ASM **Technology Singapore**
2, Yishun Avenue 7, Singapore 768924. Tel: (65)-67526311 , Fax: (65)-67582287

ASM EAGLE XTREME/TWIN EAGLE XTREME WIREBONDER

Host Programmers Manual

Revision 1.15
5 July 2012

Table of Contents

1. Revision History

Rev	Document Generator	Date	Reviewers	Description of Changes
1.0	Baoping Li	25 May 2009		• Base VERSION of Eagle Xtreme
1.1	Chen Chen	23 July 2009		• Standardize CEID, SVID with Eagle60
1.2	Chen Chen	18 May 2010		• Add auto save finish event
1.3	Chen Chen	23 May 2010		• Redefine all the alarm ID for Xtreme
1.4	Chen Chen	01 June 2010		• Add new SVID for scrub parameters
1.5	Chu Yue	07 June 2010		• Add new CEID for message recognition. • Modify CEID 51 definition.
1.6	Chen Chen	06 Aug 2010		• Add new CEID, SVIDs.
1.7	Chen Chen	21 Oct 2010		• Add BSOB wire parameter in CEID 78 • Update Alarm ID.
1.8	Justin/Chu Yue	26 Dec 2010		• Add alarm ID
1.9	Justin	14 Jan 2011		• Removed unused/duplicate SVIDs
1.10	Chen Chen	16 May 2011		• Add create default strip map function.
1.11	Chu Yue	1 Sep 2011		Add CEID RPTID SVID for Idealine Add Multi Lot sequence
1.12	Chen Chen	06 Jan 2012		Add CEID 117 && 118 for capillary count hit warning limit and stop limit.
1.13	Chu Yue	24 April		Add CEID 47&&48 SVID 1400~1414, REPORTID125,CEID 125
1.14	Chu Yue	18 May		Add SVID RPTID CEID
1.15	Sun Zhaoyan	5 July 2012		Add SVID, CEID, ALARM ID

2. Introduction

2.1. DOCUMENT HISTORY

This is the first release version of ASM Eagle Xtreme Gold Wire Bonder SECS Host Programmer's Manual.

2.2. SCOPE

This document defines ASM Eagle Xtreme/Twin Eagle Xtreme/Harrier Xtreme Gold Wire Bonder SECS-II communication capabilities and interactions between the wire bonder and a host computer. It is limited to the specific behavior of the wire bonder in a communication link. Communications in this document are intended for an equipment-to-host relationship and do not address equipment-to-equipment application.

2.3. INTENT

This document serves as an introduction to the capabilities of Eagle Xtreme Gold Wire Bonder in SECS communication aspect. It is also intended to provide a basis for implementing the interaction functions for the host computer with the wire bonder.

2.4. OVERVIEW

This document is divided into sections as described below.

SECTION 2 : Introduction
This section provides the revision history, scope and intent of this document. It also provides an overview of the structure of this document.

SECTION 3 : State Model
This section describes the behavior of Eagle Xtreme Gold Wire Bonder from a host perspective.

SECTION 4 : Equipment Capabilities
This section provides a brief description of the communication capabilities defined for and supported by the wire bonder. The description of each capability includes the purpose and a brief description.

SECTION 5 : SECS Message Subset
This section provides a composite list of the SECS-II message required to implement all capabilities described.

2.5. APPLICABLE DOCUMENTS

This document references specific portions of the SEMI standards listed below:

SEMI E4-0699 SEMI Equipment Communication Standard 1
470 Message Transfer (SECS-I)
SEMI E5-0999 SEMI Equipment Communication Standard 2
1. Message Content (SECS-II)
SEMI E30-0200A Generic Model for Communication and Control of SEMI Equipment (GEM)
SEMI E37 High Speed SECS Message Services
-Comparison of SECS-1 and HSMS

3. STATE MODELS

A state model is a collection of states and state transitions which combine to describe the behavior of a system. The basic unit of a state model is the state. A state is a static set of conditions. If the conditions are met, the state is current. These conditions may be stimulated by the operator input at the wire bonder, received SECS message, expired timer, and changes in the sensors reading.

This model includes definition of the conditions which delineate a state, the action/reactions possible within a state, the event which triggers transitions to other states, and the process of transiting between states. The following section contains the finite state models for Eagle Xtreme Gold Wire Bonder. Supporting figures are also included. These state models describe the behavior of the wire bonder from a host perspective in a compact and easy to understand format.

3.1. COMMUNICATIONS STATE

The communications state model defines the behavior of the wire bonder in relation to the existence or absence of a communication link with the host. It can be viewed as a logical connection between equipment and host rather than a physical connection.

There are two major states of SECS communications, DISABLED and ENABLED, where the ENABLED state has two sub states, NOT COMMUNICATING and COMMUNICATING.

DISABLED
In this state SECS-II communication with a host computer is non-existent. If the operator switches from ENABLED to DISABLED, all SECS-II communications cease immediately. All messages queued to

send shall be discarded, and all further action on any open transactions and conversation shall be terminated.

A simulated ICON is displayed on the screen to give feedback on the current communications state.

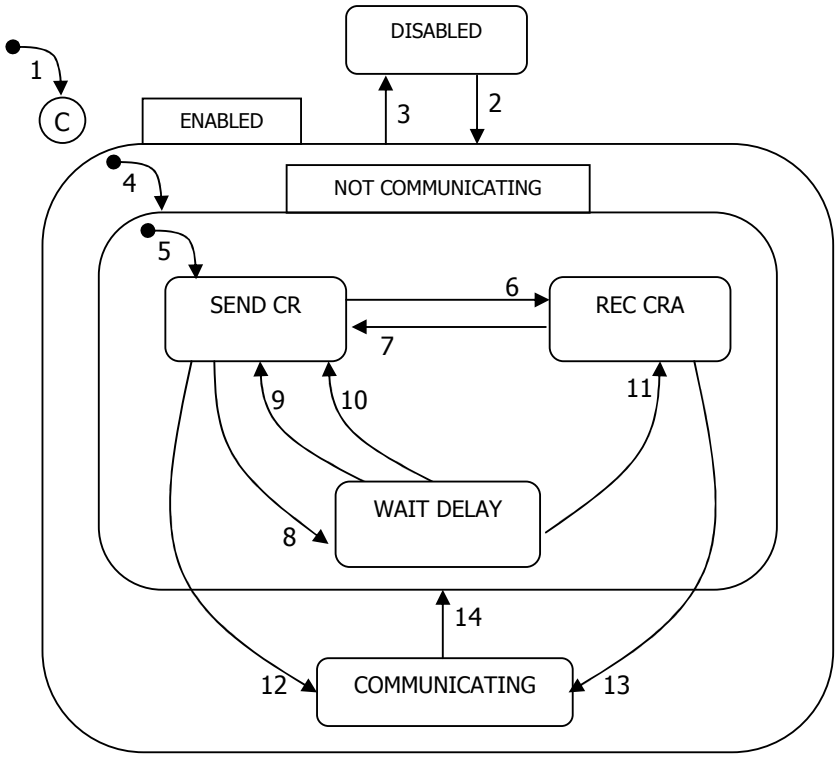
Not Exists	- DISABLED
RED	- NOT COMMUNICATING
GREEN / YELLOW	- COMMUNICATING (Color depends on Control State)

ENABLED
Whenever communication is enabled, either during system initialization or through operator selection, the sub state NOT COMMUNICATING is active until communications are formally established.

NOT COMMUNICATING
No message other than S1F13, S1F14 and S9Fx shall be sent while the wire bonder is in this sub state. The wire bonder will discard all messages received from the host other than S1F13 and S1F14. It also periodically attempts to establish communications with the host computer by issuing S1F13 until communications are successfully established.

COMMUNICATING
In this sub state, SECS communications with a host computer must be maintained. In the event of communications failure, the wire bonder will return to the NOT COMMUNICATING sub state and attempt to re-establish communications with the host system. The wire bonder will remain in this sub state until communications are disabled or a communication failure occurs.

ASMGEM Communication State Diagram



ASMGEM Communication State Table

#	CURRENT STATE	TRIGGER	NEW STATE	ACTION	COMMENT
1	(Entry to COMMUNICATIONS)	System Initialization	System Default	None	System default may be ENABLED or DISABLED
2	DISABLED	Operator switches from DISABLED to ENABLED	ENABLED	None	SECS II communications are started
3	ENABLED	Operator switches from ENABLED to DISABLED	DISABLED	The simulated LED is removed	SECS II communications are prohibited
4	(Entry to ENABLED)	Any entry to ENABLED state	NOT COMMUNICATING	A simulated RED LED is displayed on the top right of the screen	May enter through system initialization to ENABLED, or through operator switch to ENABLED.
5	NOT COMMUNICATING	Any entry to NOT COMMUNICATING	SEND CR	Initialize communications, send S1,F13	Begin to establish communications, may receive S1,F13 from Host
6	SEND CR	Received S1,F13 from Host	REC CRA	Decode message and attempt to send S1,F14	If S1,F13 previously sent, wait for S1,F14 reply
7	REC CRA	COMMACK <> 0 of received S1,F14 or failed to send S1,F14	SEND CRA	Send S1,F13	Host disallowed a connection so try again
8	SEND CR	Connection Transaction Failure	WAIT DELAY	Initialize CommDelay timer. Dequeue all messages queued to send	Place dequeued messages on the spool buffer if spool enabled
9	WAIT DELAY	CommDelay Timer expired	SEND CR	Send S1,F13	Wait for S1,F14, may receive S1,F13 from Host
10	WAIT DELAY	Received a message other than S1,F13	SEND CR	Discard message, no reply and send S1,F13	Indicates opportunity to establish communication
11	WAIT DELAY	Received an S1,F13 message	REC CRA	Decode message and send S1,F14	Host attempting to establish communications
12	SEND CR	Received S1,F14 with COMMACK = 0	COMMUNICATING	The simulated LED becomes GREEN / YELLOW	Communication has been established
13	REC CRA	Sent S1,F14 successfully and any pending S1,F14 from Host completed without Transaction Failure	COMMUNICATING	The simulated LED becomes GREEN / YELLOW	Communication has been established. If a Transaction Fault occurs, send S9,F9
14	COMMUNICATING	Transaction Failure occurs	NOT COMMUNICATING	The simulated LED becomes RED	Communication has been lost

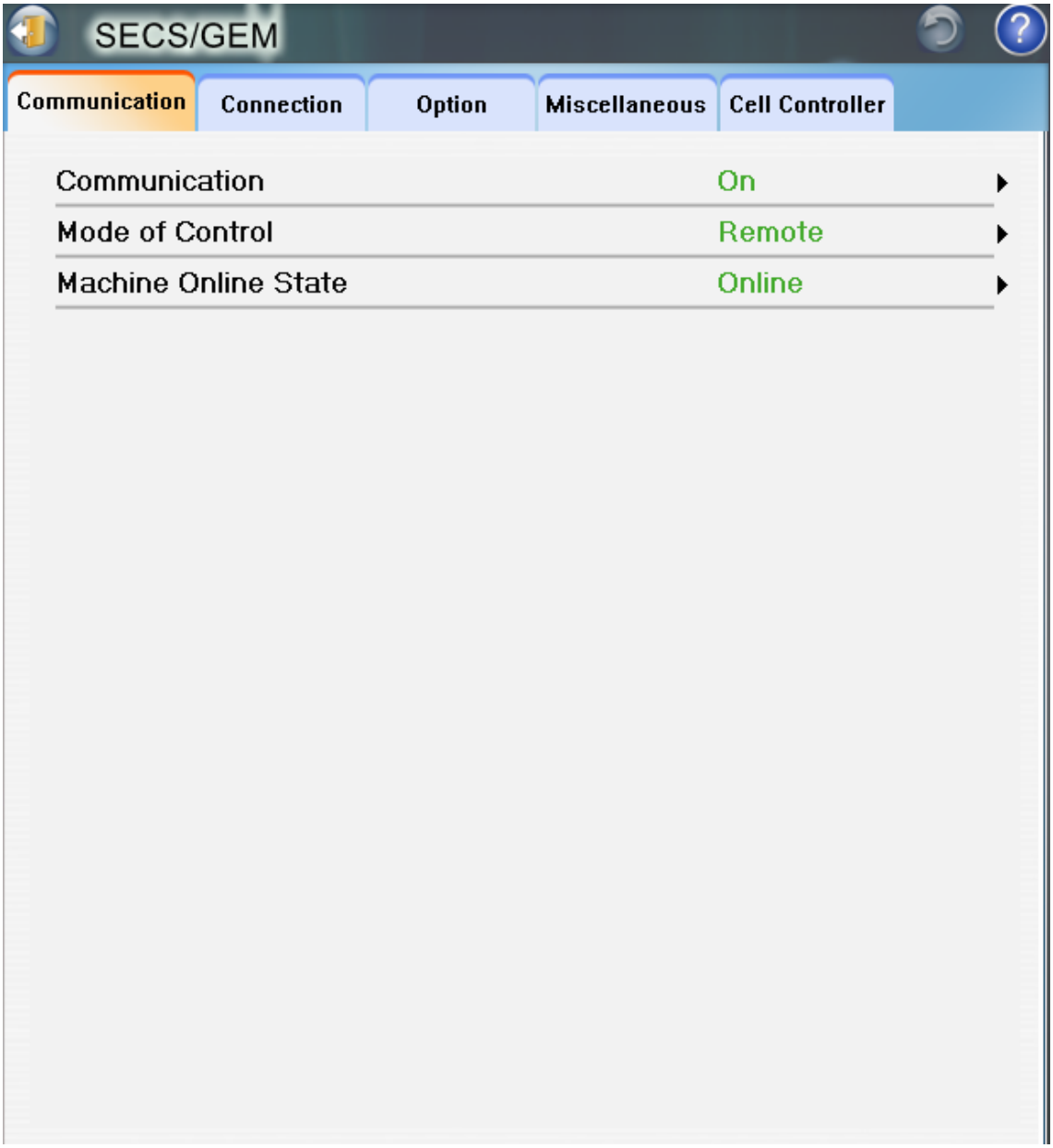
Upon equipment power-up, the wire bonder will establish its default setting for all constants and variables retained in non-volatile memory. Once system initialization has been achieved, the wire bonder will attempt to establish communications with the host computer. A successful attempt will cause the wire bonder to fall into COMMUNICATING STATE, whereas a failure will result in NOT COMMUNICATING state.

Once initialization has completed, the operator is able to enable/disable the communication at any time, by toggling the enable flag shown in the [Setup] [SECS/GEM Communication] [SECS/GEM Options] menu. There are no collection events linked to the Communications State Model.

3.2. CONTROL STATE

The intent of this capability is to allow flexibility in the control of the wire bonder in the manufacturing environment. The control state model defines the level of action / data extraction that can be exercised on the wire bonder by the host system.

The control state allows the Host 3 basic levels of control operations on the equipment, being OFFLINE, ONLINE/LOCAL and ONLINE/REMOTE.



The operator can select between these 3 levels by changing the menu items
1. **Mode Of Control (REMOTE / LOCAL)**
2. **Machine Online State (ONLINE / OFFLINE)**

in the [Configuration][Factory Automation] [SECS/GEM] [Options] page.

The ONLINE/LOCAL control state is used when it is desired that the host have no direct control over the operation of the equipment but still allow the Host to obtain data. Where as the ONLINE/REMOTE control state is used to allow the host to have "front panel" control of the wire bonder and still allow data extraction.

The OFFLINE control state is used when no control or data collection will be allowed by the Host.

A simulated LED at the top right of the display screen indicates a summary of the control state:
RED - NOT COMMUNICATING
GREEN - ONLINE (LOCAL or REMOTE)
YELLOW - OFFLINE (User or Host initiated)
Not Exist - Communication DISABLED

A change in this control state will generate a collection event.

