IDB-0008-0D

ITS Data Bus

Phone Application Messages

January 18, 2000

Printed in U.S.A.

Table of Contents

1. Scope 5
2. References 6
2.1 Applicable Documents 6
2.1.1 IDB Forum Publications 6
2.1.2 SAE Publications 6
2.2 Related Publications 6
2.2.1 IDB Forum Publications 6
2.2.2 SAE Publications 6
2.2.3 TIA and Working Group T/WG 11 “Switching and Signalling” [sic] 6
2.2.4 ISO Publications 6
2.2.5 Other Publications 6
3. Definitions 7
3.1 ASN.1 7
3.2 AAS 7
3.3 Dial Digits 7
3.4 Telephone Functional Levels 7
3.4.1 Level 0 7
3.4.2 Level 1 7
3.4.3 Level 2 7
3.4.4 Level 3 7
3.5 PAS 8
3.6 Scratchpad Buffer 8
4. Device Application Capabilities 9
4.1 Telephone General Requirements 9
4.1.1 Telephone SAPID 9
4.1.2 Suppression of Unneeded Data 9
4.1.3 Smallest Required Binary Data 9
4.1.4 IDB Audio Arbitration 9
4.2 Level One Capabilities 9
4.2.1 Message Responses 9
4.2.2 Telephone Required Messages 9
4.2.2.1 Clear Scratchpad Buffer 11
4.2.2.2 Load Dial Digits to Scratchpad 11
4.2.2.3 Load Phone Number into Scratchpad 11
4.2.2.4 Dial Active Number 11
4.2.2.5 Mute the Microphones 11
4.2.2.6 UnMute the Microphones 11
4.2.2.7 Mute Audio Output 11
4.2.2.8 Un-Mute Audio Output 12
4.2.2.9 Adjust Ringer Volume 12
4.2.2.10 Request Keypresses / Stop Keypresses / Keypress Report 12
4.2.2.11 Keypress Command 12
4.2.3 PhoneStatus Structure Definition 12
4.2.4 KeyList Structure Definition 13
4.2.5 PhoneCapability Structure Definition 13
4.2.6 PhoneMode Structure Definition 14
4.2.7 NetworkCapability Structure Definition 14
4.2.8 NetworkMode Structure Definition 15
4.2.9 Telephone Keypress Codes 15
4.3 Level 2 Capabilities 16

4.3.1 Message Responses....16
4.3.2 Telephone Level 2 Capabilities General Requirements / Notes....16
4.3.2.1 Respect Locked / Unlocked State....16
4.3.2.2 Internal Display Access....16
4.3.2.3 Start With First....17
4.3.2.4 Maintain List Position....17
4.3.2.5 Return Last Phone Number List Entry Indication....17
4.3.3 Telephone Level 2 Messages....17
4.3.3.1 Signal Strength Report....18
4.3.3.2 Battery Charge Status Report....19
4.3.3.3 Telephone's Numbers....19
4.3.3.4 Phone Number Capacity....19
4.3.3.5 Clear Phone Number Information / Clear All Phone Number Information....19
4.3.3.6 Get First / Next / Previous Stored Phone Number Info....19
4.3.3.7 Store Phone Number Information....19
4.3.3.8 Phone Number Information Reply....19
4.3.4 PhoneStatus Structure Definition....20
4.4 Level 3 Capabilities....20
4.4.1 Message Responses....20
4.4.2 Telephone Level 3 Capabilities General Requirements / Notes....20
4.4.3 Telephone Level 3 Messages....20
4.4.3.1 Emergency Call with Number....21
4.4.3.2 Security Codes....21
5. Notes....22

Table of Tables

Table 1 — IDB Telephone Required Messages....10
Table 2 — IDB Telephone Keypress Codes....15
Table 3 — IDB Telephone Level 2 Messages....17
Table 4 — IDB Telephone Level 3 Messages....21

1. Scope

This IDB Forum document details Communication Messages, used by the Application Message Layer of the Intelligent Transportation Systems (ITS) Data Bus, which is generally intended for in-vehicle use.

The message set described by this document is divided between a minimal required set and one or more optional sets for devices of greater capability. It is intended that the required set contain those messages which all devices of this type would support. It is intended that the additional sets contain additional messages that may be useful to a device of this type. If a device supports the capability represented by one or more messages in these sets, this document describes the format of message that shall be used to provide that capability.

Messages in this document are constructed using the elements described by IDB-0002 / J2366-7LX and methods described by SAE J2366-7.

The ITS Data Bus (IDB) is a non-proprietary bus, designed to allow disparate consumer, vehicle, and commercial electronic components to communicate and share information across a standard, open data bus.

2. References

2.1 Applicable Documents

The following publications form a part of this specification to the extent specified herein. The latest issue of IDB Forum and of SAE publications shall apply.

2.1.1 IDB Forum Publications

Available from http://www.idbforum.org.

IDB-0002—ITS Data Bus—Application Message Layer Lexicon, IDB Forum
IDB-0004—ITS Data Bus—Other Application Messages, IDB Forum

2.1.2 SAE Publications

Available from SAE, 400 Commonwealth Drive, Warrendale, PA 15096-0001, http://www.sae.org.

