	N/A	Byte
17	New_Rc	0 .. Max Float	Float
21	New_Yc	0 .. Max Float	Float
25~~6~~	Display Secondary	01(Display Secondary Data) 0(Off)	Byte
26	First Miscompare Flash~~t~~	1(Flash on Miscompare) 0 (Off)	Byte
27	Spare	N/A	Byte
28	Spare	N/A	Byte
29	New_Splits	Each Element 0 .. Max Float	16 Element Array of Float
93	New_Current_Color	0,1,2 (Red, Yellow, Green)	Byte
94	New_Control_Change	0 .. 255	Byte

Byte	Field Name	Possible Value(s)	Data Type
95	New_Mode	Pretimed (1000) Actuated (1001) CICAS (1002) Low_Level (1003) Adaptive (1004)	Short_Integer
97	New_Min_Green_ Times	Each Element 0 .. Max Float	16 Element Array of Float
161	New_Max_Green_ Original_Times	Each Element 0 .. Max Float	16 Element Array of Float
225	New_True_Max_Green _Times	Each Element 0 .. Max Float	16 Element Array of Float
289	New_Default_Extension Times	Each Element 0 .. Max Float	16 Element Array of Float
353	New_Consecutive Failures_Allowed	1 .. 10	Integer
357	New_Extension_Time Increment	1.0 .. 100.0	Float
361	New_Actuated_Mode	Presence (0) Recall Min (1) Recall Max (2), Max Out (3), Gap Out (4) , Pedestrian Recall (5)	16 Element Array of Byte
377	New_Actuators	0 The channel has no call – and there has been no change in this status since this frame was last transmitted (no call – no change). 1 The channel has a call - and there has been no change in this status since this frame was last transmitted (constant call - no change). 2 The channel has no call - and there has been a change in this status since this frame was last transmitted (call has gone away). 3 The channel has a call - and there has been a change in this status since this frame was last transmitted (new call).	64 Element Array of Byte

Byte	Field Name	Possible Value(s)	Data Type
441	Gap Time	2.0 – 5.0 Seconds	16 element array of Float
505	Time Before Reduction	0.0 – 200.0 Seconds	16 element array of Float
569	Time to Reduce	0.0 – 200.0 Seconds	16 element array of Float
633	Min Gap Time	0.0 – 200.0 Seconds	16 element array of Float
697	Phase Pedestrian Walk Time	0-255 Seconds	16 element array of Unsigned Byte
713	Phase Pedestrian Clearance Time	0-255 Seconds	16 element array of Unsigned Byte
729	Phase Pedestrian Omit	1=Pedestrian Omit, 0=Pedestrian Allow	16 element array of Unsigned Byte
745	Pedestrian Detector Phase Assignment	1-8 Phase	8 element array of Unsigned Byte
753	Pedestrian Detector Call	1=Pedestrian Call, 0=Pedestrian Clear	8 element array of Unsigned Byte

Table 31: GUI To Primary Data Structure

Frequency of Message: Sent by GUI when user hits START button.
Sent by GUI every 100ms only if GUI detects new input from user.

Texas Model Interface

The Texas Model for Intersection Traffic Control has been integrated into SCOPE. The Texas Model is a single intersection simulation model developed at the University of Texas under the lead of Dr. Thomas Rioux.

Data sent from the Texas Model to the Primary

From Texas Model To Scope

Description: Commands and Data sent from the Texas Model to the Primary Ada95 SCOPE program.

Size: 72 bits

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	CID II ID	0 - 256	Byte
8	Phase 1 Detector	0 (False), 1 (True)	1 bit Boolean
9	Phase 2 Detector	0 (False), 1 (True)	1 bit Boolean
10	Phase 3 Detector	0 (False), 1 (True)	1 bit Boolean
11	Phase 4 Detector	0 (False), 1 (True)	1 bit Boolean
12	Phase 5 Detector	0 (False), 1 (True)	1 bit Boolean
13	Phase 6 Detector	0 (False), 1 (True)	1 bit Boolean
14	Phase 7 Detector	0 (False), 1 (True)	1 bit Boolean
15	Phase 8 Detector	0 (False), 1 (True)	1 bit Boolean
16 .. 23	SPARE	0	Byte
24	SPARE	0	1 bit Boolean
25	SPARE	0	1 bit Boolean
26	SPARE	0	1 bit Boolean
27	SPARE	0	1 bit Boolean
28	Pedestrian 2 Detector	0 (False), 1 (True)	1 bit Boolean
29	Pedestrian 4 Detector	0 (False), 1 (True)	1 bit Boolean
30	Pedestrian 6 Detector	0 (False), 1 (True)	1 bit Boolean
31	Pedestrian 8 Detector	0 (False), 1 (True)	1 bit Boolean