3.2.1. LOCAL

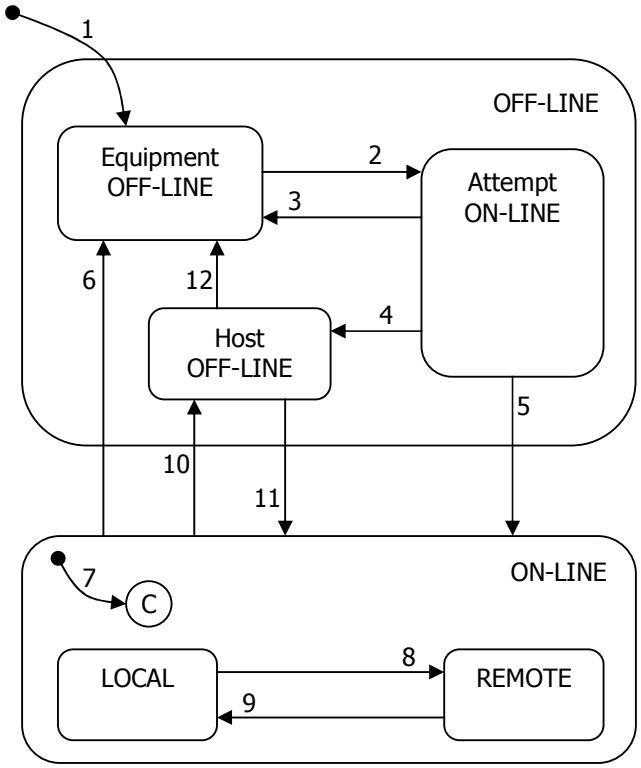
When the wire bonder is in the LOCAL control state, the operator has sole control of the wire bonder and its related process. The host has absolutely no control over any aspect of the wire bonder's process. Any host directed activity which has a direct effect on the process will not be accepted and the wire bonder will respond with an appropriate acknowledge code.

There is no restriction on data collection by the host when the wire bonder is in LOCAL control state.

3.2.2. REMOTE

In the REMOTE control state, the host has the capability to start or stop processing. However, the operator has ultimate authority in overriding host control of the wire bonder. In this mode the Host is able to send Remote Command S2,F21 and Host Command S2,F41 to direct the operation of the equipment.

ASMGEM Control State Diagram



ASMGEM Control State Table

#	CURRENT STATE	TRIGGER	NEW STATE	ACTION	COMMENT
1	(Undefined)	Entry into CONTROL state (system initialization)	Equipment OFF-LINE	None	Machine power up always resets the system to OFF-LINE
2	Equipment OFF-LINE	Machine Online State (menu) is set to ONLINE	Attempt ON-LINE	Send S1,F1	Trigger is based on the current menu setting, not on the action of toggling the state
3	Attempt ON-LINE	Communication failure during S1,F1 send	Equipment OFF-LINE	None	Unable to send the message causes a looping effect between Equipment OFF-LINE and Attempt ON-LINE
4	Attempt ON-LINE	Received S1,F0 or timeout from sending S1,F1	Host OFF-LINE	None	Assume that the Host is there because S1,F1 was successfully sent but incorrect reply.
5	Attempt ON-LINE	Host replied with S1,F2	ON-LINE	None	Host will be informed of the transition to ON-LINE at transition 7
6	ON-LINE	Operator toggles the Machine Online state (menu) to OFFLINE	Equipment OFF-LINE	Send event	"Equipment OFF-LINE" event is sent to the Host.
7	(Undefined)	Entry to ON-LINE state	(Sub-state conditional on REMOTE / LOCAL setting)	Send event	Sends either "Control State REMOTE" or "Control State LOCAL" depending on the Mode Of Control menu item
8	LOCAL	Operator toggles Mode Of Control to REMOTE	REMOTE	Send event	"Control State REMOTE" is sent to the Host. Machine is now able to accept remote commands from the Host.
9	REMOTE	Operator toggles Mode Of Control to LOCAL	LOCAL	Send event	"Control State LOCAL" is sent to the Host. Remote commands are no longer accepted
10	ON-LINE	Equipment accepts S1,F15 "Set OFF-LINE" message from Host	Host OFF-LINE	Send event	"Equipment OFF-LINE" event occurs
11	Host OFF-LINE	Equipment accepts request to go ON-LINE from Host, S1,F17	ON-LINE	None	Host will be informed of the transition to ON-LINE at transition 7
12	Host OFF-LINE	Operator toggles the Machine Online state (menu) to OFFLINE	Equipment OFF-LINE	None	Equipment was already OFFLINE so no new event is sent to the Host

NOTE: Communication is severely restricted under the OFFLINE states.
Equipment will only send
S1,F13 - To establish communication (Part of the Communication Model)
S1,F1 - To attempt to go ONLINE
S1,F14 - In response to S1,F13 (Part of the Communication Model)
S1,F18 - In response to S1,F17 (only from Host OFF-LINE state)
S9,Fx - In response to a receive error in S1,F13 or S1,F17
Sx,F0 - In response to all other primary messages (including S1,F17 if not in Host OFF-LINE state)
Equipment will discard all replies except S1,F14 and S1,F2.

3.3. EQUIPMENT PROCESSING STATE

This model describes the typical operation of the wire bonder with the emphasis on SECS implementation issues. The behavior of the wire bonder in the performance of its function at any time can be classified under one of the following states. As the Eagle Xtreme Gold Wire Bonder is using menu driven control, entering each menu can cause a transition to a state corresponding to the function of the items in that menu. The wire bonder will generate collection events for each processing state transition.

3.3.1. INIT

Init state is not an actual processing state. It is shown here just to indicate that upon completion of wire bonder internal initialization, the IDLE state is entered. No communication is allowed before the initialization completed, as the wire bonder is setting the default values for the equipment constants and variables.

3.3.2. IDLE

In this state, the wire bonder is waiting for instructions. After going through the initialization procedure and success in establishing communication with the host computer, the wire bonder is ready to accept the next command. Besides initialization, this state can be re-entered after changing the process parameters, the completion of the processing cycle, or receipt of a STOP command when doing continuous bonding.

3.3.3. SETUP

Entering this state means the operator is trying to setup the process variables or changing the control parameters. The wire bonder is in this state when the operator enters one of these menus:
[Program Menu], [Configuration Menu], [Maintenance Menu], [Utility Menu], and [System Menu].

3.3.4. EXECUTING

This is the state when the wire bonder is executing a process program automatically after receiving a start command from the operator, and will remain in this state until a STOP command is received or the process is completed. The occurrence of an event requiring external assistance will result in a transition to alarm state, where an alarm report will be sent if it is enabled and the wire bonder is in communicating state.

EXECUTING – LOCAL

Processing and remote control commands from the host are restricted. The host is unable to initiate a change of state. This state is entered if the START key I pressed when the wire bonder is in IDLE mode.

EXECUTING – REMOTE

Processing is under host control. However, the operator will have the ability to override the host control. The STOP command, and REMOTE / LOCAL selection is always available.

3.3.5. PAUSE

In this state, processing is suspended and the equipment is awaiting a command. Transition to pause state may be operator initiated, i.e., when key '1' (Single Bond) is pressed, or host initiated at via the STOP command, when the wire bonder is doing auto bonding. Upon receipt of a START command from the host or when key '0' (Auto Bond) is processed, the wire bonder will request the operator to resume to the executing state.

3.3.6. ALARM

Any event that may endanger people, equipment or material being processed will result in transition to the alarm state and the generation of an appropriate alarm. Processing will be suspended until the necessary actions have resulted in the alarm being cleared.

3.3.7. UPLOAD

The process program in AB339E Gold Wire Bonder refers to the entire set of process parameters and control parameters. The process programs transferred between the wire bonder and the host computer are unformatted process program. The UPLOAD action refers to transferring the process program from the wire bonder to the host computer, while the DOWNLOAD action means the opposite direction.

Uploading process programs can be initiated by either the operator at the wire bonder or by the Host system.

The process state will become UPLOAD only if the operator initiates the upload.

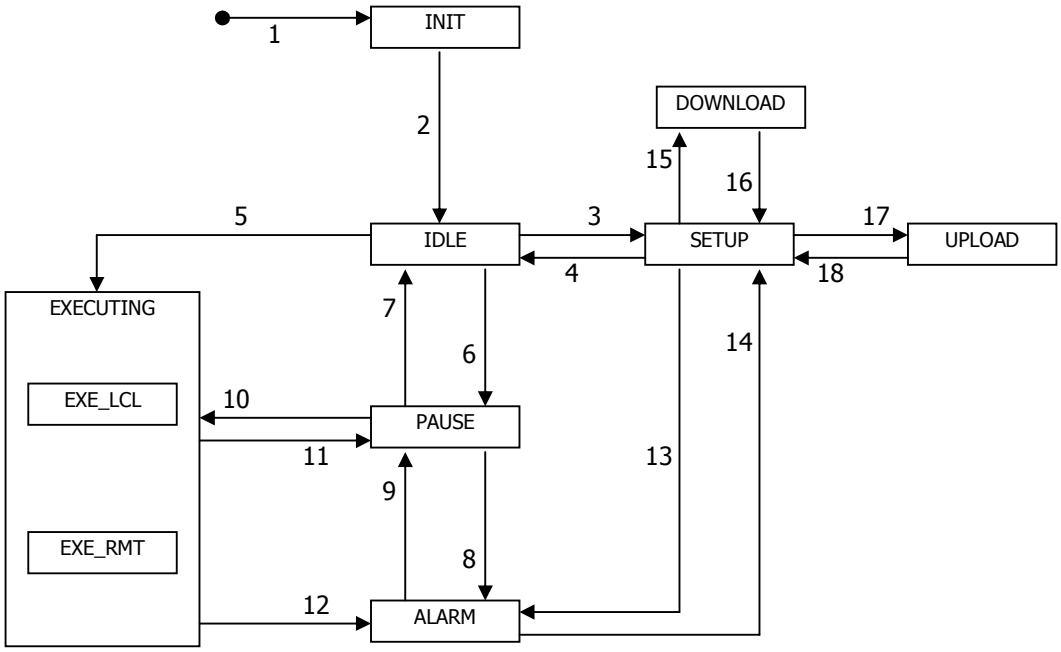
3.3.8. DOWNLOAD

The program download operation can be initiated by either the operator or the Host computer. The process state will become DOWNLOAD only if the operator initiates the download.

With this method, the Host can send multiple programs to the wire bonder before the current process lot is completed. If the machine is bonding during the download process, the bonding may pause for a few seconds when a new process program is received, as the data must be decompressed and stored to hard disk. The bonding quality is not affected at any time. It has been taken into consideration that the operator is the person who knows when will the current lot completed, and therefore he should have the final decision to switch to the next process program.

If the operator initiated the download, then the process program will automatically be loaded into memory when it is received. The DOWNLOAD state will remain until the process program is fully loaded into memory from its storage space.

ASMGEM Processing State Diagram



ASMGEM Processing State Table

#	CURRENT STATE	TRIGGER	NEW STATE	ACTION	COMMENT
1	(Entry to INIT)	System Initialization	INIT	None	No communication at this time as the system is still powering up
2	INIT	Initialization complete	IDLE	The tower lamp becomes YELLOW	Able to accept Remote commands and unformatted process programs if such options enabled.
3	IDLE	User navigates to one of the following menus – Setup - Teach - Wire Parameters - Parameter - WH Menu - WH Setup	SETUP	None	User has indicated that some parameters need adjustment or a new device is to be set up
4	SETUP	User finishes adjusting parameters or setting up a new device	IDLE	None	Returning to the Main menu or jumping to the Auto Bond menu
5	IDLE	User selects Start Auto bond from the menu and a valid process program is in memory	EXECUTING (EXE_LCL if Remote Control is DISABLED otherwise EXE_RMT)	Machine starts performing the bonding of devices. The tower lamp becomes GREEN	The operator has taught or loaded a valid process program and adjusted the parameters to a stage where the machine is ready to bond the device.
6	IDLE	User selects Start Single bond from the menu and a valid process program is in memory	PAUSE	Machine will perform any necessary PR, stop at the next wire to be bonded and show the Single Bond menu	The operator has taught or loaded a valid process program and adjusted the parameters to a stage where the machine is ready to bond the device
7	PAUSE	User quits from the Single Bond menu or the Host sends a remote command to exit from bonding	IDLE	None	The operator or the Host has cancelled any bonding operation
8	PAUSE	User attempts to start bonding but the machine detects an error and raises an alarm	ALARM	Alarm is set and the tower lamp becomes RED	The alarm message will display on the screen and wait for the operator to press a key.
9	ALARM	User responds to the alarm signal and presses a key	PAUSE	Alarm is cleared and the tower lamp becomes YELLOW	The alarm message is cleared and the operator is free to correct the cause of the alarm
10	PAUSE	User selects Start Auto Bond from the Single Bond menu	EXECUTING (EXE_LCL if Remote Control is DISABLED otherwise EXE_RMT)	Machine continues the bonding cycle and the tower lamp becomes GREEN	Operator has verified the bonding quality and now continues to bond devices
11	EXECUTING	User presses the Single Bond button, or the machine stops automatically because of an enabled Last LF or Pause function, or Host sent a relevant Remote Command	PAUSE	Single Bond menu is displayed and the tower light becomes YELLOW	Usually performed by the operator while checking on the quality of the bonding manually

ASMGEM Processing State Table Continued

#	CURRENT STATE	TRIGGER	NEW STATE	ACTION	COMMENT
12	EXECUTING	During the bonding of the device an error is encountered eg. PR Failure	ALARM	Alarm is set and the tower lamp becomes RED	The alarm message will display on the screen and wait for the operator to press a key.
13	SETUP	During the adjustment an error was encountered	ALARM	Alarm is set and the tower lamp becomes RED	The alarm message will display on the screen and wait for the operator to press a key.
14	ALARM	Operator responds to the error generated during any setup operations	SETUP	Alarm is cleared and the tower lamp becomes YELLOW	The alarm message is cleared and the operator is free to correct the cause of the alarm
15	SETUP	Operator selects Load from Host	DNLOAD	Asks the user to key in the name to get from the host	If the user inputs the name, then the file will be requested to be downloaded from the Host. A message is displayed on the screen informing the operator that a file is being downloaded
16	DNLOAD	File downloaded from the Host or the user cancelled the operation	SETUP	None	If the download is complete, then the operator will be asked whether to load the process program. If the operator answers affirmative, then the process program is loaded into memory before the state changes to SETUP
17	SETUP	Operator selects Save To Host or Copy Pgm To Host	UPLOAD	Asks the user to key in a filename to save or select an existing file from the file list	If the user selects the filename then the file will be sent to the Host in ".WB" format. A message is displayed on the screen informing the operator that the file is being uploaded
18	UPLOAD	File uploaded to Host or the user cancelled the operation	SETUP	None	

4. EQUIPMENT CAPABILITIES

This section describes the wire bonder interface capabilities. These interface operations are initiated through the communications interface using SECS-II messages.

The ASM Eagle Xtreme Gold Wire Bonder supports the following capabilities:

- 4.1 Establish Communications
- 4.2 On-line Identification
- 4.3 Data Collection
  - 4.3.1 Status Data Collection
  - 4.3.2 Variable Data Collection
  - 4.3.3 Event Data Collection
- 4.4 Dynamic Event Report Configuration
- 4.5 Alarm Management
- 4.6 Remote Control
- 4.7 Equipment Constants
- 4.8 Process Program Management
- 4.9 Error Messages
- 4.10 Clock
- 4.11 Terminal Services
- 4.12 Diagnostics

4.1. ESTABLISH COMMUNICATIONS

The establish communications capability provides a means of formally establishing communications following power-up or any loss of communications between communicating partners. Communications between host and wire bonder are formally established through the use of the Establish Communications Request / Establish Communications Acknowledge Transaction (S1F13/S1F14).

In conjunction with the Communications State Model, when the wire bonder is in NOT COMMUNICATING state, it will disregard all host messages except the Establish Communication Request and Establish Communications Acknowledge Transaction. It will also attempt to establish communications by periodically sending the Establish Communications Request until the communications have been formally established. The host may also try establishing communications with the wire bonder at any time when the communications state is ENABLED.

4.1.1. Establish Communications Messages

S1, F13 Establish Communications Request (CR) **S, H ⇔ E, reply**

Description : Provides a formal means for the initialization of the communication, either upon power up or to re-establish after a break in communications. Ie. This should always be the first message that is sent in setting up the communications link. Please refer to the Communications State Model for details on the repeat interval and state transitions required before the equipment is said to be in the COMMUNICATING state.

Structure : L, 2
1. <A MDLN>
2. <A SOFTREV>

Exception : The host sends a zero-length list to the equipment.

Note :

MDLN ASCII string with a maximum length of 6 bytes used to hold the machine model number. For the case of the Eagle Xtreme wire bonder it will always be "Eg XT"

SOFTREV ASCII string with a maximum length of 6 bytes used to hold the version number of the software running on the machine. This will always return the PC version number. Eg. If the machines PC software version is 09.05.92, then the data stored in this variable will be "090592"

S1, F14 Establish Communications Request Acknowledge (CRA) **S , H ⇔ E**

Description : Accepts or denies the request from the originator to set up the communications link.