J2366-7—ITS Data Bus—Application Message Layer
J2366-7LX—ITS Data Bus—Application Message Layer Lexicon

2.2 Related Publications

2.2.1 IDB Forum Publications

IDB-0009—ITS Data Bus—User Interface Application Messages, IDB Forum

2.2.2 SAE Publications

J2366-1—ITS Data Bus—Protocol Physical Layer
J2366-2—ITS Data Bus—Protocol Link Layer
J2366-4—ITS Data Bus—Protocol Thin Transport Layer

2.2.3 TIA and Working Group T/WG 11 “Switching and Signalling” [sic]

Available from Global Engineering Documents, http://global.ihs.com.

TIA/EIA IS-789A Electrical Specification for the Portable Phone to Vehicle Interface
CEPT-T/CS 34-01 (Innsbruck 1981), Edition of May 15, 1986—Arrangements of Push-Buttons and the Symbols for Their Designation

2.2.4 ISO Publications

8824-1 International Standard, “Information technology — Abstract Syntax Notation One (ASN.1): Specification of basic notation”
(also, related standards 8824-2, 8824-3, and 8824-4)
8825: 1996 - Information Technology — ASN.1 Encoding Rules
8825-1 International Standard, “Information Technology — ASN.1 Encoding Rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)
8825-2 - International Standard, “Information Technology — ASN.1 Encoding Rules: Specification of Packed Encoding Rules (PER)

2.2.5 Other Publications

Available from the Internet Mail Consortium, http://www.imc.org.

RFC 2425—A MIME Content-Type for Directory Information — http://www.imc.org/rfc2425
RFC 2426—vCard MIME Directory Profile — http://www.imc.org/rfc2426

3. Definitions

Additional definitions applicable to this document can be found in SAE J2366-7.

3.1 ASN.1

ASN.1 is a formal specification language for expressing bit patterns related to communication protocols. Some of the expressive elements of ASN.1 have been included in this document, in order to more clearly specify the document's intent. ASN.1 allows a number of encoding methods. The ASN.1 in this document shall be interpreted as expressing the specified elements in the most compact form possible. Thus, if an element is specified as an integer having a value that ranges from zero to seven [`INTEGER (0..7)`], it means that exactly three bits are allocated to that purpose. Reading down a structure definition, more significant bits are specified before less significant bits. More significant octets are transmitted before less significant octets, whether or not they are part of a single `INTEGER` or other type. ASN.1 notation is used in this document only for clarity as to the bits to be transmitted. There is no intent that an ASN.1 interpreter needs to be included within any device.

3.2 AAS

Automobile Adaptation System

3.3 Dial Digits

The standard set of dial digits is 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, *, and #. (see 2.2.3)

3.4 Telephone Functional Levels

The TIA working group has defined four "levels" of telephone functional capability. The IDB required minimum message list is comprised of a combination of Levels 0 and 1. The following definitions of the different levels is taken from the TIA definitions.

3.4.1 Level 0

This level is reserved only for legacy phones which have been introduced prior to Calendar Year (CY) 2001. Level 0 must support the functionality of a basic HF cellular kit plus the ability for the AAS to answer and terminate the call. Messages that are part of this level are required for IDB compliance.

3.4.2 Level 1

Lowest common denominator set of messages that released phones after CY 2001 will support. Level 1 includes the functionality of level 0 in addition to the ability to dial phone number. Messages that are part of this level are required for IDB compliance.

3.4.3 Level 2

More advanced messages that only selected phones(future) phones would support. Level 2 includes level 1 and the functionality of a smart HF cellular kit such as phone memory management (recall and store), display management, etc. All messages in Level 2 must be supported in order for a device to claim Level 2 compliance.

3.4.4 Level 3

An extensive set of advanced messages that legacy or future phones would support. Level 3 includes level 2 and access to highly secured phone functions such as lock code, security code, cellular phone

mobile number, phone settings control, etc. A device is considered Level 3 capable if it supports at least one Level 3 message.

3.5 PAS

Phone Adaptation System

3.6 Scratchpad Buffer

As used in this document, the “Scratchpad” is a device temporary memory location for such purposes as storing user phone number entry, storing message driven phone number entry, and storing outgoing touch tone “digits”.

4. Device Application Capabilities

This chapter details the capabilities that apply to the use of devices of this type over the IDB.

4.1 Telephone General Requirements

The following requirements apply all IDB telephones, regardless of capability level.

4.1.1 Telephone SAPID

The SAPID value for Telephone related messages shall be 256.

4.1.2 Suppression of Unneeded Data

Some messages are defined as allowing multiple data type / value pairs. In some cases, one or more of these pairs might not carry useful information. For example, the reporting of the value of the Call Timer allows the number of days a call has been in progress to be reported. Some calls will not have a duration that exceeds a full day. For such calls, the UDay type / value pair would have a value of zero. In such cases, the use of the UDay type / value pair is not required, as the zero value is assumed.

4.1.3 Smallest Required Binary Data

For those messages that specify the use of an unsigned binary data type using the string “UBinary_*”, the message shall be constructed using the smallest of the UBinary data types that is capable of holding the value to be supplied.

4.1.4 IDB Audio Arbitration

A telephone that uses the IDB Audio Bus shall use the IDB Audio Arbitration messaging protocol.

4.2 Level One Capabilities

IDB Compliant telephones shall, at a minimum, support and act appropriately on all messages included in Telephone Functional Levels 0 and 1. These messages are found in Table 1.

4.2.1 Message Responses

The IDB J2366-7 Application Message Layer protocol provides a mechanism for messages in response to Command or Request messages. These response messages incorporate the IDBResult code. Several values for the IDBResult have been defined in J2366-7LX.