Bits	Field Name	Possible Value(s)	Data Type
32 .. 39	SPARE	0	Byte
40 .. 47	SPARE	0	Byte
48 .. 55	SPARE	0	Byte
56 .. 63	SPARE	0	Byte
64	SPARE	0	1 bit Boolean
65	SPARE	0	1 bit Boolean
66	SPARE	0	1 bit Boolean
67	SPARE	0	1 bit Boolean
68	Restart	0 (False), 1 (True)	1 bit Boolean
69	SPARE	0	1 bit Boolean
70	IO Mode 1	0 (False), 1 (True)	1 bit Boolean
71	IO Mode 2	0 (False), 1 (True)	1 bit Boolean

Table 32: Texas Model to SCOPE Structure

Frequency of Message: Sent based on Texas Model input parameters.

Data sent from the Primary to the Texas Model

From Scope to the Texas Model

Description: Commands and Data sent from the Primary Ada95 SCOPE program to the Texas Model.

Size: 64 bits

Structure:

Bits	Field Name	Possible Value(s)	Data Type
0 .. 7	CID II ID	0 - 256	Byte
8	Phase Data	See CID II Format Below	3 Bytes
9	Overlap Data	Future Implementation	2 Bytes
9	Pedestrian Data	Future Implementation	2 Bytes

Table 33: SCOPE To Texas Model Structure

Frequency of Message: Sent every 100 ms.

Data sent from the Primary to the Data Recording Process

Set Data Message Size

Description: Data sent from the Primary Ada95 SCOPE program to the Data Recording Process to tell it how large each following data message will be in bytes.

Size: 8 bytes

Structure:

Byte	Field Name	Possible Value(s)	Data Type
1	**Set Data Message Size**	**0**	**byte**
2	**Spare1**	**N/A**	**byte**
3	**Spare2**	**N/A**	**byte**
4	**Spare3**	**N/A**	**byte**
5	**Message Size in Bytes**	**1 - max unsigned 32 bit integer**	**unsigned 32 bit integer**

Table 34: SCOPE Set Data Message Size

SCOPE Data to Save

Description: Data sent from the Primary Ada95 SCOPE program to the Data Recording Process.

Size: 240 bytes

Structure:

Byte	Field Name	Possible Value(s)	Data Type
0	SCOPE Data to Save Cmd	1	Unsigned Byte
1	Spare 1	N/A	N/A
2	Spare 2	N/A	N/A
3	Spare 3	N/A	N/A
4	**Message ID**	1 - max unsigned integer	32 bit unsigned integer
8	Timestamp	0 - max time	Timeval
16	Actuators 1-4	4 times (00,01,10,11) where 00 => no call, 01 => new call, 10 => has call, 11 => had call	Unsigned Byte

Byte	Field Name	Possible Value(s)	Data Type
17	Actuators 5-8	4 times (00,01,10,11) where 00 => no call, 01 => new call, 10 => has call, 11 => had call	Unsigned Byte
18	Actuators 9-12	4 times (00,01,10,11) where 00 => no call, 01 => new call, 10 => has call, 11 => had call	Unsigned Byte
19	Actuators 13-16	4 times (00,01,10,11) where 00 => no call, 01 => new call, 10 => has call, 11 => had call	Unsigned Byte
20	Actuators 17-20	4 times (00,01,10,11) where 00 => no call, 01 => new call, 10 => has call, 11 => had call	Unsigned Byte
21	Actuators 21-24	4 times (00,01,10,11) where 00 => no call, 01 => new call, 10 => has call, 11 => had call	Unsigned Byte
22	Actuators 25-28	4 times (00,01,10,11) where 00 => no call, 01 => new call, 10 => has call, 11 => had call	Unsigned Byte
23	Actuators 29-32	4 times (00,01,10,11) where 00 => no call, 01 => new call, 10 => has call, 11 => had call	Unsigned Byte
24	Actuators 33-36	4 times (00,01,10,11) where 00 => no call, 01 => new call, 10 => has call, 11 => had call	Unsigned Byte
25	Actuators 37-40	4 times (00,01,10,11) where 00 => no call, 01 => new call, 10 => has call, 11 => had call	Unsigned Byte
26	Actuators 41-44	4 times (00,01,10,11) where 00 => no call, 01 => new call, 10 => has call, 11 => had call	Unsigned Byte
27	Actuators 45-48	4 times (00,01,10,11) where 00 => no call, 01 => new call, 10 => has call, 11 => had call	Unsigned Byte
28	Actuators 49-52	4 times (00,01,10,11) where 00 => no call, 01 => new call, 10 => has call, 11 => had call	Unsigned Byte
29	Actuators 53-56	4 times (00,01,10,11) where 00 => no call, 01 => new call, 10 => has call, 11 => had call	Unsigned Byte