Structure : L,2
1. <B COMMACK>
2. L,2
    1. <A MDLN>
    2. <A SOFTREV>

Exception : The host sends a zero-length list for item 2 to the equipment. (ie. Doesn't send MDLN or SOFTREV)

Note :

COMMACK 1 byte binary informing whether establish communications is allowed at this time.
0 = Accepted
1 = Denied, Try Again
2~63 Reserved

MDLN and SOFTREV are only valid if COMMACK = 0

MDLN ASCII string with a maximum length of 6 bytes used to hold the machine model number. For the case of the Eagle Xtreme wire bonder it will always be "EAGLE"

SOFTREV ASCII string with a maximum length of 6 bytes used to hold the version number of the software running on the machine. This will always return the PC version number. Eg. If the machines PC software version is 09.05.92, then the data stored in this variable will be "090592"

4.1.2. Establish Communications Scenarios

Host Attempts to Establish Communications

COMMENT	HOST	EQUIPMENT	COMMENT
		Communications State is ENABLED	
Establish Communications Request (CR)			
	S1,F13 →		
		Establish Communications Request Acknowledge (CRA) ← S1,F14 Reply COMMACK = Accept (0) Communications State = COMMUNICATING	

Host Attempts to Establish Communications

COMMENT	HOST	EQUIPMENT	COMMENT
		Communications State = NOT COMMUNICATING [LOOP]	
		[LOOP] – SEND Establish Communications Request (CR) ← S1,F13	
Establish Communications Request Acknowledge (CRA)			
	S1,F14 →		
		[IF] S1,F14 received without timeout [THEN] exit_loop – SEND [ELSE] Delay for Establish Communications Timeout [ENDIF]	
		[END_LOOP] – SEND [IF] COMMACK = Accept (0) [THEN] Communications State = COMMUNICATING exit_loop [ELSE] Reset timer for delay, and delay for Establish Communications Timeout [ENDIF]	
		[END_LOOP]	

If the case arises where the Equipment and the Host both send the S1,F13 at the same time, please refer to the Communication State model earlier in this manual for details on the response.

4.2. ON-LINE IDENTIFICATION

This capability allows the host to verify the presence and identify of the wire bonder. The wire bonder will respond with the equipment model and software revision included in the reply message.

4.2.1. ON-LINE Identification Messages

S1, F1 Are You There Request I **S, H ⇔ E, reply**

Description : Establishes whether the equipment is ON-LINE. If the equipment responds with an S1, F0, it means that the equipment is OFF-LINE. If the Host responds with an S1, F0, the equipment will treat the message as a communication timeout.

Structure : Header only

S1, F2 On Line Data (D) **S , H ⇔ E**

Description : Indicates that the equipment or the Host is ON-LINE (alive).

Structure : L,2
1. <A MDLN>
2. <A SOFTREV>

Exception : The host sends a zero-length list to the equipment. (ie. Doesn't send MDLN or SOFTREV)

Note :

MDLN ASCII string with a maximum length of 6 bytes used to hold the machine model number. For the case of the Eagle Xtreme wire bonder it will always be "Eg XT"

SOFTREV ASCII string with a maximum length of 6 bytes used to hold the version number of the software running on the machine. This will always return the PC version number. Eg. If the machines PC software version is 09.05.92, then the data stored in this variable will be "090592"

S1, F15 Request OFF-LINE (ROFL) **S, H → E, reply**

Description : Sent by the Host when it wants the equipment to transition to the OFF-LINE state.

Structure : Header only

S1, F16 OFF-LINE Acknowledge (OFLA) **S , H ← E**

Description : Replies to the Host acknowledge or error.

Structure : <B OFLACK>

Note :
OFLACK Informs whether the equipment went to OFF-LINE state
0 = OFF-LINE Acknowledge
1~63 Reserved

S1, F17 Request ON-LINE (RONL) **S, H → E, reply**

Description : Sent by the Host when it wants the equipment to transition to the ON-LINE state.

Structure : Header only

S1, F18 ON-LINE Acknowledge (ONLA) **S , H ← E**

Description : Replies to the Host acknowledge or error.

Structure : <B ONLACK>

Note :
OFLACK Informs whether the equipment went to ON-LINE state
0 = ON-LINE Acknowledge
1 = ON-LINE Not Allowed
2 = Equipment Already ON-LINE
3~63 Reserved

4.2.2. ON-LINE Identification Scenarios

Host Checks Equipment Is ON-LINE

COMMENT	HOST	EQUIPMENT	COMMENT
Are You There Request I			
	S1,F1 →		
		On Line Data (D) informing MDLN and SOFTREV ← S1,F2	

Host Requests OFF-LINE

COMMENT	HOST	EQUIPMENT	COMMENT
Request OFF-LINE (ROFL)			
	S1,F15 →		
		[IF] Equipment is OFF-LINE [THEN] Equipment does not process requests ← S1, F0 [ELSE] Equipment acknowledges and goes OFF-LINE ← S1, F16 "Equipment OFF-LINE" event ← S6, F11	
Event Report Acknowledge (ERA)			
	S6, F12 →		
		[END_IF]	

Host Sets ON-LINE

COMMENT	HOST	EQUIPMENT	COMMENT
Request ON-LINE (RONL)			
	S1, F17 →		
		[IF] Equipment is not in HOST OFF-LINE control state [THEN] Equipment denies request (ONLACK = 1) ← S1, F18 [ELSE] Equipment acknowledges request (ONLACK = 0) ← S1, F18 "Control State LOCAL / REMOTE" event ← S6, F11	
Event Report Acknowledge (ERA)			
	S6, F12 →		
		[END_IF]	

4.3. DATA COLLECTION

Data collection allows the host to monitor wire bonder activity via event reporting and query of selected status variables.

STATUS DATA COLLECTION
This capability allows the host to query the wire bonder for selected status information by specifying the corresponding SVIDs. Upon a request from the host, the wire bonder sends the host the value of the selected status variables. The host may also request the description (name and units) of any or all available status variables.

VARIABLE DATA COLLECTION
This capability allows the host to query for the wire bonder data variables. The host may request a report containing data variables from the equipment by specifying the RPTID. It is assumed that the report has been previously defined, i.e., using the Define Report S2F33 transaction.

EVENT DATA COLLECTION
This capability allows the wire bonder to notify the host upon the occurrence of an equipment collection event and of any change in the equipment state. The event-based approach to data collection provides automatic notification to the host of equipment activities. A specific event report containing a set of predefined variables will be generated upon the occurrence of any event. Each collection event has unique identifier (CEID). The corresponding event report can consist of a list of report identified by RPTIDs and the variables attached to each report and are identified by SVIDs and/or EVIDs.

The wire bonder also responds to a host request for specific information from a specified event report.

4.3.1. Data Collection Messages

S1, F3 Selected Equipment Status Request (SSR) **S, H → E, reply**

Description : Host requests the equipment to report selected values of its status.

Structure : L, n {n = # of SVIDs following}
2. <U2 $SVID_1$>
.
.
n. <U2 $SVID_n$)

Exception : A zero length list means the Host is requesting the equipment to report ALL SVIDs.

S1, F4 Selected Equipment Status Data (SSD) **M , H ← E**

Description : Equipment reports the value of each SVID in the order requested by S1, F3.

Structure : L, n {n = # of SVIDs following}
3. <V SV_1>
.
.
n. <V SV_n)

Exception : A zero length returned for SV_i means that $SVID_i$ does not exist.

S1, F11 Status Variable Namelist Request (SVNR) **S, H → E, reply**

Description : Host requests the equipment to identify certain status variables.

Structure : L, n {n = # of SVIDs following}
4. <U2 $SVID_1$>
.
.
n. <U2 $SVID_n$)

Exception : A zero length list means the Host is requesting the equipment to report ALL SVIDs.

S1, F12 Status Variables Namelist Reply (SVNRR) **M , H ← E**

Description : Equipment reports the name and units of each SVID in the order requested.

Structure : L, n {n = # of SVIDs following}
1. L, 3
    1. <U2 $SVID_1$>
    2. <A $SVNAME_1$>
    3. <A $UNITS_1$>
2. L, 3
.
.
n. L, 3
    1. <U2 $SVID_n$>
    2. <A $SVNAME_n$>
    3. <A $UNITS_n$>

Note : Empty strings will be returned for $SVNAME_i$ and $UNITS_i$ for an $SVID_i$ that does not exist.

S6, F15 Event Report Request (ERR) **S, H → E, reply**

Description : Host requests the equipment to send a given report group.

Structure : <U2 CEID>

S6, F16 Event Report Data (ERD) **M , H ← E**

Description : Equipment sends reports linked to a given CEID.

Structure : L, 3

1. <U1 DATAID>
2. <U2 CEID>
3. L, a {a = # of linked reports}
    1. L, 2
        1. <U2 $RPTID_1$>
        2. L, b {b = # of variables in report}
            1. <V V_1>
            .
            .
            b. <V V_b>
    .
    .
    a. L, 2
        1. <U2 $RPTID_a$>
        2. L, c {c = # of variables in report}
            1. <V V_1>
            .
            .
            c. <V V_c>

Exceptions : If there are no reports linked to the event, a 'null' report is assumed. A zero-length list for # of reports means there are no reports linked to the given CEID.

S6, F19 Individual Report Request (IRR) **S, H → E, reply**

Description : Host requests a defined report from the equipment

Structure : <U2 RPTID>

S6, F20 Individual Report Data (IRD) **M, H ← E**

Description : Variable data defined for the given RPTID is sent to the host.

Structure : L, n {n = # of variable data items}
5. <V V_1>
.
.
n. <V V_n>

Exceptions : A zero length list means RPTID is not defined

S6, F5 Multi-block Data Send Inquire (MBI) **S, H ← E, reply**

Description : If S6, F11 is multi-block, then the equipment will first request from the Host permission to proceed to send it.

Structure : L, 2
1. <U1 DATAID>
2. <U2 DATALENGTH>

S6, F6 Multi-block Grant (MBG) **S , H → E**

Description : Host grant status to equipment to send a multi-block report

Structure : <B GRANT6>

Note :
GRANT6 Represents the permission to send granted by the Host
0 = Permission granted
1 = Busy, try again
2 = Not interested
>2 = Other errors
3~63 Reserved

S6, F11 Event Report Send (ERS) **M , H ← E, reply**

Description : Equipment sends a defined, linked, and enabled group of reports to the host upon the occurrence of an event (CEID).
If the data will require the sending of a multi-block message, this message will be proceeded by the S6, F5 / S6, F6 Inquire / Grant transaction.

Structure : L, 3

1. <U1 DATAID>
2. <U2 CEID>
3. L, a {a = # of linked reports}
    1. L, 2
        1. <U2 $RPTID_1$>
        2. L, b {b = # of variables in report}
            1. <V V_1>
            .
            .
            b. <V V_b>
    .
    .
    a. L, 2
        1. <U2 $RPTID_a$>
        2. L, c {c = # of variables in report}
            1. <V V_1>
            .
            .
            c. <V V_c>

Exceptions : If there are no reports linked to the event, a 'null' report is assumed. A zero-length list for # of reports means there are no reports linked to the given CEID.

S6, F12 Event Report Acknowledge (ERA) **S, H → E**

Description : Host acknowledges whether it was able to accept the report sent.

Structure : <B ACKC6>

Note :

ACKC6 Indicates whether the Host was able to accept the report.
0 = Accepted
>0 = Error, not accepted
1~63 Reserved

4.3.2. Data Collection Scenarios

Host Requests Equipment Status Report

COMMENT	HOST	EQUIPMENT	COMMENT
Selected Equipment Status Request (SSR) with specified SVIDs			
	S1, F3 →		
		Selected Equipment Status Data (SSD) ← S1, F4	

Host Requests Equipment Status Variable Name List

COMMENT	HOST	EQUIPMENT	COMMENT
Status Variable Namelist Request (SVNR) with specified SVIDs			
	S1, F11 →		
		Status Variable Namelist Reply (SVNRR) ← S1, F12	

Host Requests Event Report

COMMENT	HOST	EQUIPMENT	COMMENT
Event Report Request (ERR)			
	S6, F15 →		
		Event Report Data (ERD) ← S6, F16	

Collection Event Occurs on the Equipment

COMMENT	HOST	EQUIPMENT	COMMENT
		[IF] Event Report will be Multi-block [THEN] send Multi-block Data Send Inquire (MBI) ← S6, F5	
Multi-block Grant (MBG)			
	S6, F6 →		
		[ENDIF] Event Report Send (ERS) ← S6, F11	
Event Report Acknowledge (ERA)			
	S6, F12 →		

4.4. DYNAMIC EVENT REPORT CONFIGURATION

This capability allows the host to increase or decrease the data flow according to need. The wire bonder supports the following event report configuration functionality through the SECS-II interface :

- Host definition / deletion of custom report
- Host linking / unlinking of defined reports to specified collection event
- Host enabling / disabling the reporting of specified collection events

The wire bonder can be instructed by the host to enable or disable reporting of collection events on an individual or collective basis. Reports may be attached to an event report message. These reports are specifically linked to the desired event and typically contain variable data relating to that event.

The host is able to create reports and link them to events via the SECS-II interface.
The basic steps are:

1. create reports with S2, F33,
2. link the reports to an event with S2, F35, and
3. enable event reporting with S2, F37.

Note: All details of the reports generated, linked and events enabled, are stored to non-volatile memory. Thus if the machine is turned off and then on, the setting by the Host will still be valid. This is not the case though if the machine undergoes a COLD START or has a SOFTWARE UPGRADE.

4.4.1. Dynamic Event Report Configuration Messages

S2, F33 Define Report (DR) **M , H → E, reply**

Description : Allows the Host to define a group of reports for the equipment.
If the data will require the sending of a multi-block message, this message will be processed by the S2, F39 / S2, F40 Inquire / Grant transaction.

Structure : L, 2

1. <U1 DATAID>
2. L, a {a = # of reports to define}
   1. L, 2
      1. <U2 $RPTID_1$>
      2. L, b {b = # of variables in report}
         1. <U2 VID_1>
         .
         .
         b. <U2 VID_b>
   .
   .
   a. L, 2
      1. <U2 $RPTID_a$>
      2. L, c {c = # of variables in report}
         1. <U2 VID_1>
         .
         .
         c. <U2 VID_c>

Exceptions : A list of zero length following DATAID deletes all report definitions and associated links.
A list of zero length following RPTID, deletes report type RPTID. All CEID links to this RPTID are also deleted.

S2, F34 Define Report Acknowledge (DRA) **S, H ← E**

Description : Acknowledge or error. If an error condition is detected, the entire message is rejected and no change is made to any reports on the equipment.

Structure : <B DRACK>

Note :

DRACK Indicates whether the Equipment was able to generate the reports.
0 = Accepted
1 = Denied. Insufficient space
2 = Denied. Invalid format
3 = Denied. At least one RPTID already defined
4 = Denied. At least one VID does not exist
>4 = Other Error, not accepted
1~63 Reserved

S2, F35 Link Event Report (LER) **M , H → E, reply**

Description : Allows the Host to link a group of defined reports to an event (CEID). Once linked the event is by default DISABLED.
If the data will require the sending of a multi-block message, this message will be processed by the S2, F39 / S2, F40 Inquire / Grant transaction.

Structure : L, 2

1. <U1 DATAID>
2. L, a {a = # of events }
    1. L, 2
        1. <U2 $CEID_1$>
        2. L, b {b = # of reports linked to $CEID_1$}
            1. <U2 $RPTID_1$>
            .
            .
            b. <U2 $RPTID_b$>
    .
    .
    a. L, 2
        1. <U2 $CEID_a$>
        2. L, c {c = # of reports linked to $CEID_a$}
            1. <U2 $RPTID_1$>
            .
            .
            c. <U2 $RPTID_c$>

Exceptions : A list of zero length following CEID deletes all report links to that event.

S2, F36 Link Event Report Acknowledge(LERA) **S, H ← E**

Description : Acknowledge or error. If an error condition is detected, the entire message is rejected and no change is made to any events on the equipment.