IDB Compliant telephone shall support the generation and interpretation of response status messages that incorporate the IDBResult code.

The IRCrcError value does not apply to the messages defined in this document.

The IRDeviceNotPresent value does not apply to Level One Capabilities messages.

The IRNotSupported value does not apply to Level One Capabilities messages.

The IRSecurityError value does not apply to Level One Capabilities messages.

4.2.2 Telephone Required Messages

The following list of telephone messages is the set required for IDB Compliance. This list is based upon the TIA Level 0 and Level 1 message groups. The two levels are combined into a single table, because the Level 1 capability set is the long-term minimum set intended. For the short term, in which Level 0

capable phones can be IDB compliant, only the Phone Status Request, Phone Status OnChange, Phone Status, Answer Call, Terminate Call, Turn Phone On, and Turn Phone Off are Level 0 messages. These messages are noted in Table 1 by an “X” in the “Lvl 0” column.

Additional explanation and definition of these messages can be found in the sub-paragraphs that follow Table 1.

Table 1 — IDB Telephone Required Messages

IDB Telephone Required Messages

Lvl 0	Message	MapClass *UnitType*	Action *Amount*	Modifier *UnitType*	Object *Amount*	Feature *UnitType*
[Phone Information Group]						
X	Phone Status Request	Request	AReport		Telephone	DeviceStatus
X	Phone Status OnChange	Command	AReport	OnChange	Telephone	DeviceStatus
X	Phone Status	Reply/Datagram *UBinaryVariable*	AReport *PhoneStatus*			
	Phone Capability Request	Request	AReport			Capacity
	Phone Capability OnChange	Command	AReport	OnChange		Capacity
	Phone Capability	Reply/Datagram *UBinaryVariable*	ANone *PhoneCapability*			
	Phone Mode Request	Request	AReport			DeviceStatus
	Phone Mode OnChange	Command	AReport	OnChange		DeviceStatus
	Phone Mode	Reply/Datagram *UBinaryVariable*	ANone *PhoneMode*			
[Call Processing Group]						
X	Answer Call	Command	Answer			
X	Terminate Call	Command	CloseHangup			
	Clear Scratchpad Buffer	Command	Clear			PhoneNumber
	Load Dial Digits to Scratchpad	Command *UPhoneNumber*	Load *##########*	Next		PhoneNumber
	Load Phone Number into Scratchpad	Command *UPhoneNumber*	Load *##########*	OutOutput		PhoneNumber
	Dial Active Number	Command	CallDial			
[Audio Management Group]						
	Mute the Microphones	Command	Mute	On		Input
	Un-Mute the Microphones	Command	Mute	Off		Input
	Mute Audio Output	Command	Mute	On		Output
	Un-Mute Audio Output	Command	Mute	Off		Output
	Adjust Ringer Volume Up	Command	SetWrite	UpN	Chime	Volume
	Adjust Ringer Volume Dn	Command	SetWrite	DownS	Chime	Volume
[Network Information Group]						
	Network Capability Request	Request	AReport		Transmission	Capacity
	Network Capability OnChange	Command	AReport	OnChange	Transmission	Capacity
	Network Capability	Reply/Datagram *UBinaryVariable*	AReport *NetworkCapability*			
	Network Mode Request	Request	AReport		Transmission	DeviceStatus
	Network Mode OnChange	Command	AReport	OnChange	Transmission	DeviceStatus
	Network Mode	Reply/Datagram *UBinaryVariable*	AReport *NetworkMode*			
[User Interface Group]						
X	Turn Phone On	Command	OnEnableArmlock			

IDB Telephone Required Messages

Lvl 0	Message	MapClass *UnitType*	Action *Amount*	Modifier *UnitType*	Object *Amount*	Feature *UnitType*
X	Turn Phone Off	Command	OffDisableUnlock			
	Request Keypresses	Command	AReport	OnChange		Keypress
	Stop Keypresses	Command	Stop	OnChange		Keypress
	Keypress Report	Datagram *UBinaryVariable*	AReport *KeyList*			Keypress
	Keypress Command	Command *UBinaryVariable*	SetWrite *KeyList*			Keypress
	Hands-Free Mode	Command	SelectUse			External
	Hand-Set Mode	Command	SelectUse			Internal

4.2.2.1 Clear Scratchpad Buffer

This command shall purge the entire Scratchpad Buffer.

4.2.2.2 Load Dial Digits to Scratchpad

While no call is in progress, this message shall cause the specified digit data to be appended to the existing contents of the Scratchpad. During a call, this command shall cause the specified digit data to be appended to the existing contents of the Scratchpad and shall cause the DTMF tones that correspond to the Dial Digits loaded to be transmitted. The reply message to this command should contain the number of Dial Digits successfully loaded. If this is done, it shall use the UBinary_08 UnitType, immediately following the UIDBCodes type / value pair in the APLResults message.

4.2.2.3 Load Phone Number into Scratchpad

This command shall download an entire phone number into the phone's Scratchpad Buffer, replacing the Scratchpad Buffer's previous contents. The reply message to this command should contain the number of Dial Digits successfully loaded. If this is done, it shall use the UBinary_08 UnitType, immediately following the UIDBCodes type / value pair in the APLResults message.