Byte	Field Name	Possible Value(s)	Data Type
30	Actuators 57-60	4 times (00,01,10,11) where 00 => no call, 01 => new call, 10 => has call, 11 => had call	Unsigned Byte
31	Actuators 61-64	4 times (00,01,10,11) where 00 => no call, 01 => new call, 10 => has call, 11 => had call	Unsigned Byte
32	Pedestrian Calls 1 - 8	8 times (0 or 1) 0 => no call, 1 => call	Unsigned Byte
33	Control Mode	0 => Pretimed, 1 => Actuated	Unsigned Byte
34	Phase On Queue	8 times (0 or 1) queued or not	8 bit Boolean array
35	Spare 5	N/A	Unsigned Byte
36	Current Actuated Mode for Phase 1	(xxxxx000) => Presence (xxxxx001) => Min Recall (xxxxx010) => Max Recall (xxxxx011) => Max Out (xxxxx100) => Gap Out	Unsigned Byte
37	Current Actuated Mode for Phase 2	(xxxxx000) => Presence (xxxxx001) => Min Recall (xxxxx010) => Max Recall (xxxxx011) => Max Out (xxxxx100) => Gap Out	Unsigned Byte
38	Current Actuated Mode for Phase 3	(xxxxx000) => Presence (xxxxx001) => Min Recall (xxxxx010) => Max Recall (xxxxx011) => Max Out (xxxxx100) => Gap Out	Unsigned Byte
39	Current Actuated Mode for Phase 4	(xxxxx000) => Presence (xxxxx001) => Min Recall (xxxxx010) => Max Recall (xxxxx011) => Max Out (xxxxx100) => Gap Out	Unsigned Byte
40	Current Actuated Mode for Phase 5	(xxxxx000) => Presence (xxxxx001) => Min Recall (xxxxx010) => Max Recall (xxxxx011) => Max Out (xxxxx100) => Gap Out	Unsigned Byte
41	Current Actuated Mode for Phase 6	(xxxxx000) => Presence (xxxxx001) => Min Recall (xxxxx010) => Max Recall	Unsigned Byte

Byte	Field Name	Possible Value(s)	Data Type
		(xxxxx011) => Max Out (xxxxx100) => Gap Out	
42	Current Actuated Mode for Phase 7	(xxxxx000) => Presence (xxxxx001) => Min Recall (xxxxx010) => Max Recall (xxxxx011) => Max Out (xxxxx100) => Gap Out	Unsigned Byte
43	Current Actuated Mode for Phase 8	(xxxxx000) => Presence (xxxxx001) => Min Recall (xxxxx010) => Max Recall (xxxxx011) => Max Out (xxxxx100) => Gap Out	Unsigned Byte
44	Current Min Green Time Phase 1	0.0 to 200.0	Float
48	Current Min Green Time Phase 2	0.0 to 200.0	Float
52	Current Min Green Time Phase 3	0.0 to 200.0	Float
56	Current Min Green Time Phase 4	0.0 to 200.0	Float
60	Current Min Green Time Phase 5	0.0 to 200.0	Float
64	Current Min Green Time Phase 6	0.0 to 200.0	Float
68	Current Min Green Time Phase 7	0.0 to 200.0	Float
72	Current Min Green Time Phase 8	0.0 to 200.0	Float
76	Current Max Green Time Phase 1	0.0 to 200.0	Float
80	Current Max Green Time Phase 2	0.0 to 200.0	Float
84	Current Max Green Time Phase 3	0.0 to 200.0	Float
88	Current Max Green Time Phase 4	0.0 to 200.0	Float
92	Current Max Green	0.0 to 200.0	Float