Structure : <B LRACK>

Note :

LRACK Indicates whether the Equipment was able to link the reports to the events.
0 = Accepted
1 = Denied. Insufficient space
2 = Denied. Invalid format
3 = Denied. At least one CEID link already defined
4 = Denied. At least one CEID does not exists
5 = Denied. At least one RPTID does not exists
>5 = Other Error, not accepted
5~63 Reserved

S2, F37 Enable / Disable Event Report (EDER) **S , H → E, reply**

Description : Allows the Host to enable or disable the reporting for a group of collection events (CEIDs).

Structure : L, 2
1. <BOOLEAN CEED> {enable / disable}
2. L, n {n = # of events}
   1. <U2 $CEID_1$>
   .
   .
   n. <U2 $CEID_n$>

Exceptions : A list of zero length following CEED means enable / disable all CEIDs that exist on the equipment.

S2, F38 Enable / Disable Event Report Acknowledge (EERA) **S, H ← E**

Description : Acknowledge or error. If an error condition is detected, the entire message is rejected and no change is made to any events on the equipment.

Structure : <B ERACK>

Note :
ERACK Indicates whether the Equipment was able to link the reports to the events.
0 = Accepted
1 = Denied. At least one CEID does not exists
>1 = Other Error, not accepted
2~63 Reserved

S2, F39 Multi-block Inquire (DMBI) **S , H → E, reply**

Description : If S2, F33 or S2, F35 are multi-block, they must be proceeded with this transaction.

Structure : L, 2
1. <U1 DATAID>
2. <U2 DATALENGTH>

S2, F40 Multi-block Grant (DMBG) **S, H ← E**

Description : Grant permission to send multi-block message

Structure : <B GRANT>

Note :
GRANT Permission code
0 = Permission Granted
1 = Busy, Try Again
2 = No Space Available
3 = Duplicate DATAID
>3 = Equipment Specific Error Code
4~63 Reserved

4.4.2. Dynamic Event Report Configuration Scenarios

Host Sets up Collection Event Reporting

COMMENT	HOST	EQUIPMENT	COMMENT
[IF] Define Report is Multi-block [THEN] send Multi-block Inquire (DMBI)			
	S2, F39 →		
		Multi-block Grant (DMBG) ← S2, F40	
[END_IF] Send Define Report (DR) with report definitions			
	S2, F33 →		
		Data received and reports generated. Store details in non-volatile storage. Send Define Report Acknowledge (DRA) with DRACK = 0 ← S2, F34	
[IF] Link Event / Report is Multi-block [THEN] send Multi-block Inquire (DMBI)			
	S2, F39 →		
		Multi-block Grant (DMBG) ← S2, F40	
[END_IF] Send Link Event Report (LER) with Event and Report details			
	S2, F35 →		
		Link made between the CEID and the RPTIDs. Store details in non-volatile storage. Send Link Event Report Acknowledge (LERA) with LRACK = 0 ← S2, F36	
Enable CEID so that reports can be automatically generated by the Equipment. Send Enable / Disable Event Report (EDER)			
	S2, F37 →		
		Send Enable / Disable Event Report Acknowledge (EERA) with ERACK = 0. Store details in non-volatile storage of which CEIDs are enabled. ← S2, F38	

4.5. ALARM MANAGEMENT

The alarm management capability provides for host notification and management of alarm conditions occurring on the wire bonder, through the Alarm Report Send / Acknowledge transaction (S5F1/S5F2). Each alarm defined in the wire bonder has a unique alarm identifier, ALID. Section 11 lists the ALID, alarm text and category code, for each alarm defined. Besides, the alarm management capability also provides mechanisms for :

- Uploading a list of alarm texts
- Enabling and disabling the notification of specific alarms
- Host query of alarms set and enabled status on the wire bonder

On the Eagle Xtreme the clearing of an alarm occurs when the operator presses a key in response to reading the alarm message on the screen, and not for the actual correction of the alarm.

4.5.1. Alarm Management Messages

S5, F1 Alarm Report Send (ARS) **S , H ← E, reply**

Description : Equipment uses this message to report a change in or presence of an alarm condition. This message will be sent when the alarm is set, and again when the alarm is cleared.

Structure : L, 3
1. <B ALCD>
2. <U2 ALID>
3. <A[40] ALTX>

Note :

ALCD	Alarm code byte bit 8 = 1 means alarm set bit 8 = 0 means alarm cleared bit 7-1 is alarm category (not used in GEM)
ALID	Alarm Identification Code
ALTX	Alarm text description

S5, F2 Alarm Report Acknowledge (ARA) **S, H → E**

Description : Acknowledge or error

Structure : <B ACKC5>

Note :

ACKC5	Informs whether the alarm information was accepted 0 = Accepted >0 = Error, not accepted 1~63 Reserved

S5, F3 Enable / Disable Alarm Send (EAS) **S , H → E, reply**

Description : Changes the state of the enable bit in the equipment. The enable bit determines if the alarm will be sent to the host.

Structure : L, 2
1. <B ALED>
2. <U2 ALID>

Exception : A zero length item for ALID means all alarms

Note :

ALED	Alarm code byte bit 8 = 1 means enable alarm bit 8 = 0 means disable alarm bit 7-1 is not used in GEM
ALID	Alarm Identification Code

S5, F4 Enable / Disable Alarm Acknowledge (EAA) **S, H ← E**

Description : Acknowledge or error

Structure : <B ACKC5>

Note :

ACKC5	Informs whether the alarm information was accepted 0 = Accepted >0 = Error, not accepted 1~63 Reserved

S5, F5 List Alarms Request (LAR) **S , H → E, reply**

Description : Requests the equipment to send alarm information to the host

Structure : <U2 $ALID_1$, . . . ,U2 $ALID_n$>

Exception : A zero length item means send all possible alarms regardless of the state of ALED.

Note :
ALID Alarm Identification Code

S5, F6 List Alarm Data (LAD) **M, H ← E**

Description : Contains the alarm data known to the equipment.

Structure : L, m {m = # of alarms}
1. L, 3
    1. <B $ALCD_1$>
    2. <U2 $ALID_1$>
    3. <A[40] $ALTX_1$>
2. L, 3
.
.
m. L, 3
    1. <B $ALCD_m$>
    2. <U2 $ALID_m$>
    3. <A[40] $ALTX_m$>

Exception : If m=0, no response can be made. A zero length item returned for $ALCD_i$ or $ALTX_i$ means that alarm (ALID) does not exist.

Note :
ALCD Alarm code byte
bit 8 = 1 means alarm set
bit 8 = 0 means alarm cleared
bit 7-1 is alarm category (not used in GEM)
ALID Alarm Identification Code
ALTX Alarm text description

S5, F7 List Enabled Alarm Request (LEAR) **S , H → E, reply**

Description : Requests the equipment to send information on the enabled alarms

Structure : Header only

S5, F8 List Enabled Alarm Data (LEAD) **M, H ← E**

Description : Contains the alarm data known to the equipment, only for those alarms that are currently enabled.

Structure : L, m {m = # of enabled alarms}
1. L, 3
    1. <B ALCD$_1$>
    2. <U2 ALID$_1$>
    3. <A[40] ALTX$_1$>
2. L, 3
.
.
m. L, 3
    1. <B ALCD$_m$>
    2. <U2 ALID$_m$>
    3. <A[40] ALTX$_m$>

Note :

ALCD	Alarm code byte bit 8 = 1 means alarm set bit 8 = 0 means alarm cleared bit 7-1 is alarm category (not used in GEM)
ALID	Alarm Identification Code
ALTX	Alarm text description

4.5.2. Alarm Management Scenarios

Enable / Disable Alarms

COMMENT	HOST	EQUIPMENT	COMMENT
Enable / Disable Alarm Send (EAS)			
	S5, F3 →		
		Enable / Disable Alarm Acknowledge (EAA)	
		← S5, F4	

Upload Alarm Information

COMMENT	HOST	EQUIPMENT	COMMENT
List Alarms Request (LAR) or			
List Enabled Alarm Request (LEAR)			
	S5, F5 or S5, F7 →		
		List Alarm Data (LAD) or	
		List Alarm Data (LEAD)	
		← S5, F6 or S5, F8	

Send Alarm Report

COMMENT	HOST	EQUIPMENT	COMMENT
		Alarm occurrence detected by the equipment.	
		[IF] Alarm report enabled	
		[THEN] Alarm Report Send (ARS)	
		← S5, F1	
Alarm Report Acknowledge (ARA)			
	S5, F2 →		
		[END_IF]	
		[IF] Alarm event enabled	
		[THEN] Event Report Send (ERS)	
		← S6, F11	
Event Report Acknowledge (ERA)			
	S6, F12 →		
		[END_IF]	

4.6. REMOTE CONTROL

This capability provides the host with a level of control over wire bonder operations. It allows the wire bonder to respond to host initiated commands which control the wire bonder operations while in the REMOTE state. The Remote Command Send (S2F21) is utilized to allow the host to send a remote command, specified by RCMD, to the wire bonder. The successful transaction a remote command will result in an action that generates a collection event upon completion.

4.6.1. Remote Control Messages

S2, F21 Remote Command Send (RCS) **S , H → E, reply**

Description : Causes the equipment to perform some predefined activity.

Structure : <U1 RCMD>

Note :
RCMD Remote command code

S2, F22 Remote Command Acknowledge (RCA) **S, H ← E**

Description : Acknowledge or error

Structure : <U1 CMDA>

Note :
CMDA Remote command acknowledge code
0 = Completed or done
1 = Command does not exist
2 = Can not perform now
>2 = Other equipment specific error
3~63 Reserved

S2, F41 Host Command Send (HCS) **S , H → E, reply**

Description : Causes the equipment to perform some predefined activity using the data in the associated parameters.

Structure : L, 2

1. <A[20] RCMD>
2. L, n {n = # of parameters}
    1. L, 2
        1. <A[20] $CPNAME_1$>
        2. <V $CPVAL_1$>
    .
    .
    n. L, 2
        1. <A[20] $CPNAME_n$>
        2. <V $CPVAL_n$>

Note :

RCMD	Remote command text
CPNAME	Parameter text
CPVAL	Parameter data

S2, F42 Host Command Acknowledge (HCA) **S, H ← E**

Description : Acknowledge or error. If the command is not accepted due to one or more invalid parameters (ie. HCACK=3), then a list of the invalid parameters will be returned containing the parameter name and reason for being invalid.

Structure : L, 2

1. <B HCACK>
2. L, n {n = # of parameters}
    1. L, 2
        1. <A[20] $CPNAME_1$>
        2. <B $CPACK_1$>
    .
    .
    n. L, 2
        1. <A[20] $CPNAME_n$>
        2. <B $CPACK_n$>

Note :

HCACK Host Command Parameter Acknowledge Code
0 = Acknowledge, command has been performed
1 = Command does not exists
2 = Can not perform now
3 = At least one parameter is invalid
4 = Acknowledge, command will be performed with completion signaled later by an event
5 = Rejected, Already in the desired condition
6 = No such object exists
7~63 Reserved

CPNAME Parameter text

CPACK Command Parameter Acknowledge Code
1 = Parameter Name (CPNAME) does not exist
2 = Illegal Value specified for CPVAL
3 = Illegal Format specified for CPVAL
>3 = Other equipment specific error
3~63 Reserved

4.6.2. Remote Control Scenarios

Remote Command Send

COMMENT	HOST	EQUIPMENT	COMMENT
Remote Command Send (RCS)			
	S2, F21 →		
		Remote Command Acknowledge (RCA) ← S2, F22	

Host Command Send

COMMENT	HOST	EQUIPMENT	COMMENT
Host Command Send (HCS)			
	S2, F41 →		
		Host Command Acknowledge (HCA) ← S2, F42 [IF] Command Accepted (HCACK = 0 or 4) [THEN] Event Report Send (ERS)any State Change ← S6, F11	
Event Report Acknowledge (ERA)			
	S6, F12 →		
		[END_IF]	

4.7. EQUIPMENT CONSTANT

This capability provides a method for the host to read and to change the value of selected equipment constants on the wire bonder. The following operations are supported :

- Allows the host to change the value of one or more equipment constant
- Allows the host to determine the current value of equipment constants
- Allows the host to retrieve basic information about the equipment constants available at the equipment

4.7.1. Equipment Constant Messages

S2, F13 Equipment Constant Request (ECR) **S , H → E, reply**

Description : Host requests the equipment send data on the equipment constant variables.

Structure : L, n {n = # of ECIDs}
6. <U2 $ECID_1$>
.
.
n. <U2 $ECID_n$>

Exception : A zero length list means report all ECV's according to the predefined order.

S2, F14 Equipment Constant Data (ECD) **M, H ← E**

Description : Replies all the EC data in the order requested.

Structure : L, n {n = # of ECVs}
7. <V ECV_1>
.
.
n. <V ECV_n>

Exception : A zero length ECV_i means that $ECID_i$ does not exist.

Note :
ECV_i Data for the ECID in the format defined for that particular variable.
If the Host requests a large (complete) list, the equipment may take a few seconds to reply as the data can be quite large.

S2, F15 New Equipment Constant Send (ECS) **S , H → E, reply**

Description : Host requests a change to one or more equipment constant variables.

Structure : L, n {n = # of ECIDs}
1. L, 2
    1. <U2 $ECID_1$>
    2. <V ECV_1>
2. L, 2
.
.
n. L, 2
    1. <U2 $ECID_n$>
    2. <V ECV_n>

Note :
ECV Must be in the same data format as specified for the particular ECID.

S2, F16 Equipment Constant Acknowledge (ECA) **S, H ← E**

Description : Acknowledge or error. If EAC contains a non-zero error code, then it means that the equipment has NOT changed any of the ECIDs specified in the S2, F15 request.

Structure : <B EAC>

Note :
EAC Indicates whether the equipment constants were updated.
0 = Acknowledge
1 = Denied. At least one constant does not exists
2 = Denied. Busy
3 = Denied. At least one constant out of range
>3 = Other equipment specific error
4~63 Reserved

S2, F29 Equipment Constant Namelist Request (ECNR) **S , H → E, reply**

Description : Host requests for the equipment to send basic information about what equipment constants are available.

Structure : L, n {n = # of ECIDs}
8. <U2 $ECID_1$>
.
.
n. <U2 $ECID_n$>

Exception : A zero length list means send information for all ECIDs.

S2, F30 Equipment Constant Namelist (ECN) **M, H ← E**

Description : Data of Equipment Constants in the order requested.

Structure : L, n {n = # of equipment constants}
1. L, 6
1. <U2 $ECID_1$>
2. <A $ECNAME_1$>
3. <V $ECMIN_1$>
4. <V $ECMAX_1$>
5. <V $ECDEF_1$>
6. <A $UNITS_1$>
.
.
n. L, 6
1. <U2 $ECID_n$>
2. <A $ECNAME_n$>
3. <V $ECMIN_n$>
4. <V $ECMAX_n$>
5. <V $ECDEF_n$>
6. <A $UNITS_n$>

Note :

ECNAME	Text description of the equipment constant
ECMIN, ECMAX, ECDEF	Min, Max, Default value of the constant in the required type. These are not valid for textual type of equipment constants.
UNITS	Text description of the data units.

If the Host requests a large (complete) list, the equipment may take a few seconds to reply as the data can be quite large.

4.7.2. Equipment Constant Scenarios

Host Request a name list of Equipment Constants

COMMENT	HOST	EQUIPMENT	COMMENT
Equipment Constant Namelist Request (ECNR)			
	S2, F29 →		
		Equipment Constant Namelist (ECN) ← S2, F30	

Host Changes the Value of some Equipment Constants

COMMENT	HOST	EQUIPMENT	COMMENT
New Equipment Constant Send (ECS)			
	S2, F15 →		
		Equipment Constant Acknowledge (ECA) ← S2, F16 (EAC = 0)	

Host Requests for the Current Value of some Equipment Constants

COMMENT	HOST	EQUIPMENT	COMMENT
Equipment Constant Request (ECR)			
	S2, F13 →		
		Equipment Constant Data (ECD) ← S2, F14	

4.8. PROCESS PROGRAM MANAGEMENT

This capability allows the management and transfer of process programs between the host and the wire bonder. However, the wire bonder is able to load and execute only one process program at a time. No modification or uploading / downloading of any process program may occur while that program is being executed. The Eagle Xtreme Gold Wire Bonder supports only unformatted program operations. ASM Wire bonders support both host initiated as well as equipment initiated uploading and downloading. Please avoid uploading and downloading of bond program during the bonding, because during that time equipment is busy and so in host side the process may fail due to reply timeout error.