4.2.2.4 Dial Active Number

This command shall cause whatever dial string is in the Scratchpad to be dialed as a call initiation. The reply message to this command should contain the number of Dial Digits successfully loaded. If this is done, it shall use the UBinary_08 UnitType, immediately following the UIDBCodes type / value pair in the APLResults message.

4.2.2.5 Mute the Microphones

This command shall mute the audio path to the phone, whatever the location and connectivity of the microphone(s) being used. In other words, the uplink audio is killed.

4.2.2.6 UnMute the Microphones

This command shall restore the audio path to the phone, whatever the location and connectivity of the microphone(s) being used. In other words, the uplink audio is restored.

4.2.2.7 Mute Audio Output

This command shall mute the audio path from the phone, whatever the location and connectivity of the speaker(s) being used. This operation refers only to the muting of the telephone's audio output.

4.2.2.8 Un-Mute Audio Output

This command shall restore the audio path from the phone, whatever the location and connectivity of the speaker(s) being used. This operation refers only to the (un-)muting of the telephone's audio output.

4.2.2.9 Adjust Ringer Volume

These commands shall adjust the volume of the ring tone(s) produced by the phone to signal an incoming call. Each command shall affect the volume up or down to the phone's next ring signal volume level. When a command is received by the phone which would adjust the volume beyond the maximum or minimum level, it shall respond with a return code of IRFailure.

4.2.2.10 Request Keypresses / Stop Keypresses / Keypress Report

The Request Keypresses command directs the telephone to report future keypresses entered on its internal keypad. Those keypresses shall be reported using the Keypress Report Datagram message. A Node shall use the Stop Keypresses command when it is no longer interested in receiving the Keypress information.

If a single Node is currently requesting the Keypress information, the report shall be delivered as a point to point message. If multiple Nodes are currently requesting the Keypress information, the report shall be delivered as a broadcast message. When no Nodes are currently requesting the Keypress information, the telephone shall stop transmitting the Keypress Report Datagram messages.

4.2.2.11 Keypress Command

This command shall cause the phone to perform the same function that it would have performed if a user had pressed the specified key. A telephone is at liberty to implement Keypress Command capabilities for Keypress values that do not correspond to physical buttons on its control panel. However, only Keypress values defined in this document shall be used.

4.2.3 PhoneStatus Structure Definition

The PhoneStatus structure contains information on the current state of the telephone unit. For all single bit fields within the structure, a '1'B value shall mean "yes" and a '0'B value shall mean "no", unless otherwise specified.

```
PhoneStatus    ::== SEQUENCE {
   -- octet 0
   datalength  INTEGER (0..15),        -- # of octets following
   phoneison   INTEGER (0..1),
   dockedcrdle INTEGER (0..1),         -- Phone Docked in Cradle?
   conntopas   INTEGER (0..1),         -- Phone connected to PAS?
   handsfree   INTEGER (0..1),         -- Phone in Hands Free mode?
   -- octet 1
   handset     INTEGER (0..1),         -- 0: headset, 1: handset
   svcavail    INTEGER (0..1),         -- Service Available?
   roaming     INTEGER (0..1),         -- Phone Roaming?
   systembusy  INTEGER (0..1),         -- System is busy?
   ringing     INTEGER (0..1),         -- Ringing – Incoming Call?
   voicenow    INTEGER (0..1),         -- Voice call in progress?
   reserved0   INTEGER (0..3),         -- For level 0 phone status
   -- octet 2
   phlocked    INTEGER (0..1),         -- Phone is locked?
   keylocked   INTEGER (0..1),         -- Keypad is locked?
   outenable   INTEGER (0..1),         -- Speaker Enabled (0: mute)
```

```
   inenable    INTEGER (0..1),         -- Microphone Enabled?
   callmissed  INTEGER (0..1),         -- Incoming Call was missed?
   waitans     INTEGER (0..1),         -- Outgoing call pending?
   reserved1   INTEGER (0..3),         -- For level 1 phone status
   -- octet 3
         -- Defined in 4.3.4
   -- octets 4-16 are reserved
}
```

4.2.4 KeyList Structure Definition

The KeyList structure provides for an expandable set of key definitions. The most common keys are defined first, such that they can fit within the first octet. The first octet is divided into two semi-octets. The most significant semi-octet is used to specify expansion of up to 16 additional octets worth of keypress values. The least significant semi-octet is used to specify the 16 most common keys.

The KeyList structure shall always be the minimum size necessary to indicate the intended keypress. The intended keypress shall always be found in the last octet of the structure. This means that if the structure has an octet 1, the least significant semi-octet of octet 0 (zero) shall be ignored on receipt and shall be set to '0'H when transmitted. Similarly, if octet 2 is the last octet, the keypress information shall be found there, with the least significant semi-octet of octet 0 and all of octet 1 ignored on receipt and set to '00'H when transmitted. This continues through octet 16 (the 17th octet of the structure).

The most common keys are the digits 0-9, *, #, “Clear Digit”, “Clear Display”, “Power”, and “Send”. They are represented by values '0'H through 'F'H, respectively.