Byte	Field Name	Possible Value(s)	Data Type
	Time Phase 5		
96	Current Max Green Time Phase 6	0.0 to 200.0	Float
100	Current Max Green Time Phase 7	0.0 to 200.0	Float
104	Current Max Green Time Phase 8	0.0 to 200.0	Float
108	Current True Max Time Phase 1	0.0 to 200.0	Float
112	Current True Max Time Phase 2	0.0 to 200.0	Float
116	Current True Max Time Phase 3	0.0 to 200.0	Float
120	Current True Max Time Phase 4	0.0 to 200.0	Float
124	Current True Max Time Phase 5	0.0 to 200.0	Float
128	Current True Max Time Phase 6	0.0 to 200.0	Float
132	Current True Max Time Phase 7	0.0 to 200.0	Float
136	Current True Max Time Phase 8	0.0 to 200.0	Float
140	Cur Gap Time Phase 1	0.0 to 20.0	Float
144	Cur Gap Time Phase 2	0.0 to 20.0	Float
148	Cur Gap Time Phase 3	0.0 to 20.0	Float
152	Cur Gap Time Phase 4	0.0 to 20.0	Float
156	Cur Gap Time Phase 5	0.0 to 20.0	Float
160	Cur Gap Time Phase 6	0.0 to 20.0	Float
164	Cur Gap Time Phase 7	0.0 to 20.0	Float
168	Cur Gap Time Phase 8	0.0 to 20.0	Float
172	Cur Time To Reduce 1	0.0 to 20.0	Float
176	Cur Time To Reduce 2	0.0 to 20.0	Float
180	Cur Time To Reduce 3	0.0 to 20.0	Float

Byte	Field Name	Possible Value(s)	Data Type
184	Cur Time To Reduce 4	0.0 to 20.0	Float
188	Cur Time To Reduce 5	0.0 to 20.0	Float
192	Cur Time To Reduce 6	0.0 to 20.0	Float
196	Cur Time To Reduce 7	0.0 to 20.0	Float
200	Cur Time To Reduce 8	0.0 to 20.0	Float
204	Cur Min Gap Time Phase 1	0.0 to 20.0	Float
208	Cur Min Gap Time Phase 2	0.0 to 20.0	Float
212	Cur Min Gap Time Phase 3	0.0 to 20.0	Float
216	Cur Min Gap Time Phase 4	0.0 to 20.0	Float
220	Cur Min Gap Time Phase 5	0.0 to 20.0	Float
224	Cur Min Gap Time Phase 6	0.0 to 20.0	Float
228	Cur Min Gap Time Phase 7	0.0 to 20.0	Float
232	Cur Min Gap Time Phase 8	0.0 to 20.0	Float
236	Adjustment Time	0.0 to 20.0	Float

Table 35: SCOPE Data To Save

Get History Buffer Size

Description: Command Sent to Data Recording to get the **number of messages** saved in the history buffer.

Size: 8 bits

Structure:

Byte	Field Name	Possible Value(s)	Data Type
1	**Get History Buffer Size**	**2**	**byte**

Table 36: SCOPE Get History Size In Messages Saved

Frequency of Message: Sent upon event (Display History Buffer on GUI).

Get Entire History Buffer

Description: Command Sent to Data Recording to get the **entire saved history** data.

Size: 8 bits

Structure:

Byte	Field Name	Possible Value(s)	Data Type
1	Get Entire History Buffer	3	byte

Table 37: SCOPE Get History Size In Messages Saved

Get A Single History Buffer Message

Description: Command Sent to Data Recording to get the a single **saved history** message.

Size: 64 bits

Structure:

Byte	Field Name	Possible Value(s)	Data Type
1	Get History Buffer Message	4	byte
2	Spare1	N/A	byte
3	Spare2	N/A	byte
4	Spare3	N/A	byte
5	Message ID	1 - max unsigned 32 bit integer	unsigned 32 bit integer

Table 38: SCOPE Get A Single Saved Data Buffer Message

Stop Recording

Description: Data sent from the Primary Ada95 SCOPE program to the Data Recording Process to tell it the current recording session is over and to close the data file.

Size: 1 byte

Byte	Field Name	Possible Value(s)	Data Type
1	Stop Recording	7	byte

Data sent from the Data Recording Process to The Primary

Number of Messages Recorded

Description: Command Sent to Data Recording to return the number of messages recorded.

Size: 64 bits

Structure:

Byte	Field Name	Possible Value(s)	Data Type
1	**Number Messages Recorded**	**5**	**byte**
2	**Spare1**	**N/A**	**byte**
3	**Spare2**	**N/A**	**byte**
4	**Spare3**	**N/A**	**byte**
5	**Number**	**1 - max unsigned 32 bit integer**	**unsigned 32 bit integer**

Table 39: External Data Recording Number of Messages Stored.