Note: ***Uploading*** refers data flow direction is from equipment to Host
Downloading refers data flow direction is from host to equipment

4.8.1. Process Program Management Messages

These messages are used to manage and transfer process program. Commonly known as Recipe Upload / Download, but also includes directory listing and file deletion.

S7, F1 Process Program Load Inquire (PPI) S, H ⇔ E, reply

Description : This message is used to initiate the transfer of a process program or to select from stored programs. The message may be used to initiate the transfer of an unformatted process program (S7,F3 / S7,F4).

Structure : L, 2
1. <A PPID>
2. <U4 LENGTH>

Note :

PPID ASCII string with a maximum length of 24 bytes and includes and extension of ".WB" (optional). Valid characters are A-Z, 0-9 and a single period.
Eg. "SIOC389238.WB"

LENGTH Indicates the size of the PPBODY that will be sent with S1,F3 in bytes. This is required so that the machine can reserve sufficient memory to receive the incoming file.

S7, F2 Process Program Load Grant (PPG) S, H ⇔ E

Description : This message gives permission for the process program to be loaded.

Structure : <B PPGNT>

Note :

PPGNT 1 byte value representing the permission status as follows
0 = OK (allowed to download the file PPBODY with S7,F3)
1 = File already exists
2 = Insufficient space available to accept the file (memory)
3 = Invalid PPID
4= Busy, try later
5 = Will not accept
> 5 Other error
6~63 Reserved

S7, F3 Process Program Send (PPS) **M, H ⇔ E, reply**

Description : The program is sent. If S7,F3 is multi-block, it must be preceded by the S7,F1 / S7,F2 Inquire / Grant transaction.

Structure : L, 2
1. <A PPID>
2. <B PPBODY>

Note :

PPID ASCII string with a maximum length of 24 bytes and includes and extension (optional) of ".WB". Valid characters are A-Z, 0-9 and a single period.
Eg. "SIOC389238.WB"
This must match the PPID name sent in any preceding S7,F1 / S7,F3 transaction.

PPBODY Binary data of the entire WB archive. The length must be the same size as LENGTH specified in any preceding S7,F1 / S7,F2 transaction.

S7, F4 Process Program Acknowledge (PPA) **S, H ⇔ E**

Description : Acknowledge or error.

Structure : <B ACKC7>

Note :

ACKC7 1 byte value representing the acknowledge of the file as follows
0 = Accepted without error
1 = Permission not granted
2 = Length or Data Corruption error
4 = PPID not found
6 = Equipment busy
7~63 Reserved

Warning : If the S7,F3 message is multi-block and not preceded with the S7,F1 / S7,F2 transaction, then the equipment will respond with S9,F11 Data Too Long (DLN) followed by many S9,F7 Illegal Data (IDN) for each subsequent block. This is due to the equipment not having a sufficient receive buffer to accept a large primary message. After the S9,F11 is sent, all subsequent blocks will be decoded, but because it is a pure data block the format is invalid and thus the S9,F7 is sent.

S7, F5 Process Program Request (PPR) **S, H ⇔ E, reply**

Description : This message is used to request the transfer of a process program.

Structure : <A PPID>

Note :
PPID ASCII string with a maximum length of 24 bytes and includes and extension (optional) of ".WB".(optional). Valid characters are A-Z, 0-9 and a single period.
Eg. "SIOC389238.WB"

S7, F6 Process Program Data (PPD) **M, H ⇔ E**

Description : This message is used to transfer a process program.

Structure : L, 2
1. <A PPID>
2. <B PPBODY>

Exception : A zero-length list means request denied.
A zero length list in S7F6 will be sent under the following conditions:
1. File doesn't exist
2. File exists but is empty
3. Cannot allocate a memory buffer to hold the file contents
4. Unable to read the file contents
5. Callback to application function does not return GEMOK status.

Note :
PPID ASCII string with a maximum length of 24 bytes and includes and extension (optional) of ".WB". Valid characters are A-Z, 0-9 and a single period.
Eg. "SIOC389238.WB"
This must match the PPID name sent in the preceding S7,F5.
PPBODY Binary data of the entire WB archive.

S7, F17 Delete Process Program Send (DPS) **S, H → E, reply**

Description : This message is used by the host to request the equipment to delete process programs from equipment storage.

Structure : L, n {n = Number of process programs to be deleted}
9. <A $PPID_1$>
.
.
n. <A $PPID_n$>

Exception : If n=0, then delete all Process Programs

Note :
PPID ASCII string with a maximum length of 24 bytes and includes and extension (optional) of ".WB". Valid characters are A-Z, 0-9 and a single period.
Eg. "SIOC389238.WB"

S7, F18 Delete Process Program Acknowledge (DPA) **S , H ← E**

Description : Acknowledge or error

Structure : <B ACKC7>

Note :
ACKC7 1 byte binary indicating the status of the delete operation
0 = Accepted
1 = Permission not granted
2 = Length error
4 = PPID not found
6 = Equipment busy
7~63 Reserved

S7, F19 Current EPPD Request (RER) **S, H → E, reply**

Description : This message is used to request the transmission of the current equipment process program directory (EPPD). This is a list of all the PPIDs of the process programs stored in the equipment.

Structure : Header only

S7, F20 Current EPPD Data (RED) **M, H ← E**

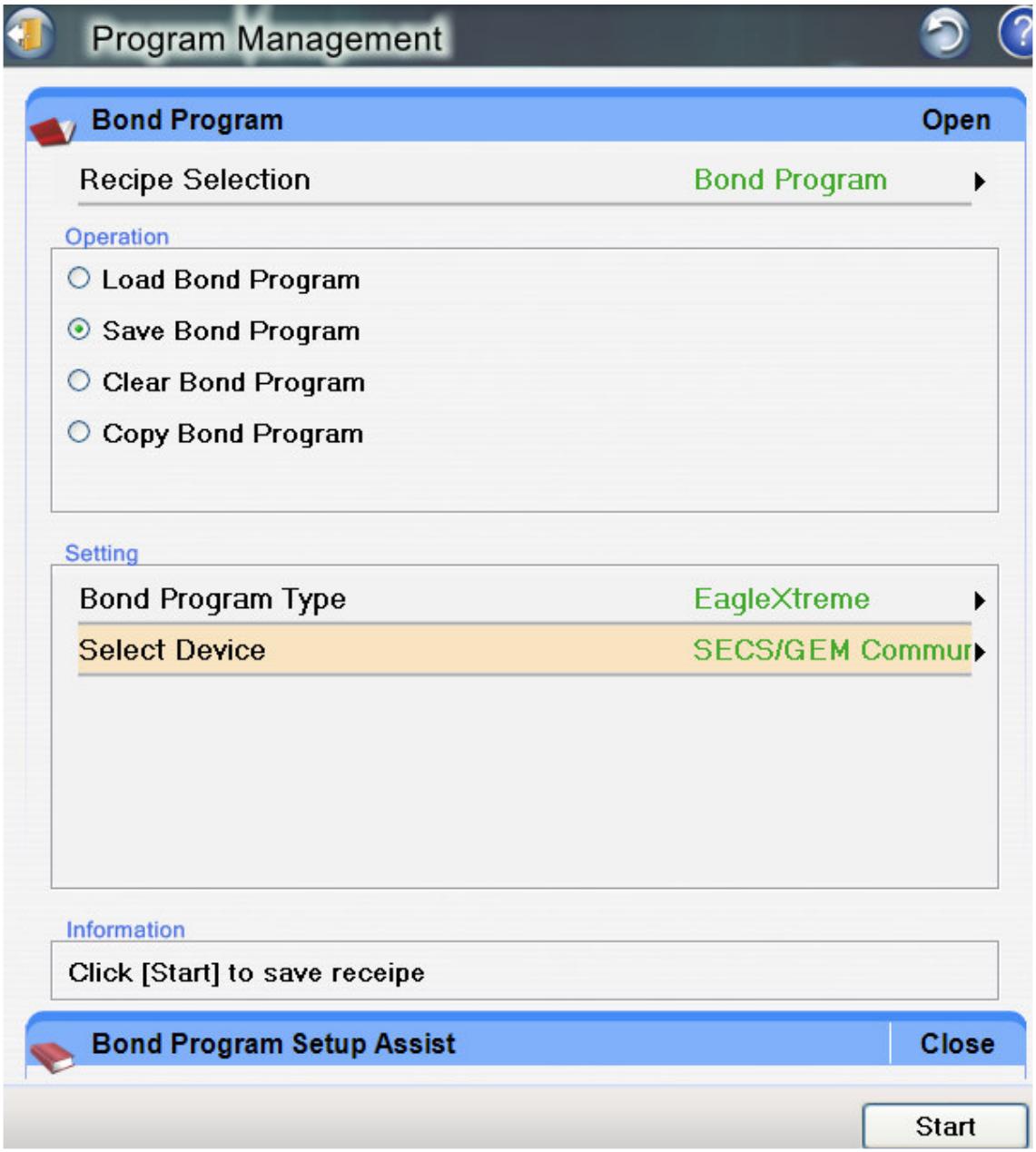
Description : This message is used to transmit the current EPPD.

Structure : L, n {n = number of process programs in the directory}
10. <A $PPID_1$>
.
.
n <A $PPID_n$>

Note :
PPID ASCII string with a maximum length of 24 bytes and includes and extension (optional) of ".WB". Valid characters are A-Z, 0-9 and a single period.
Eg. "SIOC389238.WB"

4.8.2. Process Program Management

1. Equipment Sends a process program to Host



On the Equipment, this function will be called when the operator selects menu items
1. Save Bond Program
2. Copy Bond Program

from the Program management page.
The Select Device should be SECS/GEM Communication

If the operator selects **Save Bond Program to Host** then the following steps will be performed:

2. The operator will be prompted to enter a name under which to save the Recipe currently stored in memory.
3. The Recipe will then be saved from memory to a temporary directory of the hard disk.
4. The relevant files will be compressed into a WB archive.
5. The compressed archive will be sent to the Host (S7,F1 / S7,F2 followed by S7,F3 / S7,F4) with an on screen message informing the user of the upload operation.
6. The files stored on temporary directory of the hard disk will be deleted.

If the operator selects **Copy Prog to Host**, then these steps will occur:

1. The operator will be presented with a list of all the available files stored on the hard disk. The operator will select which file should be sent to the Host.
2. The relevant files will be compressed into a WB archive and stored in a temporary directory of hard disk .
3. The compressed archive will be sent to the Host (S7,F1 / S7,F2 followed by S7,F3 / S7,F4) with an on screen message informing the user of the upload operation.
4. The compressed archive file stored on temporary directory of the hard disk will be deleted.

Note :

1)

If **Transfer PR Images** (from SECS/GEM Options) is set to ENABLE, then the PR pattern files will also be stored in the archive. If set to DISABLE, only the bonding co-ordinates and parameters will be stored. This can significantly reduce the size of the file transferred and thus reduce the transfer time. The side effect is that during the loading of the Recipe to memory, there will be a message "Unable to load templates" and the operator will need to manually reload the PR.

2)

If **Download Recipe to** (from SECS/GEM Options) is set as

a) **HDD only** : After downloading , the recipe will be stored in equipment's hard disk only.

b) **RAM only** : After downloading , the recipe will be stored into equipment's RAM (Random Access Memory) only. In this case recipe after downloading will be automatically loaded into equipment's RAM . No copy will be stored in Hard disk.

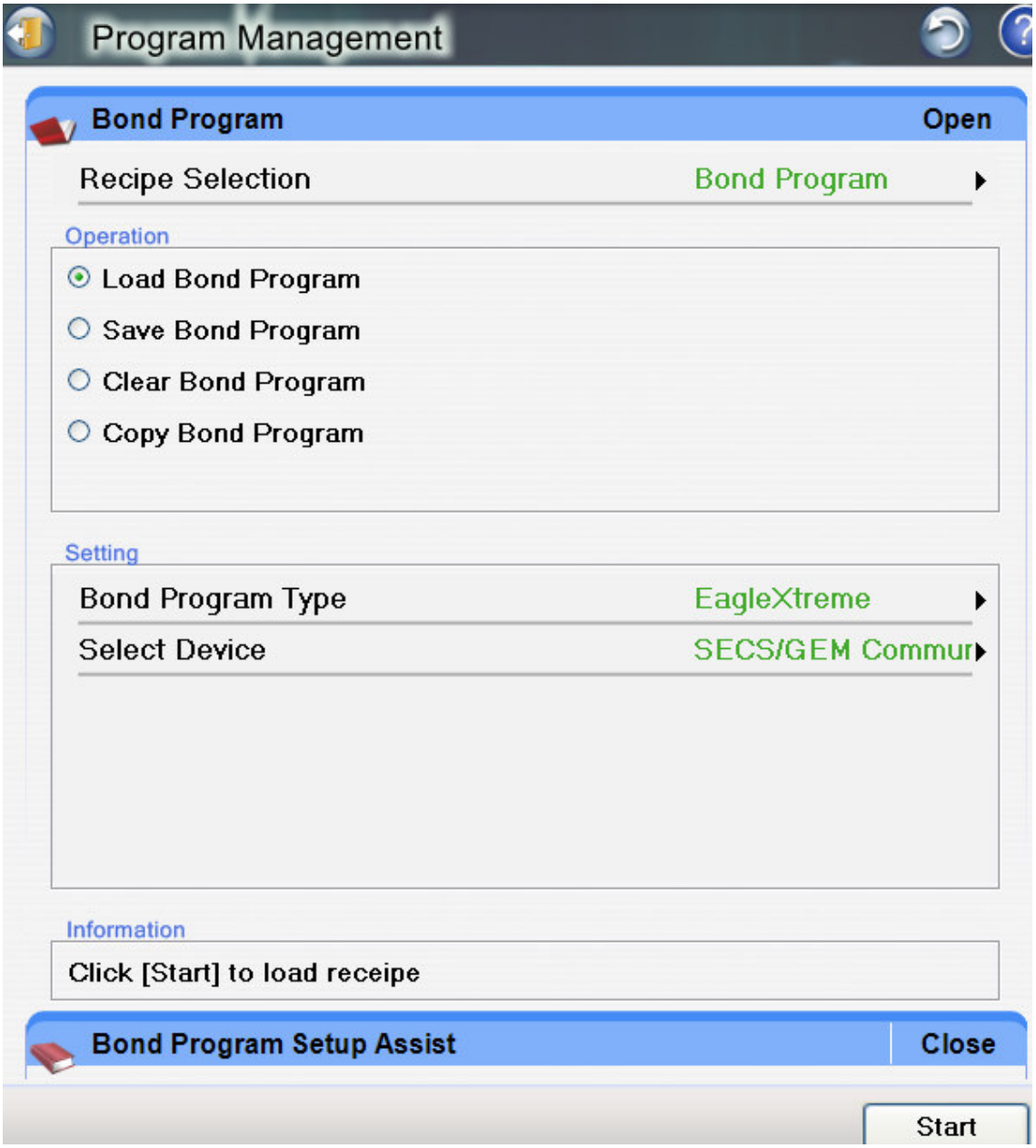
c) **HDD&RAM** : After downloading , the recipe will be first stored into equipment's hard disk and then it automatically loaded into equipment's RAM

3)

If **Host Recipe Extension** (from SECS/GEM Options) is set to ENABLE , then the recipe name will be saved in host computer hard disk with extension ".WB". Eg: TSSOP24T.WB
If this option set to DISABLED then the recipe name will be saved in host computer hard disk without extension ".WB" eg: TSSOP24T

COMMENT	HOST	EQUIPMENT	COMMENT
		Process Program Load Inquire (PPI) ← S7, F1	
Process Program Load Grant (PPG)	(PPGNT = 0) S7, F2 →		
		Process Program Send (PPS) ← S7, F3	
Process Program Acknowledge (PPA)	S7, F4 →		

1. Equipment requests Host send Process Program



On the Equipment, this function will be called when the operator selects the menu item
1. Load Bond Program
from the Program Management page.
The select Device should be SECS/GEM Communication

If the operator selects **Load Bond Program from Host**, then the following steps will be performed:

1. The operator will be prompted to enter the name of a Recipe currently stored on the Host computer.
2. If the file selected is already available on the equipment's hard disk, then the operator will be asked to confirm whether to overwrite the existing file. If the operator answers negative the operation ceases at this point.
3. S7,F5 will be sent to the Host with the name encoded as PPID
4. If the file is available on the Host, then the replied S7,F6 will contain the data for the compressed file archive.
5. If the file is not available on the Host, then the replied S7,F6 will only contain an empty list. The equipment will display a message on the screen indicating that the load operation failed.
6. Once the file archive has been received, it will be placed on the hard disk in the appropriate format. If the option is "RAM only" then no copy will be stored in the hard disk
7. After successful (no corruption detected) downloading and storage, the operator will be asked to confirm whether to load this file into memory.
8. If the operator answers affirmative, then the Recipe will be loaded into memory and become the current Process Program in use.