Both “standard” and “additional” keypress code values are defined in Table 2.

```
KeyList  ::== SEQUENCE {
   -- octet 0
   datalength  INTEGER (0..15),        -- # of octets following
   commonkey   INTEGER (0..15),        -- Common keypress codes
   -- octet 1
   keys01      INTEGER (0..255),       -- Additional keypress codes
   -- octets 2-16 are reserved
}
```

4.2.5 PhoneCapability Structure Definition

The PhoneCapability structure contains information on the capabilities of a phone. For all single bit fields within the structure, a '1'B value shall mean “yes” and a '0'B value shall mean “no”, unless otherwise specified.

```
PhoneCapability      ::== SEQUENCE {
   -- octet 0
   datalength  INTEGER (0..15),        -- # of octets following
   reserved0   INTEGER (0..1),         -- reserved
   amps        INTEGER (0..1),         -- AMPs mode
   cdma800     INTEGER (0..1),         -- CDMA 800 MHz mode
   cdma19      INTEGER (0..1),         -- CDMA 1.9 GHz mode
   -- octet 1
   tdma800     INTEGER (0..1),         -- TDMA 800 MHz mode
   tdma19      INTEGER (0..1),         -- TDMA 1.9 GHz mode
   gsm800      INTEGER (0..1),         -- GSM 800 MHz mode
   gsm18       INTEGER (0..1),         -- GSM 1.8 GHz mode
   gsm19       INTEGER (0..1),         -- GSM 1.9 MHz mode
   pdc800      INTEGER (0..1),         -- PDC 800 MHz mode
```

```
   pdc15       INTEGER (0..1),         -- PDC 1.5 GHz mode
   reserved1   INTEGER (0..1),         -- reserved
   -- octet 2
   -- Phone Capabilities
   voicerecog  INTEGER (0..1),         -- Voice Recognition
   necancel    INTEGER (0..1),         -- Noise/Echo Cancellation
   callwait    INTEGER (0..1),         -- Call Waiting
   voicemsg    INTEGER (0..1),         -- Voice Message
   textmsg     INTEGER (0..1),         -- Text Message
   dataxmit    INTEGER (0..1),         -- Data Transmit
   pda         INTEGER (0..1),         -- PDA
   reserved2   INTEGER (0..1),         -- reserved
   -- octets 3-16 are reserved
}
```

4.2.6 PhoneMode Structure Definition

The PhoneMode structure contains information on the current mode(s) in which it is operating. For all single bit fields within the structure, a '1'B value shall mean “yes” and a '0'B value shall mean “no”, unless otherwise specified.

```
PhoneMode      ::== SEQUENCE {
   -- octet 0
   datalength  INTEGER (0..15),        -- # of octets following
   reserved0   INTEGER (0..1),         -- reserved
   amps        INTEGER (0..1),         -- AMPs mode
   cdma800     INTEGER (0..1),         -- CDMA 800 MHz mode
   cdma19      INTEGER (0..1),         -- CDMA 1.9 GHz mode
   -- octet 1
   tdma800     INTEGER (0..1),         -- TDMA 800 MHz mode
   tdma19      INTEGER (0..1),         -- TDMA 1.9 GHz mode
   gsm800      INTEGER (0..1),         -- GSM 800 MHz mode
   gsm18       INTEGER (0..1),         -- GSM 1.8 GHz mode
   gsm19       INTEGER (0..1),         -- GSM 1.9 MHz mode
   pdc800      INTEGER (0..1),         -- PDC 800 MHz mode
   pdc15       INTEGER (0..1),         -- PDC 1.5 GHz mode
   reserved1   INTEGER (0..1),         -- reserved
   -- octet 2
   -- Phone Capabilities / Modes
   voicerecog  INTEGER (0..1),         -- Voice Recognition
   necancel    INTEGER (0..1),         -- Noise/Echo Cancellation
   callwait    INTEGER (0..1),         -- Call Waiting
   voicemsg    INTEGER (0..1),         -- Voice Message
   textmsg     INTEGER (0..1),         -- Text Message
   dataxmit    INTEGER (0..1),         -- Data Transmit
   pda         INTEGER (0..1),         -- PDA
   reserved2   INTEGER (0..1),         -- reserved
   -- octets 3-16 are reserved
}
```

4.2.7 NetworkCapability Structure Definition

The NetworkCapability structure is intended to provide all combinations of cellular or wireless network of which the telephone is capable of using. In this version of the document, the structure definition is incomplete, requiring input from the cellular / wireless carriers. In this version of the document, no IDB

messages reference this structure. For all single bit fields within the structure, a '1'B value shall mean “yes” and a '0'B value shall mean “no”, unless otherwise specified.

Note: Although there are messages to access it, the NetworkCapability structure is not yet defined.

```
NetworkCapability    ::== SEQUENCE {
   -- octet 0
   datalength  INTEGER (0..15),        -- # of octets following
   tbd         INTEGER (0..15),        -- All Elements TBD
   -- octets 1-16 are TBD
}
```

4.2.8 NetworkMode Structure Definition

The NetworkMode structure contains information on the type and state of the phone system(s) to which the telephone is connected. For all single bit fields within the structure, a '1'B value shall mean “yes” and a '0'B value shall mean “no”, unless otherwise specified.

Note: Although there are messages to access it, the NetworkMode structure is not yet defined.

```
NetworkMode    ::== SEQUENCE {
   -- octet 0
   datalength  INTEGER (0..15),        -- # of octets following
   tbd         INTEGER (0..15),        -- All Elements TBD
   -- octets 1-16 are TBD
}
```

4.2.9 Telephone Keypress Codes

In Table 2, the correspondence between codes in the KeyList structure and the keypress meanings is defined. Each code is given as a two-part value. The first part indicates the octet in which the value is meaningful. The second part indicates the value in that octet that corresponds to the given keypress.