Data Message

Description: Data sent from the data recording process to the SCOPE Primary when commanded. If the data recording process receives a get history buffer element command, it sends only one of these structures back. If it receives a get entire history buffer command from the Primary, then the data recording process sends the entire data recording history, **A MESSAGE AT A TIME.**

Size: 302 bytes

Structure:

Byte	Field Name	Possible Value(s)	Data Type
0	Data Message Cmd	6	Unsigned Byte
1	Spare 1	N/A	N/A
2	Spare 2	N/A	N/A
3	Spare 3	N/A	N/A
4	**Message ID**	1 - max unsigned integer	32 bit unsigned integer
8	Timestamp	0 - max time	Timeval

Byte	Field Name	Possible Value(s)	Data Type
16	Actuator 1 - 64	(0,1,2,3) where 0 => no call, 1 => new call, 2 => has call, 3 => had call	array 1-64 of Unsigned Bytes
80	Pedestrian Calls 1 - 8	(0 or 1) 0 => no call, 1 => call	array 1-8 of Unsigned Byte
88	Control Mode	0 => Pretimed, 1 => Actuated	Unsigned Byte
89	Phase 1-8 On Queue	(0 or 1) queued or not	array 1-8 of Unsigned Byte
97	Spare 5	N/A	Unsigned Byte
98	Current Actuated Mode for Phases 1 - 8	(xxxxx000) => Presence (xxxxx001) => Min Recall (xxxxx010) => Max Recall (xxxxx011) => Max Out (xxxxx100) => Gap Out	array 1-8 of Unsigned Byte
106	Current Min Green Time Phases 1 - 8	0.0 to 200.0	array 1-8 of Float
138	Current Max Green Time Phases 1 - 8	0.0 to 200.0	array 1-8 of Float
170	Current True Max Time Phases 1 - 8	0.0 to 200.0	array 1-8 of Float
202	Cur Gap Time Phases 1 - 8	0.0 to 20.0	array 1-8 of Float
234	Cur Time To Reduce Phases 1 - 8	0.0 to 20.0	array 1-8 of Float
266	Cur Min Gap Time Phase 1	0.0 to 20.0	array 1-8 of Float
298	Adjustment Time	0.0 to 20.0	Float

CID II Format

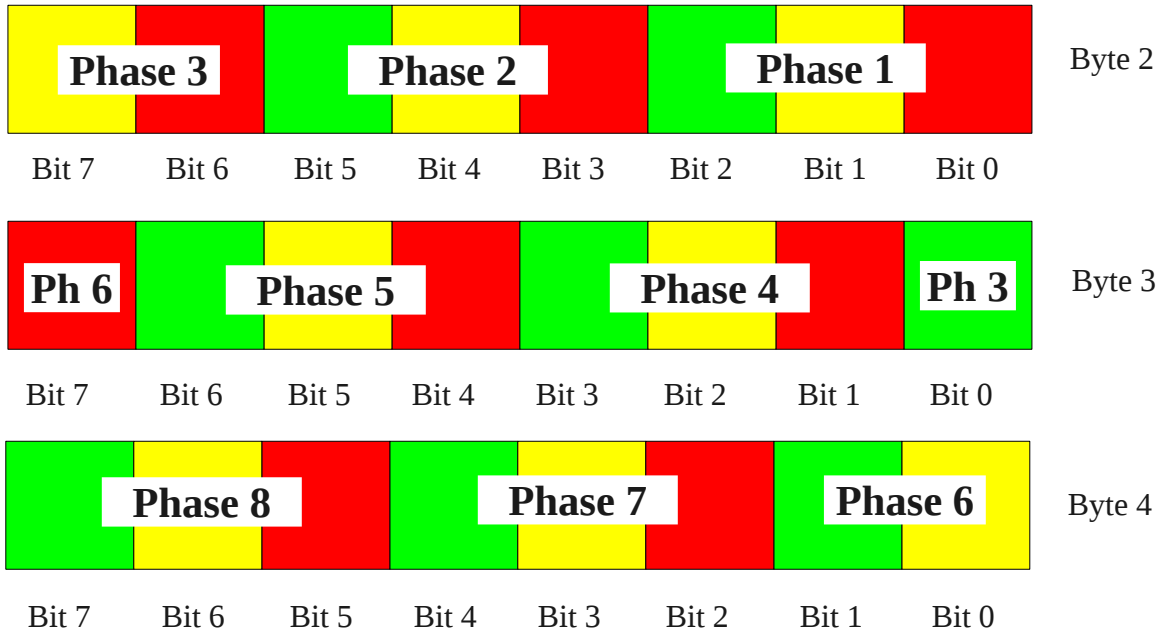

Table 40: CID Format for Texas Model Interface