Once the file has been downloaded and stored to the hard disk, it will be available at any time to be manually loaded by the operator using the normal Load from Hard Disk menu item.

COMMENT	HOST	EQUIPMENT	COMMENT
		Process Program Request (PPR) ← S7, F5	
Process Program Data (PPD)	S7, F6 →		

2. Host requests Equipment send Process Program

If the Host requests the file with S7,F5 and the file exists on the hard disk, then the following occurs:

1. Compress the files into a WB archive.
2. If the PPID doesn't exist, then an empty list will be sent to the Host via S7,F6.
3. If the PPID does exists, then the WB data will be sent to the Host via S7,F6.

COMMENT	HOST	EQUIPMENT	COMMENT
Process Program Request (PPR)	S7, F5 →		
		Process Program Data (PPD) ← S7, F6	

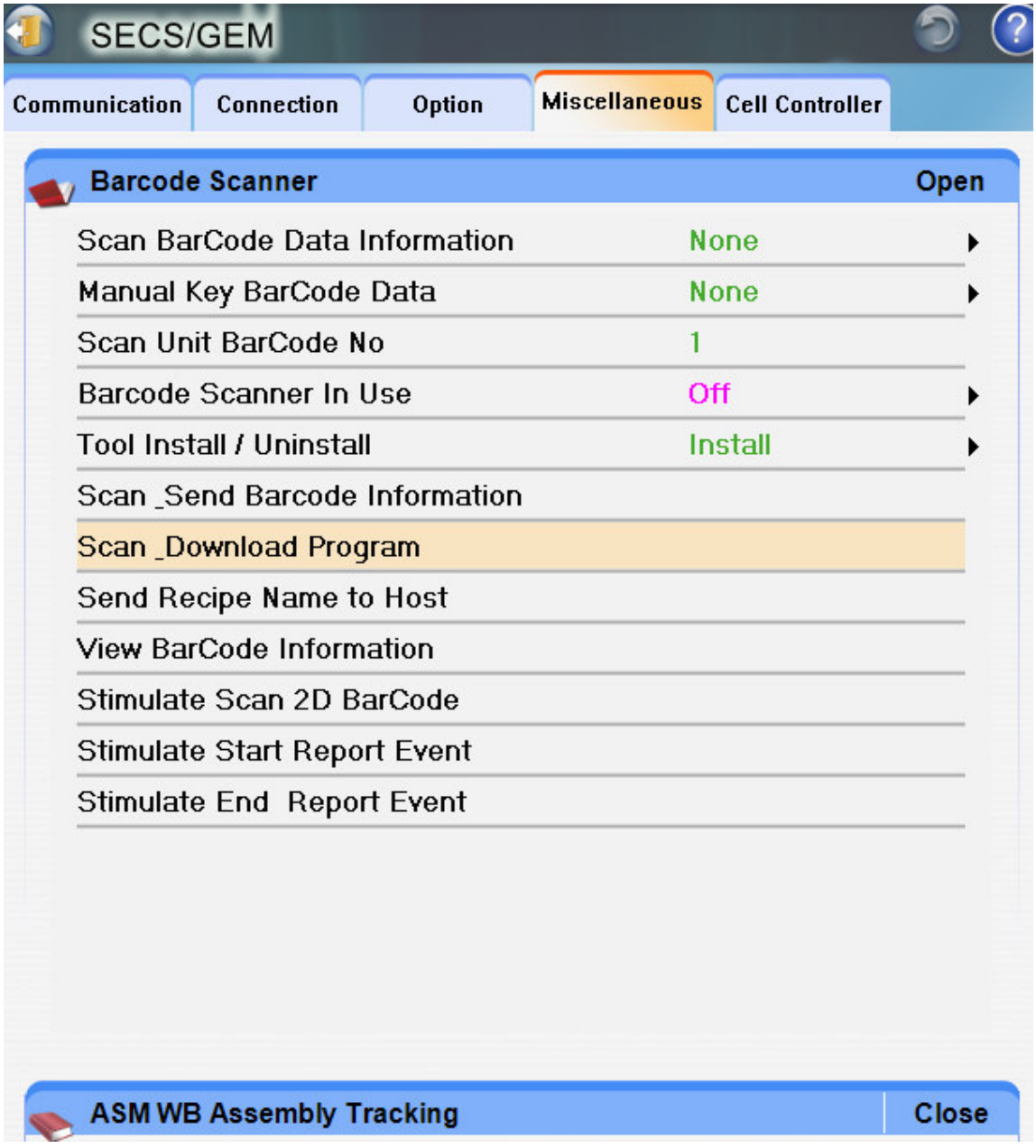
Host Sends Process Program to Equipment

COMMENT	HOST	EQUIPMENT	COMMENT
Process Program Load Inquire (PPI)	S7, F1 →		
		Process Program Load Grant (PPG) ← S7, F2 (PPGNT = 0)	
	Process Program Send (PPS) S7, F3 →		
		Process Program Acknowledge (PPA) ← S7, F4	

3. Equipment download program through Barcode scanner

This allows the operator to download the bond program by scanning the Barcode

The menu is located under "SECS/GEM Communications"



On the Equipment, this function will be called when the operator selects the menu item
1. Scan & Download Program
from the Barcode Scanner page.

If the operator selects **Scan &Download Program**, then the following steps will be performed:

2) The operator will be prompted to Scan the Barcode with a Barcode scanner
3) After successful scan the Barcode string (the program name) will be displayed on the bonder monitor and user can press "Enter" button to send the program name to host or press "STOP" button to quit out from the message and discard the scan result.
4) Once operator press ENTER button after barcode scan, S7,F5 will be sent to the Host with the program name encoded as PPID
5) If the file is available on the Host, then the replied S7,F6 will contain the data for the compressed file archive.
6) If the file is not available on the Host, then the replied S7,F6 will only contain an empty list. The equipment will display a message on the screen indicating that the load operation failed.
7) Once the file archive has been received, it will be placed on the hard disk in the appropriate format. If the option is "RAM only" then no copy will be stored in the hard disk
8) After successful (no corruption detected) downloading and storage, the operator will be asked to confirm whether to load this program into memory.
9) If the operator answers affirmative, then the Recipe will be loaded into memory and become the current Process Program in use.

Once the file has been downloaded and stored to the hard disk, it will be available at any time to be manually load by the operator using the normal Load from Hard Disk menu item.

COMMENT	HOST	EQUIPMENT	COMMENT
		Process Program Request (PPR) ← S7, F5	
Process Program Data (PPD)	S7, F6 →		

1. Host requests Process Program Deletion

Purpose : Allows the Host to selectively delete Recipe files from the Equipment's hard disk (main directory only as the Host cannot access sub-directories). One reason to do this could be to ensure the operator cannot load the wrong Recipe from the hard disk.

Eg.
The host deletes all the Recipes on the hard disk before downloading the Recipe file to use for the next lot. It is then impossible for the operator to load the wrong file, because the latest download will be the only file available.

Eg.
Recipe files that are obsolete can be deleted from all the machines directories for better housekeeping practices.

COMMENT	HOST	EQUIPMENT	COMMENT
Delete Process Program Send (DPS)			
	S7, F17 →		
		Delete Process Program Acknowledge (DPA) Delete files from hard disk ← S7, F18	

2. Host Request Process Program Directory Contents

Purpose : Allows the Host to find out what are the currently available Recipe files stored on the equipment's hard disk (main directory only as the Host cannot access sub-directories).

One reason to do this could be to check whether a Recipe (that is scheduled to be downloaded) is already on the hard disk of the machine, and thus doesn't need to be sent again.

Another use may be to double check that the file sent from the host has been received correctly and stored in the hard disk.

It could also be used to double check that the file selected for deletion has really been removed from the equipment's hard disk.

COMMENT	HOST	EQUIPMENT	COMMENT
Current EPPD Request (RER)			
	S7, F19 →		
		Current EPPD Data (RED) ← S7, F20	

4.9. ERROR MESSAGE

Error messages provide the host with information describing the reason for a particular message or communication fault detected by the wire bonder. The wire bonder will inform the host that it cannot process a message due to an incorrect :

- device ID
- message stream type
- message function
- message format

- data format
- message has more data than it can handle
- wire bonder transaction timer expires

The wire bonder treats the above conditions as application level errors and shall not take any further action on any message in error. Error messages are invoked whenever the wire bonder detects communication or message faults. Reader may refer to ASMGEM Protocol Reference for more detail.

4.9.1. Error Messages

S9, F1 Unrecognized Device ID (UDN) **S, H ← E**

Description : The Device ID in the message block header did not correspond to the setting of the Device ID in the equipment.

Structure : <B MHEAD>

Note :
MHEAD Block header of the message that generated the error.

S9, F3 Unrecognized Stream Type (USN) **S, H ← E**

Description : The equipment does not recognize the stream type specified in the message.

Structure : <B MHEAD>

Note :
MHEAD Block header of the message that generated the error.

S9, F5 Unrecognized Function Type (UFN) **S, H ← E**

Description : The equipment does not recognize the function type specified in the message.

Structure : <B MHEAD>

Note :
MHEAD Block header of the message that generated the error.

S9, F7 Illegal Data (IDN) **S, H ← E**

Description : The stream and function information in the message was recognized, but a data format for an item was unexpected or unrecognized.

Structure : <B MHEAD>

Note :
MHEAD Block header of the message that generated the error.

S9, F9 Transaction Timer Timeout (TTN) **S, H ← E**

Description : Informs the host that a reply message was not received within the T3 (Transaction Timeout) limit.

Structure : <B SHEAD>

Note :
SHEAD Block header of the primary message sent for which a reply was expected.

S9, F11 Data Too Long (DLN) **S, H ← E**

Description : Indicates that the equipment cannot handle a message because it contains more data than the receive buffer can manage.

Structure : <B MHEAD>

Note :
MHEAD Block header of the message that generated the error.

S9, F13 Conversation Timeout (CTN) **S, H ← E**

Description : A message was expected (for which resources have been allocated), but never received within the conversation timeout limit. The resources have been released when this message was sent.

Structure : L, 2
1. <A[6] MEXP>
2. <V EDID>

Note :
MEXP Expected message that was not received in the form "SxxFyy"
EDID Expected data identification. Currently conversation timeout only occurs when waiting for an S7, F3. Under this condition, EDID is set to the expected PPID. ie. <A EDID> = <A PPID>

4.9.2. Error Message Scenarios

Message Content Faults

COMMENT	HOST	EQUIPMENT	COMMENT
Host sends a message			
	Sx, Fy →		
		[IF] Message Device ID doesn't match equip Device ID [THEN] Unrecognized Device ID (UDN) ← S9, F1 (Header from Sx, Fy)	
		[ELSE_IF] Stream type in message is not recognized [THEN] Unrecognized Stream Type (USN) ← S9, F3 (Header from Sx, Fy)	
		[ELSE_IF] Function type in message is not recognized [THEN] Unrecognized Function Type (UFN) ← S9, F5 (Header from Sx, Fy)	
		[ELSE_IF] Data content of message or item invalid [THEN] Illegal Data (IDN) ← S9, F5 (Header from Sx, Fy)	
		[ELSE_IF] Sx, Fy is larger than the receive buffer [THEN] Data Too Long (DLN) ← S9, F5 (Header from Sx, Fy) [END_IF]	

Message Transaction Timeout Fault

COMMENT	HOST	EQUIPMENT	COMMENT
		Equipment sends a message expecting a reply ← Sx, Fy	
	…	…	
		No reply message received causing the transaction timer to expire Transaction Timer Timeout (TTN) ← S9, F9 (Header from Sx, Fy)	

Conversation Timeout Fault

COMMENT	HOST	EQUIPMENT	COMMENT
Host sends a message warning that a message will follow that requires special set up of equipment resources (memory)			
	Sx, Fy →		
		Equipment allocates resources, starts conversation timer, and replies. Equipment stores what is the next expected transaction. ← Sx, Fy+1	
	…	…	
		No new message received causing the conversation timer to expire OR message received that was not part of the expected transaction Resources are released Conversation Timer Timeout (CTN) ← S9, F13 (MEXP = Expected "SxxFyy")	

4.10. CLOCK

The clock capability enables host management of time-related activities and occurrences associated with the wire bonder. The purpose of the clock capability is to enable time stamping of collection event and alarm reports. Time stamping is useful for resolving relative order of event/alarm occurrences and scheduling of wire bonder activities by the host.

The ability for the host to instruct the wire bonder to set an internal clock to a specified time value is needed for effective time management and synchronization between host and equipment.

The clock supports 2 different modes of operations.
<A TIME> can be either:

11. Non-Y2k Compliant 12 digit format "YYMMDDhhmmss"
12. Y2k Compliant 16 digit format "YYYYMMDDhhmmsscc"

Where:

YY	= Year	00 to 99
YYYY	= Year2k	0000 to 9999
MM	= Month	01 to 12
DD	= Day	01 to 31
hh	= Hour	00 to 23
mm	= Minute	00 to 59
ss	= Second	00 to 59
cc	= Centi-second	00 to 99

The mode of operation is selected via the Equipment Constant TIMEFORMAT.

4.10.1. Clock Messages

S2, F17 Date and Time Request (DTR) **S, H ⇔ E**

Description : Allows the equipment or the Host to request the current date and time settings on the other party.

Structure : Header only.

S2, F18 Date and Time Data (DTD) **S, H ⇔ E**

Description : Current date and time data

Structure : <A TIME>

Note :
TIME 12 / 16 byte data of the date and time.

S2, F31 Date and Time Set Request (DTS) **S, H → E**

Description : Allows the Host to set the date and time of the equipment.

Structure : <A TIME>

S2, F32 Date and Time Set Acknowledge (DTA) **S, H ← E**

Description : Acknowledge that the date and time has been set

Structure : <B TIACK>

Note :
TIACK Time Acknowledge Code
0 = OK
1 = Error, not done
2~63 Reserved

4.10.2. Clock Scenarios

Get Current Equipment Date / Time

COMMENT	HOST	EQUIPMENT	COMMENT
Host requests			
Date and Time Request (DTR)			
S2, F17 →			
		Data and Time Data (DTD)	
		← S2, F18	
		Equipment Requests	
		Date and Time Request (DTR)	
		← S2, F17	
Date and Time Data (DTD)			
	S2, F18 →		

Set Equipment Date / Time

COMMENT	HOST	EQUIPMENT	COMMENT
Date and Time Set Request (DTS)			
	S2, F31 →		
		Date and Time Set Acknowledge (DTA)	
		← S2, F32	

4.11. TERMINAL SERVICES

The Eagle Xtreme supports a simple text display method that allows the Host to display an English text message to the operator. This could be used to inform the operator of why the equipment has been stopped by the host, or to give information on a QC result.

4.11.1. Terminal Service Messages

S10, F3 Terminal Display, Single (VTN) **S, H → E, reply**

Description : English message to be displayed

Structure : L, 2
1. <B TID>
2. <A TEXT>

Note :

TID	Terminal number should always be set to 0= Single or Main terminal
TEXT	English display text (ASCII) less than 80 chars

S10, F4 Terminal Display, Single Acknowledge (VTA) **S, H ← E**

Description : Acknowledge or error

Structure : <B ACKC10>

Note :

ACKC10	Acknowledge Code 0 = Accepted for display 1 = Message will not be displayed 2 = Terminal not available 3~63 Reserved

4.11.2. Terminal Service Scenarios

Host Displays an English Message on the Screen

COMMENT	HOST	EQUIPMENT	COMMENT
Terminal Display, Single (VTN)			
	S10, F3 →		
		Message displayed on the screen Terminal Display, Single Acknowledge (VTA) ← S10, F4	

4.12. DIAGNOSTIC TESTS

To check that the hardware and software link is operational, a Loop back Diagnostics test can be performed periodically. This will send a block of data between the Host / Equipment and the other party will echo the data back exactly. If the received data does not match the sent data, then there is something wrong with the hardware and action needs to be taken to diagnose why.