All code values that are not specified by this document are reserved. This includes octet 1 values '0E'H through 'FF'H and all values for octets 2 through 16. As new codes are assigned, consideration should be given to assigning more commonly used keypresses to earlier octets and less commonly used keypresses to later octets, to optimize the use of network bandwidth.

Keypress values are listed based upon the meaning of the keypress, regardless of what specific legend is present on or near the physical key.

Table 2 — IDB Telephone Keypress Codes

IDB Telephone Keypress Codes

Code	Keypress	Code	Keypress	Code	Keypress
[Standard Keypress Group]					
0, '0'H	“0”	0, '1'H	“1”	0, '2'H	“2”
0, '3'H	“3”	0, '4'H	“4”	0, '5'H	“5”
0, '6'H	“6”	0, '7'H	“7”	0, '8'H	“8”
0, '9'H	“9”	0, 'A'H	“*”	0, 'B'H	“#”
0, 'C'H	“Clear Digit”	0, 'D'H	“Clear Display”	0, 'E'H	“Power”
0, 'F'H	“Send”				
[Additional Keypress Group]					
1,'00'H	“+”	1,'01'H	“End”	1,'02'H	“Recall”
1,'03'H	“Function”	1,'04'H	“Store”	1,'05'H	“Enter”
1,'06'H	“Back”	1,'07'H	“Scroll Up”	1,'08'H	“Scroll Down”

IDB Telephone Keypress Codes

Code	Keypress	Code	Keypress	Code	Keypress
1,'09'H	“E911”	1,'0A'H	“Emergency Assistance”	1,'0B'H	“Roadside Assistance”
1,'0C'H	“Information Assistance”	1,'0D'H	“Pause”		

4.3 Level 2 Capabilities

Telephones that are IDB Compliant (supporting all Level One Capabilities) that also support and act appropriately on all messages included in Telephone Functional Level 2 shall be deemed to be IDB Level 2 Compliant.

4.3.1 Message Responses

The IDB J2366-7 Application Message Layer protocol provides a mechanism for messages in response to Command or Request messages. These response messages incorporate the IDBResult code. Several values for the IDBResult have been defined in J2366-7LX.

IDB Compliant telephone shall support the generation and interpretation of response status messages that incorporate the IDBResult code.

The IRCrcError value does not apply to the messages defined in this document.

The IRDeviceNotPresent value does not apply to Level Two Capabilities messages.

The IRNotSupported value does not apply to Level Two Capabilities messages.

The IRSecurityError value does not apply to Level Two Capabilities messages.

4.3.2 Telephone Level 2 Capabilities General Requirements / Notes

A telephone should detect whether a GPS Node is active on the IDB. If a GPS Node is detected, the phone should query the current date / time from it and set its internal clock automatically.

An IDB Level 2 Compliant telephone shall support the “Date / Time Request” and “Date / Time Reply” messages found in IDB-0004. An IDB Level 2 Compliant telephone should support the “Date / Time Set Command” message found in IDB-0004.

4.3.2.1 Respect Locked / Unlocked State

A telephone that uses an “unlock code” to allow access to certain functions is in the “Locked State” when the code has not been entered to enable such access. A telephone that does not provide such a capability, or which has had the unlock code entered, is in the “Unlocked State”. A telephone that is in the “Locked” state shall not allow any IDB operation that would not be allowed (due to the lack of enabled access) via an attached keypad.

4.3.2.2 Internal Display Access

IDB Compliant telephones providing IDB access to their internal display(s) shall implement the display handling documented in IDB-0009.

4.3.2.3 Start With First

Use of the Get First / Next / Previous messages to access each member of a list shall begin with a Get First. Behavior of Get Next or Get Previous without a preceding Get First (subsequent to the telephone joining the IDB) is undefined.

Requests for members of lists that are inherently chronological lists (*e.g.*, last numbers dialed) shall begin by returning the most recent such member for the Get First operation. Each Get Next operation shall cause the next less recent member to be returned. Each Get Previous operation shall cause the next more recent member to be returned.

4.3.2.4 Maintain List Position

A telephone should maintain track of the state of each Get First / Next / Previous for each IDB node that has issued a Get First. Multiple devices accessing a telephone via Get Next / Previous that does not maintain such state information may receive misleading results.

4.3.2.5 Return Last Phone Number List Entry Indication

A telephone that successfully returns a telephone number list entry that is the last entry in the requested direction shall indicate this condition. This shall be done by including an additional Type / Value pair in the message that includes the returned information. The Type shall be a UBoolean with a Value of “T”. If there is a single entry in the specified list, any request (First, Next, or Previous) that references that list shall indicate the “last” condition, as described in this paragraph.

A telephone number list entry that is not the last entry shall not include the extra UBoolean Type / Value pair in its returned message.

4.3.3 Telephone Level 2 Messages

The following list of telephone messages is a set of IDB messages that may be implemented in addition to the set of required messages. This list is based upon the TIA Level 2 message group. Implementation of all Level 2 messages is required for Level 2 Compliance.

Additional explanation and definition of these messages can be found in the sub-paragraphs that follow Table 3.