4.12.1. Diagnostic Messages

S2, F25 Loop back Diagnostic Request (LDR) **S, H ⇔ E, reply**

Description : Sends a binary string that should be echoed to test the protocol and communication circuits.

Structure : <B ABS>

Note :
ABS Any binary data (size < 200) that should be echoed back

S2, F26 Loop back Diagnostic Data (LDD) **S, H ⇔ E**

Description : The echoed binary string

Structure : <B ABS>

Note :
ABS Will be exactly the same as the sent data if the hardware / software link is working correctly.

4.12.2. Diagnostic Scenarios

Loop back Test the Communication Link

COMMENT	HOST	EQUIPMENT	COMMENT
Host Tests			
Loop back Diagnostic Request (LDR)			
	S2, F25 →		
		Loop back Diagnostic Data (LDD) ← S2, F26	
		Equipment Tests Loop back Diagnostic Request (LDR) ← S2, F25	
Loop back Diagnostic Data (LDD)			
	S2, F26 →		

5. SECS MESSAGE SUBSET

The following sections contain a summary of the SECS-II message subsets that are implemented on the Eagle Xtreme wire bonder.

5.1. Host Initiated Messages

The following table lists a set of host initiated messages and equipment responses.

	HOST INITIATES				WIRE BONDER RESPONSE	
S/F	**NAME**	**MN**		**S/F**	**NAME**	**MN**
S1F1	Are You There Request	R		S1F2	On Line Data	D
S1F3	Selected Equipment Status Request	SSR		S1F4	Selected Equipment Status Data	SSD
S1F11	Status Variable Namelist Request	SVNR		S1F12	Status Variable Namelist Reply	SVNRR
S1F13	Establish Communications Request	CR		S1F14	Establish Communications Request Acknowledge	CRA
S1F15	Request OFF-LINE	ROFL		S1F16	OFF-LINE Acknowledge	OFLA
S1F17	Request ON-LINE	RONL		S1F18	ON-LINE Acknowledge	ONLA
S2F13	Equipment Constant Request	ECR		S2F14	Equipment Constant Data	ECD
S2F15	New Equipment Constant Send	ECS		S2F16	New Equipment Constant Acknowledge	ECA
S2F17	Date and Time Request	DTR		S2F18	Date and Time Data	DTD
S2F21	Remote Command Send	RCS		S2F22	Remote Command Acknowledge	RCA
S2F25	Loop back Diagnostic Request	LDR		S2F26	Loop back Diagnostic Data	LDD
S2F29	Equipment Constant Namelist Request	ECNR		S2F30	Equipment Constant Namelist	ECN
S2F31	Date and Time Set Request	DTS		S2F32	Date and Time Set Acknowledge	DTA
S2F33	Define Report	DR		S2F34	Define Report Acknowledge	DRA
S2F35	Link Event Report	LER		S2F36	Link Event Report Acknowledge	LERA
S2F37	Enable/Disable Event Report	EDER		S2F38	Enable/Disable Event Report Acknowledge	EDEA
S2F39	Multi-Block Inquire	MBI		S2F40	Multi-Block Grant	MBG
S2F41	Host Command Send	HCS		S2F42	Host Command Acknowledge	HCA
S5F3	Enable/Disable Alarm Send	EAS		S5F4	Enable/Disable Alarm Acknowledge	EAA
S5F5	List Alarms Request	LAR		S5F6	List Alarm Data	LAD
S5F7	List Enabled Alarm Request	LEAR		S5F8	List Enabled Alarm Data	LEAD
S6F15	Event Report Request	ERR		S6F16	Event Report Data	ERD
S6F19	Individual Report Request	IRR		S6F20	Individual Report Data	IRD
S7F1	Process Program Load Inquire	PPI		S7F2	Process Program Load Grant	PPG
S7F3	Process Program Send	PPS		S7F4	Process Program Acknowledge	PPA
S7F5	Process Program Request	PPR		S7F6	Process Program Data	PPD
S7F17	Delete Process Program Send	DPS		S7F18	Delete Process Program Acknowledge	DPA
S7F19	Current EPPD Request	RER		S7F20	Current EPPD Data	RED
S10F3	Terminal Display, Single	VTN		S10F4	Terminal Display, Single Acknowledge	VTA

5.2. Equipment Initiated Messages

The following table lists the wire bonder initiated messages and corresponding expected host response.

	WIRE BONDER INITIATES				HOST RESPONSE	
S/F	**NAME**	**MN**		**S/F**	**NAME**	**MN**
S1F1	Are You There Request	R		S1F2	On Line Data	D
S1F13	Establish Communications Request	CR		S1F14	Establish Communications Request Acknowledge	CRA
S2F17	Date and Time Request	DTR		S2F18	Date and Time Data	DTD
S5F1	Alarm report Send	ARS		S5F2	Alarm Report Acknowledge	ARA
S6F5	Multi-block Data Send Inquire	MBI		S6F6	Multi-block Grant	MBG
S6F11	Event Report Send	ERS		S6F12	Event Report Acknowledge	ERA
S7F1	Process Program Load Inquire	PPI		S7F2	Process Program Load Grant	PPG
S7F3	Process Program Send	PPS		S7F4	Process Program Acknowledge	PPA
S7F5	Process Program Request	PPR		S7F6	Process Program Data	PPD
S9F1	Unrecognized Device ID	UDN				
S9F3	Unrecognized Stream Type	USN				
S9F5	Unrecognized Function Type	UFN				
S9F7	Illegal Data	IDN				
S9F9	Transaction Timer Timeout	TTN				
S9F11	Data Too Long	DLN				
S9F13	Conversation Timeout	CTN				
S14F1	Get Attribute Request	GAR		S14F2	Get Attribute Data	GAD
SxF0	Abort Transaction	SxF0				

6. EQUIPMENT CONSTANTS

ECID	Description	Type	Default	Max	Min	Comment
0	EQUIPMENT_NAME	6-byte ASCII	Eg XT	-	-	Internal only
1	SOFTWARE_REVISION	6-byte ASCII	020537	-	-	Internal only
2	COMMUNICATION_RETRY	2-byte int(us)	3	1000	1	Internal only
3	MAX_SPOOL_TRANSMIT	2-byte int(us)	40	100	0	Internal only
4	OVER_WRITE_SPOOL	boolean	0	1	0	Internal only
22	TIMEFORMAT	1-byte int(us)	0	1	0	Internal only
28	Lot Validity	1-byte int (us)	0	1	0	Applicable only for ASM IDEALine
29	User Logon ID	32 Byte ASCII	-	-	-	Applicable only for ASM IDEALine
31	SOFT-START Timeout	1 Byte US	60	255	0	-
33	Establish Comm Timeout	2-byte int(us)	5000	20000	1000	-
34	Alarm Pager Timeout	1-byte int(us)	0	60	0	-
399	Select Loop Mode	1-byte int(us)	0	255	0	-
400 to 464	Defect in Strip		-	-	-	Applicable only for ASM IDEALine
550 to 558	New Lot Information	-	-	-	-	Applicable only for ASM IDEALine
572	No of column in strip defect map	2 byte int(us)	0	100	0	Applicable only for ASM IDEALine
573	No. of row in strip defect map	2 byte int (us)	0	100	0	Applicable only for ASM IDEALine
600	Trigger SPC instruction	40 byte ASCII	-	-	-	Applicable only for ASM IDEALine
601	Trigger SPC Reset	4 byte int (us)	31	256	0	Applicable only for ASM IDEALine
611	eCheck upon pgm modification	1 byte int (us)	0	1	0	0: No 1: Yes
612	eCheck Allow Bond status	1 byte int (us)	1	2	0	0: No 1: Yes 2: Comparing
613	eCheck Bond sampling unit	2 bytes int (us)	0	30000	0	
614	eCheck by Host	1 byte int (us)	0	1	0	0: No 1: Yes
652	User name	30 bytes ASCII	-	-	-	(for ASM Internal QA)
653	Security PWD Level 1	10 bytes ASCII	-	-	-	To update Level 1
654	Security PWD Level 2	10 bytes ASCII	-	-	-	To update Level 2
655	Security PWD Level 3	10 bytes ASCII	-	-	-	To update Level 3
656	Security PWD Level 4	10 bytes ASCII	-	-	-	To update Level 4
658	User name	I byte int (us)	0	1		0: Login PASS 1: Login FAIL
659	Operator ID	32 bytes ASCII	-	-	-	For Special Customer
670	Current Lot ID	32 bytes ASCII	-	-	-	For Special Customer
671	Current Lot Quantity	4 byte int (us)	0	1410065407	0	For Special Customer
672	Remote Control Mode	1 byte int (us)	0	1	0	-
673	Statistic Start Time	32 Byte ASCII				Format "yyyy-mm-dd hh:mm:ss"
674	Statistic End Time	32 Byte ASCII				Format "yyyy-mm-dd hh:mm:ss"

675	BPC sampling unit during bonding	2 bytes int (us)	0	30000	0	-

Comment:
Internal Only – Can be set by the host, but is not displayed on the bonder menu
Available to Operator – Can be set by the Host or the Operator. When changed by the operator, a Change EC event will be sent to the Host informing of the variable adjusted.

7. STATUS VARIABLES

SVID	Description	Type	Default	Max	Min	Comment
5	Clock	Date & Time TIMEFORMAT = 0 12 characters (Non Y2K Compliant) YYMMDDHHMMSS TIMEFORMAT = 1 16 characters (Y2K Compliant) YYYYMMDDHHMMSSCC	-	-	-	-
6	Control State	Binary	1-EQUIP OFFLINE	5- ONLINE REMOTE	1-EQUIP OFFLINE	1-EQUIP OFFLINE 2-ATTEMP ONLINE 3-HOST OFFLINE 4-ONLINE LOCAL 5-ONLINE REMOTE
7	Previous Process State	2-byte int(s)	2-Idle	11-Upload	1-INIT	1-Init 2-Idle 3-Setup 4-ReadyLocal 5-Exe-Local 6-Pause 7-Alarm 8-Ready-Remote 9-Exe-Remote 10-Download 11-Upload
8	Process State	2-byte int(s)	2-Idle	11-Upload	2-Idle	Same as Previous Process State
9	SPOOL_COUNT_ACTUAL	2-byte int(us)	0	1000	0	-
10	SPOOL_COUNT_TOTAL	2-byte int(us)	0	1000	0	-
11	SPOOL_FULL_TIME	Date & Time	-	-	-	-
12	SPOOL_START_TIME	Date & Time	-	-	-	-
13	AlarmID	2-byte int(us)	0	255	0	-
14	ALARM_SET	1-byte int(us)	0	255	0	-
15	PauseReason	2-byte int(s)	0	2	0	0-Alarm 1-Equip / Operator 2-Host
16	OPERATOR_COMMAND	1-byte int(us)	0	100	0	-
17	PPChangeName	24-byte ASCII	-	-	-	-
18	PPChangeStatus	1-byte int(us)	1	3	1	1-Created 2-Edited

						3-Deleted
19	PPExecName	24-byte ASCII	-	-	-	-
20	TRANSITION_TYPE	1-byte int(us)				
21	EventsEnabled	List	-	-	-	List of current CEID enabled.
23	AlarmClock	16-byte ASCII	-	-	-	Date & Time
24	AlarmsSet	Variable list	-	-	-	ALID
25	Machine ID	11 Byte ASCII	-	-	-	-
26	Last PM Date	16 Byte ASCII	-	-	-	Date & Time
27	Installation Date	16 Byte ASCII	-	-	-	Date & TIME
30	EquipPerformLastDateReset	16-byte ASCII	-	-	-	Date & Time
35	Soft Start Activation	1-byte int(s)	0	1	0	0: Disable 1: Enable
36	Type of Wire Parameter changed	2-byte int (S)	0	16	0	0: No change 1: Force1 2: Force2 3: Power1 4: Power2 5: Time1 6: Time2 7: Contact Force1 8: Contact Force2 9: Contact Power1 10: Contact Power2 11: 12: Scrub Control 13: Stick Detect 1 14: Stick Detect 2 15: BSOB/BBOS 16: Tail Length
37	Barcode String	32 bytes – ASCII	-	-	-	
39	PPPreviousName	24-byte ASCII	-	-	-	-
41	AutoLoop – Reverse Height	2 byte-int (s)	10	100	10	
42	AutoLoop – Reverse Distance Factor	2 byte-int (s)	0	100	0	
43	AutoLoop – Loop Correction	2 byte-int (s)	0	100	-100	
44	AutoLoop – Loop Correction Slope	2 byte-int (s)	0	50	-50	
45	AutoLoop – Search Delay Base	2 byte-int (s)	1	50	1	
46	AutoLoop – Search Delay Factor	2 byte-int (s)	0	50	0	
47	AutoLoop – Sync Offset	2 byte-int (s)	0	32767	-32768	
48	AutoLoop – DEC Sample Offset	2 byte-int (s)	0	127	-127	
50	QautoLoop – Reverse Height	2 byte-int (s)	10	100	10	
51	QautoLoop – Reverse Distance Factor	2 byte-int (s)	0	500	0	
52	QautoLoop – Loop Correction	2 byte-int (s)	0	127	-127	

53	QautoLoop – Loop Correction Slope	2 byte-int (s)	0	50	-50
54	QautoLoop – Search Delay Base	2 byte-int (s)	1	50	1
55	QautoLoop – Search Delay Factor	2 byte-int (s)	0	50	0
56	QautoLoop – Sync Offset	2 byte-int (s)	0	32767	-32768
57	QautoLoop – Reverse Height Ratio	1 byte-int (us)	0	255	0
58	QautoLoop – DEC Sample Offset	2 byte-int (s)	0	127	-127
60	TSOP – Reverse Height	2 byte-int (s)	10	100	10
61	TSOP – Reverse Distance Factor	2 byte-int (s)	0	100	0
62	TSOP – Loop Correction	2 byte-int (s)	0	100	-100
63	TSOP – Loop Correction Slope	2 byte-int (s)	0	50	-50
64	TSOP – Search Delay Base	2 byte-int (s)	1	50	1
65	TSOP – Search Delay Factor	2 byte-int (s)	0	50	0
66	TSOP – Sync Offset	2 byte-int (s)	0	32767	-32768
67	TSOP – Reverse Height Ratio	1 byte-int (us)	0	4	0
68	TSOP – DEC Sample Offset	2 byte-int (s)	0	127	-127
70	NormalLoop – Reverse Height	2 byte-int (s)	10	100	10
71	NormalLoop – Reverse Distance Factor	2 byte-int (s)	0	100	0
72	NormalLoop – Loop Correction	2 byte-int (s)	0	100	-100
73	NormalLoop – Loop Correction Slope	2 byte-int (s)	0	50	-50
74	NormalLoop – Search Delay Base	2 byte-int (s)	1	50	1
75	NormalLoop – Search Delay Factor	2 byte-int (s)	0	50	0
76	NormalLoop – Sync Offset	2 byte-int (s)	0	32767	-32768
77	NormalLoop – DEC Sample Offset	2 byte-int (s)	0	127	-127
80	NLSquareLoop – Loop Height	2 byte-int (s)	2	500	5
81	NLSquareLoop – Flat Length	2 byte-int (s)	50	100	10
82	NLSquareLoop – Reverse Height	2 byte-int (s)	10	100	10
83	NLSquareLoop – Reverse Distance Factor	2 byte-int (s)	0	100	0
84	NLSquareLoop – Loop Correction	2 byte-int (s)	0	100	-100
85	NLSquareLoop – Loop Correction Slope	2 byte-int (s)	0	50	-50
86	NLSquareLoop – Search	2 byte-int (s)	1	50	1