Table 3 — IDB Telephone Level 2 Messages

IDB Telephone Level 2 Messages

Message	**MapClass** ***UnitType***	**Action** ***Amount***	**Modifier** ***UnitType***	**Object** ***Amount***	**Feature** ***UnitType***
[Phone Information Group]					
Receive Signal Strength Request	Request	AReport	InInput		SignalStrength
Receive Signal Strength Report	Reply *UBinary_08*	ANone *'xx'H*			
Battery Charge Status Request	Request	AReport	Current		Battery
Battery Charge Status Report	Reply *UBinary_08*	ANone *nn*			
Request Telephone’s Numbers	Request	AReport	All / Current		PhoneNumber
Telephone’s Numbers Reply	Reply *UPhoneNumber*	ANone *##########*			
[Call Processing Group]					
Call Timer Request	Request	AReport	Current		Time

IDB Telephone Level 2 Messages

Message	**MapClass** ***UnitType***	**Action** ***Amount***	**Modifier** ***UnitType***	**Object** ***Amount***	**Feature** ***UnitType***
Call Timer Reply	Reply *UDay* *Uhhmmss*	ANone *nn* *hhmmss*	*Uyyyymmdd*	*yyyymmdd*	
Caller ID Request	Request	AReport	Incoming		PhoneNumber
Caller ID Reply	Reply *UvCard*	ANone *[vCard Data]*			
[Audio Management Group]					
(No Level 2 Messages at this time)					
[Network Information Group]					
(No Level 2 Messages at this time)					
[User Interface Group]					
(No Level 2 Messages at this time)					
[Phonebook Management Group]					
Get Phone Number Capacity	Request	AReport		WhitePages	Capacity
Phone Number Capacity	Reply *UBinary_**	ANone *[Total Value]*	*UBinary_**	*[Avail Value]*	
Clear Phone Number Information	Command *UBinary_**	Clear *[value]*		WhitePages	
Clear All Phone Number Information	Command	Clear	All	WhitePages	
Recall Phone Number Information	Request *UBinary_**	AReport *[value]*			PhoneNumber
Get First/Next/Previous Stored Phone Number Info	Request	AReport	First / Next / Previous	WhitePages	PhoneNumber
Store Phone Number Information	Command *UBinary_**	Save *[value]*	*UvCard*	WhitePages *[vCard Data]*	PhoneNumber
Get First/Next/Previous Last Number Dialed	Request	AReport	First / Next / Previous	Output	PhoneNumber
Get First/Next/Previous Last Number Received	Request	AReport	First / Next / Previous	Input	PhoneNumber
Phone Number Information Reply	Reply *UvCard* *UBoolean*	ANone *[vCard Data]* *"T"*	*UBinary_**	*[value]*	
[Keyboard & Display Group] — Redundant with "User Interface"??? Eliminate???					
(No Level 2 Messages at this time)					
[Data Exchange Group]					
(No Level 2 Messages at this time)					
[Miscellaneous Group]					
(No Level 2 Messages at this time)					

4.3.3.1 Signal Strength Report

The Signal Strength Report messages are intended to provide a means of remote replication of the signal strength display internal to the telephone, or which might be internal to the telephone. It is not intended to provide an absolute and standardized measure of signal strength or quality.

The UBinary_08 returned in the Report is divided into two four-bit values. The most significant four bits shall be used to represent the number of steps of signal quality provided by the phone. The least significant four bits shall be used to represent the number of steps of signal quality that are currently true.

```
SignalStrength ::== SEQUENCE {
   availablestrength INTEGER (0..15),        -- # of “steps” available
   currentstrength   INTEGER (0..15),        -- # of “steps” currently
}
```

4.3.3.2 Battery Charge Status Report

The Value returned by the Battery Charge Status Reply shall be a single octet. The most significant bit of this value shall be '1'B if the battery is being charged, and '0'B if it is not being charged. The remaining seven bits of the octet shall contain a value from 0 to 100, which shall indicate the percentage of charge remaining in the telephone’s internal battery. Telephones may limit the precision of their reported charge state to the (rounded up) 10% of maximum charge.

4.3.3.3 Telephone’s Numbers

A “Request Telephone Number” message shall be used to retrieve the phone numbers by which the telephone instrument is “known”. When specified with a Modifier of “Current”, only those (zero or more) numbers that are currently valid shall be returned. When specified with a Modifier of “All”, all numbers associated with the telephone shall be returned, whether currently valid or not. The response is the “Telephone Numbers Reply” message. Although shown with a single phone number being returned, the UPhoneNumber / value pairs shall be repeated in a single Reply message, with one pair used for each number being reported.

4.3.3.4 Phone Number Capacity

The “Get Phone Number Capacity” request allows a Node to learn how many Phone Number memory entries exist in the telephone and how many of those entries are available. The “Phone Number Capacity” reply shall provide the total capacity first, and the number of available entries second.

4.3.3.5 Clear Phone Number Information / Clear All Phone Number Information

The “Clear Phone Number Information” command shall cause the specified or all internal telephone Phone Number memories to be cleared.

This “All” variant of this command is intended to be used prior to a phone directory backup restoration, or similar operation.

4.3.3.6 Get First / Next / Previous Stored Phone Number Info

This set of messages enables a Node to peruse the internal telephone Phone Number memories’ contents.

4.3.3.7 Store Phone Number Information

A variant on this messages is to use a UBoolean type with a “T” value, instead of the UBinary indication of a specific location. When a phone receives a message of this format, it shall store the data in any available memory location. It should store the data into the first available memory location.

4.3.3.8 Phone Number Information Reply

The UBinary type / value pair in this message is an optional part of the message. It should be provided on responses to the “Get First / Next / Previous Stored Phone Number Info” messages. When provided, it shall contain a “handle” which can be used by the Clear, Recall, and Store Phone Number messages that accept a UBinary type / value pair to specify a particular memory location within the phone.

4.3.4 PhoneStatus Structure Definition

The PhoneStatus structure contains information on the current state of the telephone unit. For all single bit fields within the structure, a '1'B value shall mean “yes” and a '0'B value shall mean “no”, unless otherwise specified.

```
PhoneStatus    ::== SEQUENCE {
   -- octet 0
         -- Defined in 4.2.3
   -- octet 1
         -- Defined in 4.2.3
   -- octet 2
         -- Defined in 4.2.3
   -- octet 3
   lowbat       INTEGER (0..1),          -- Battery Low?
   necancel     INTEGER (0..1),          -- Noise/Echo Cancel On?
   callwaits    INTEGER (0..1),          -- Call is Waiting?
   emergcall    INTEGER (0..1),          -- Emergency Call in progress?
   datacall     INTEGER (0..1),          -- Data Call in progress?
   reserved2    INTEGER (0..7)           -- For level 2 phone status
   -- octets 4-16 are reserved
}
```

4.4 Level 3 Capabilities

Telephones that are IDB Compliant (supporting all Level One Capabilities) that are also IDB Level 2 Compliant (supporting all Level Two Capabilities) that support any Level 3 Capabilities shall be deemed to be IDB Level 3 Capable. The Level 3 Capabilities in this document shall be considered draft information or a statement of intent for more advanced capabilities / features. No telephone shall be IDB Level 3 Capable, based upon this revision of the document.

4.4.1 Message Responses

The IDB J2366-7 Application Message Layer protocol provides a mechanism for messages in response to Command or Request messages. These response messages incorporate the IDBResult code. Several values for the IDBResult have been defined in J2366-7LX.

IDB Compliant telephone shall support the generation and interpretation of response status messages that incorporate the IDBResult code.

The IRCrcError value does not apply to the messages defined in this document.

The IRDeviceNotPresent value does not apply to Level Three Capabilities messages.

The IRNotSupported value may apply to Level Three Capabilities messages.

The IRSecurityError value may apply to Level Three Capabilities messages.

4.4.2 Telephone Level 3 Capabilities General Requirements / Notes

The Level 3 Capabilities in this document shall be considered draft information or a statement of intent for more advanced capabilities / features.

4.4.3 Telephone Level 3 Messages

The following list of telephone messages is a set of IDB messages that may be implemented in addition to the messages in the required and Level 2 message sets. This list is based upon the TIA Level 3 message group. Implementation of any Level 3 messages is sufficient for Level 3 Capability.

Additional explanation and definition of these messages can be found in the sub-paragraphs that follow Table 4.

Table 4 — IDB Telephone Level 3 Messages

IDB Telephone Level 3 Messages

Message	**MapClass** *UnitType*	**Action** *Amount*	**Modifier** *UnitType*	**Object** *Amount*	**Feature** *UnitType*
[Call Processing Group]					
Emergency Call w/Number	**TBD**				
[Security Group]					
Get Unlock/Security Codes	Request	AReport	First / Next		Locks
Code Reply	Reply *UBinary_**	ANone *[value]*			

4.4.3.1 Emergency Call with Number

The precise meaning and usage of this message is still being discussed as of the publication of this document. It is included here as a kind of “place holder” and discussion point. Discussions are centering on how a telephone could provide the ability to dial a supplied number in an emergency situation (such as to a 3rd party dispatch service) even when the telephone is locked, while preventing any arbitrary number from being dialed (*e.g.*, by command from an IDB attached PDA), bypassing the user’s security codes.

Ideas floated for this capability include:

- The telephone could have one or more pre-approved “emergency” numbers that are callable without using the “unlock” code. These would be allowed in this message even if the phone were locked.
- If the phone was not locked, any phone number would be accepted.
- If the phone was locked and a non-pre-approved “emergency” number were specified, a failure indication would be returned.
- If the phone was locked and a non-pre-approved “emergency” number were specified, one / all of the pre-approved “emergency” number(s) would be called, instead.

During the discussion in this area, the idea that the unlock code would affect the ability to access the various phone lists was raised. Perhaps a locked phone would make available (via Get requests) only those numbers that had been designated as pre-approved “emergency” numbers that are dialable while the phone is in the locked state.

4.4.3.2 Security Codes

A telephone may respond to the “Get Unlock / Security Codes” message with an indication that the function is not supported. The message, when specifying a Modifier of “First” shall be a request for the telephone’s “Unlock” code. The message, when subsequently specifying a Modifier of “Next” shall be a request for the telephone’s “Security” code. Additional codes are not yet defined. A request specifying a Modifier of “Next” beyond the number of defined codes shall result in a response that the function is not supported.

Note: A telephone that fully implements the providing of codes in response to these messages eliminates most of the utility of having such codes for securing access to the phone against all but the most casual misuse.

5. Notes

Prepared by Strategis Consulting Inc., on behalf of the IDB Forum

Telephony information was derived from Telecommunications Industry Association (TIA) contributions, authored by Visteon – TR45.1.4/99.06.25 (dated 06/18/99) and TR45.1.4/99.08.19.__ (dated 8/19/99). The IDB Forum thanks Visteon for allowing it to derive this document, in part, from their copyright TIA contributions.

Printed in U.S.A.