	Delay Base				
87	NLSquareLoop – Search Delay Factor	2 byte-int (s)	0	50	0
88	NLSquareLoop – Sync Offset	2 byte-int (s)	0	32767	-32768
89	NLSquareLoop – Reverse Height Ratio	1 byte-int (us)	0	40	0
90	NLSquareLoop – DEC Sample Offset	2 byte-int (s)	0	100	0
91	NLSquareLoop – Horizontal Length	2 byte-int (s)	0	250	0
92	NLSquareLoop – Reverse Dist2 %	2 byte-int (s)	0	1000	0
93	NLSquareLoop – Reverse Dist2 Angle	1 byte-int (us)	0	255	0
95	LLSquareLoop – Loop Height	2 byte-int (s)	2	500	5
96	LLSquareLoop – Flat Length	2 byte-int (s)	50	100	10
97	LLSquareLoop – Reverse Height	2 byte-int (s)	10	100	10
98	LLSquareLoop – Reverse Distance Factor	2 byte-int (s)	0	100	0
99	LLSquareLoop – Loop Correction	2 byte-int (s)	0	100	-100
100	LLSquareLoop – Loop Correction Slope	2 byte-int (s)	0	50	-50
101	LLSquareLoop – Search Delay Base	2 byte-int (s)	1	50	1
102	LLSquareLoop – Search Delay Factor	2 byte-int (s)	0	50	0
103	LLSquareLoop – Sync Offset	2 byte-int (s)	0	32767	-32768
104	LLSquareLoop – Reverse Height Ratio	1 byte-int (us)	0	40	0
105	LLSquareLoop – DEC Sample Offset	2 byte-int (s)	0	100	0
106	LLSquareLoop – Horizontal Length	2 byte-int (s)	0	250	0
107	LLSquareLoop – Reverse Dist2 %	2 byte-int (s)	0	1000	0
108	LLSquareLoop – Reverse Dist2 Angle	1 byte-int (us)	0	255	0
110	NSSquareLoop – Loop Height	2 byte-int (s)	2	500	5
111	NSSquareLoop – Flat Length	2 byte-int (s)	50	100	10
112	NSSquareLoop – Reverse Height	2 byte-int (s)	10	100	10
113	NSSquareLoop – Reverse Distance Factor	2 byte-int (s)	0	100	0
114	NSSquareLoop – Loop Correction	2 byte-int (s)	0	100	-100
115	NSSquareLoop – Loop	2 byte-int (s)	0	50	-50

	Correction Slope				
116	NSSquareLoop – Search Delay Base	2 byte-int (s)	1	50	1
117	NSSquareLoop – Search Delay Factor	2 byte-int (s)	0	50	0
118	NSSquareLoop – Sync Offset	2 byte-int (s)	0	32767	-32768
119	NSSquareLoop – Reverse Height Ratio	1 byte-int (us)	0	40	0
120	NSSquareLoop – DEC Sample Offset	2 byte-int (s)	0	100	0
121	NSSquareLoop – Horizontal Length	2 byte-int (s)	0	250	0
122	NSSquareLoop – Reverse Dist2 %	2 byte-int (s)	0	1000	0
123	NSSquareLoop – Reverse Dist2 Angle	1 byte-int (us)	0	255	0
125	LSSquareLoop – Loop Height	2 byte-int (s)	2	500	5
126	LSSquareLoop – Flat Length	2 byte-int (s)	50	100	10
127	LSSquareLoop – Reverse Height	2 byte-int (s)	10	100	10
128	LSSquareLoop – Reverse Distance Factor	2 byte-int (s)	0	100	0
129	LSSquareLoop – Loop Correction	2 byte-int (s)	0	100	-100
130	LSSquareLoop – Loop Correction Slope	2 byte-int (s)	0	50	-50
131	LSSquareLoop – Search Delay Base	2 byte-int (s)	1	50	1
132	LSSquareLoop – Search Delay Factor	2 byte-int (s)	0	50	0
133	LSSquareLoop – Sync Offset	2 byte-int (s)	0	32767	-32768
134	LSSquareLoop – Reverse Height Ratio	1 byte-int (us)	0	40	0
135	LSSquareLoop – DEC Sample Offset	2 byte-int (s)	0	100	0
136	LSSquareLoop – Horizontal Length	2 byte-int (s)	0	250	0
137	LSSquareLoop – Reverse Dist2 %	2 byte-int (s)	0	1000	0
138	LSSquareLoop – Reverse Dist2 Angle	1 byte-int (s)	0	255	0
140	NISquareLoop – Loop Height	2 byte-int (s)	2	500	5
141	NISquareLoop – Flat Length	2 byte-int (s)	50	100	10
142	NISquareLoop – Reverse Height	2 byte-int (s)	10	100	10
143	NISquareLoop – Reverse Distance Factor	2 byte-int (s)	0	100	0
144	NISquareLoop – Loop	2 byte-int (s)	0	100	-100

	Correction				
145	NISquareLoop – Loop Correction Slope	2 byte-int (s)	0	50	-50
146	NISquareLoop – Search Delay Base	2 byte-int (s)	1	50	1
147	NISquareLoop – Search Delay Factor	2 byte-int (s)	0	50	0
148	NISquareLoop – Sync Offset	2 byte-int (s)	0	32767	-32768
149	NISquareLoop – Reverse Height Ratio	1 byte-int (us)	0	40	0
150	NISquareLoop – DEC Sample Offset	2 byte-int (s)	0	100	0
151	NISquareLoop – Horizontal Length	2 byte-int (s)	0	250	0
152	NISquareLoop – Reverse Dist2 %	2 byte-int (s)	0	1000	0
153	NISquareLoop – Reverse Dist2 Angle	1 byte-int (us)	0	255	0
155	LISquareLoop – Loop Height	2 byte-int (s)	2	500	5
156	LISquareLoop – Flat Length	2 byte-int (s)	50	100	10
157	LISquareLoop – Reverse Height	2 byte-int (s)	10	100	10
158	LISquareLoop – Reverse Distance Factor	2 byte-int (s)	0	100	0
159	LISquareLoop – Loop Correction	2 byte-int (s)	0	100	-100
160	LISquareLoop – Loop Correction Slope	2 byte-int (s)	0	50	-50
161	LISquareLoop – Search Delay Base	2 byte-int (s)	1	50	1
162	LISquareLoop – Search Delay Factor	2 byte-int (s)	0	50	0
163	LISquareLoop – Sync Offset	2 byte-int (s)	0	32767	-32768
164	LISquareLoop – Reverse Height Ratio	1 byte-int (us)	0	40	0
165	LISquareLoop – DEC Sample Offset	2 byte-int (s)	0	100	0
166	LISquareLoop – Horizontal Length	2 byte-int (s)	0	250	0
167	LISquareLoop – Reverse Dist2 %	2 byte-int (s)	0	1000	0
168	LISquareLoop – Reverse Dist2 Angle	1 byte-int (us)	0	255	0
180	File Transfer Mode	Boolean	True		
181	NSD – Stick detect 1	Boolean	False	-	-
182	NSD – Stick Detect 2	Boolean	False	-	-
183	Skip Bad Die	Boolean	False	-	-
184	Skip missing die	Boolean	False	-	-
185	Skip ink/bad unit	Boolean	False	-	-
186	Skip by checking special	Boolean	False	-	-

	pattern					
187	Skip Die by mark check	Boolean	False	-	-	
188	Check Max skip die	1 byte int (s)	0	1	0	
189	Max number of continuous skip die	2 byte int (US)	30	1000	2	
190	Skip Non stick die	Boolean	False	-	-	
191	NSOP Skip Die	Boolean	False	-	-	
192	NSOL Skip Die	Boolean	False	-	-	
193	Missing ball skip die	Boolean	False	-	-	
195	VLL Lead Width Tol	2 byte int(s)	0	3000	0	
196	VLL Search Delay	2 byte int (s)	0	200	0	
197	WCL Open Force	2 byte int(s)	50	500	0	
198	WCL Close Force	2 byte int(s)	40	500	0	
201	Torch level	2-byte int(s)	0	50	-670	
202	Program valid flag	irebon	False	-	-	
203	Number of wire	2-byte int(s)	0	2000	0	
204	Number of reference	2-byte int(s)	0	2000	0	
205	Feed power	1-byte int(us)	10	255	0	
206	EFO delay	2-byte int(s)	3	32000	0	
207	Standby power 1	1-byte int(us)	33	255	0	
208	Fire Level Factor	2-byte int(s)	10	100	0	
209	Tail length	1-byte int(s)	35	100	0	
210	PR Tolerance Retry	2-byte int(s)	50	1000	0	
211	Head time	2-byte int(s)	10	30	0	
214	Contact force 1	2-byte int(s)	0	1000	0	
215	Contact force 2	2-byte int(s)	150	1000	0	
216	Power delay	1-byte int(us)	0	50	0	
217	Standby power 2	2-byte int(s)	2	1000	0	
218	Power delay 2	1-byte int(us)	90	255	0	
219	false alarm count	4-byte int(us)	0	999999	0	
221	Ball size – EFO Parameter	2-byte int(s)	30	100	0	
222	Fire Level – EFO Parameter	2-byte int(s)	25	1000	-1000	
223	Bond Time Unit Factor	2-byte int(s)	0	1	0	0: 0.5ms 1: 1ms
224	EFO FAB Block	2 byte int(s)	0	1	0	
225	EFO Current	4 byte int(s)	3250	40000	1000	
226	Wire Size – EFO Parameter	2-byte int(s)	10	250	5	
227	FAB size – EFO Parameter	2-byte int(s)	24	255	1	
229	Ball Thickness- EFO parameter	2-byte int(s)	0	1000	0	
230	Heater Value	4-byte int(s)	0	1000	0	
231	Pre-Heater Value	4-byte int(s)	0	1000	0	
232	Post Bond Heater Value	4-byte int(s)	0	1000	0	
233	Fork Heater	4-byte int(s)	0	1000	0	
234	Over Hang Heater Value	4-byte int(s)	0	1000	0	
235	Heater2 Value	4-byte int(s)	0	1000	0	
236	Contact Time 1	1-byte int(us)	0	255	0	
237	Contact Time 2	1 byte int(us)	0	255	0	
238	Contact Power 1	2 byte int (s)	0	255	0	
239	Contact Power 2	2 byte int (s)	0	255	0	
241	Die alignment tolerance	2-byte int(s)	5	255	0	
242	Lead alignment tolerance	2-byte int(s)	50	255	0	

243	Bond tip offset in x	4-byte float	0	32767	-32767	
244	Bond tip offset in y	4-byte float	0	32767	-32767	
245	Power factor	2-byte int(s)	10	50	0	
246	Force factor	2-byte int(s)	0	5	0	
247	Preheat time	1-byte int(us)	0	50	0	
248	Bond Time 1	2-byte int(s)	22	255	0	
249	Bond Time 2	2-byte int(s)	15	255	0	
250	Bond Force 1	2-byte int(s)	45	255	0	
251	Bond Force 2	2-byte int(s)	110	255	0	
252	Bond Power 1	2-byte int(s)	70	255	0	
253	Bond Power 2	2-byte int(s)	100	255	0	
254	Loop base	2-byte int(s)	40	100	10	
255	Lead tool offset	2-byte int(s)	0	10	0	
256	Number of rows	1-byte int(s)	1	127	0	-
257	Number of columns	1-byte int(s)	1	127	0	-
258	Step & repeat flag	1-byte int(s)	0	6	0	0: None (Single Unit) 1: Matrix 2: Hybrid Forward 3: Hybrid Reverse 4: PR Look Ahead 5: Hybrid Rev Matrix 6: Ahead Matrix
259	PR search delay for die	2-byte int(s)	30	32000	0	Unit Factor 0.5ms
260	PR search delay for lead	2-byte int(s)	20	32000	0	Unit Factor 0.5ms
261	Device Name	8-byte ASCII	"ADEFAULT"	-	-	-
265	Operator ID	32 bytes ASCII	-	-	-	
270	Power Factor for 1st bond	2-byte int (US)	0	255	0	
271	Force Factor1 for 1st bond	2-byte int (US)	0	255	0	
272	SIM Option	2-byte int (US)	0	1	0	
273	Pre-Heat All unit	1 byte int (US)	1	1	0	
274	EFO Current FAB BLK #0	2 byte int (US)	3250	40000	1000	
275	EFO Current FAB BLK #1	2 byte int (US)	3250	40000	1000	
276	EFO-Wire Type	1-byte int(s)	0	1	0	
277	EFO control Mode	1-byte int(s)	0	1	0	
278	EFO Enable Dual FAB	1-byte int(s)	0	1	0	
279	Auto Calc EFO Time	1-byte int(s)	1	1	0	
281	Number of unit bonded	8-byte float	0	9999999999	0	-
282	Number of wire bonded	8-byte float	0	9999999999	0	-
283	Capillary count	8-byte float	0	9999999999	0	-
284	Number of ball missing	4-byte int(us)	0	99999999	0	-
285	Wire turn count	4-byte int(us)	0	99999999	0	-
286	Number of die skipped	4-byte int(us)	0	99999999	0	-
287	Number of time encounter skip die	4-byte int(us)	0	99999999	0	-
288	Number of lead quality fail	4-byte int(us)	0	99999999	0	-
289	Number of lead tolerance fail	4-byte int(us)	0	99999999	0	-
290	Number of die quality fail	4-byte int(us)	0	99999999	0	-

291	Number of die tolerance fail	4-byte int(us)	0	99999999	0	-
292	Number of local lead fail	4-byte int(us)	0	99999999	0	-
293	Number of ejector error	4-byte int(us)	0	99999999	0	-
294	Number of output magazine fail	4-byte int(us)	0	99999999	0	-
295	Number of index error	4-byte int(us)	0	99999999	0	-
296	Number of window clamp error	4-byte int(us)	0	99999999	0	-
297	Number of 1st non-stick error	4-byte int(us)	0	99999999	0	
298	Number of 2nd non-stick error	4-byte int(us)	0	99999999	0	
299	Total Productive Time	4-byte int(us)	0	99999999	0	
300	Ideal UPH	4-byte f	0	99999999	0	
301	Net UPH	4-byte f	0	99999999	0	
303	MTBA	4-byte int(us)	0	99999999	0	Seconds
304	Number of Assists	4-byte int(us)	0	99999999	0	Seconds
305	MTBF	4-byte int(us)	0	99999999	0	Seconds
306	Number of Failures	4-byte int(us)	0	99999999	0	Seconds
307	Total Idle Time	4-byte int(us)	0	99999999	0	Seconds
308	Productive Time	4-byte int(us)	0	99999999	0	Seconds
309	Down Time	4-byte int(us)	0	99999999	0	Seconds
310	Up Time	4-byte int(us)	0	99999999	0	Seconds
311	Number of Skip ink die	4 byte int(us)	0	99999999	0	
312	Number of skip missing die	4 byte int(us)	0	99999999	0	
313	Number wire clamp errors	8 byte Float	0	999999999	0	
314	Number of Etorch errors	8 byte Float	0	999999999	0	
315	Number of Tail Short	4 byte int(us)	0	99999999	0	
316	Number of Index PR Fail	4 byte int(us)	0	99999999	0	
317	Number of Index Out of Tiebar	4 byte int(us)	0	99999999	0	
318	Number of Wire End	4 byte int(us)	0	99999999	0	
320	Processed Quantity	4-byte int(us)	0	99999999	0	
321	Rejected Quantity	4-byte int(us)	0	99999999	0	
322	Yield	4-byte float	0	100	0	%
323	E10-96 Equipment State	2-byte int(us)	1	6	1	1 Productive 2 Standby 3 Engineering 4 Sched Down 5 Unsh Down 6 NonSch Down
331	Param range – Time Base1 – Max value	2 byte int (s)	0	255	0	
332	Param range – Time Base1 – Min value	2 byte int (s)	0	255	0	
333	Param range – Time Base2 – Max value	2 byte int (s)	0	255	0	
334	Param range – Time Base2 – Min value	2 byte int (s)	0	255	0	
335	Param range – Power Base1 – Max value	2 byte int (s)	0	255	0	
336	Param range – Power	2 byte int (s)	0	255	0	

	Base1 – Min value					
337	Param range – Power Base2 – Max value	2 byte int (s)	0	255	0	
338	Param range – Power Base2 – Min value	2 byte int (s)	0	255	0	
339	Param range – Force Base1 – Max value	2 byte int (s)	0	255	0	
340	Param range – force Base1 – Min value	2 byte int (s)	0	255	0	
341	Param range – Force Base2 – Max value	2 byte int (s)	0	255	0	
342	Param range – force Base2 – Min value	2 byte int (s)	0	255	0	
361	1st Bond Scrub Control	2 byte int(s)	0	1	0	
362	1st Bond Scrub Direction	2 byte int(s)	0	9	0	
363	1st Bond Scrub Force	2 byte int(s)	9	1000	0	
364	1st Bond Scrub Power	2 byte int(s)	30	255	0	
365	2nd Bond Scrub Control	2 byte int(s)	0	8	0	
366	2nd Bond Scrub Direction	2 byte int(s)	0	3	0	
367	2nd Bond Scrub Force	2 byte int(s)	0	1000	0	
368	2nd Bond Scrub Contact Power	2 byte int(s)	0	255	0	
372	ECChangeStatus	2-byte int(us)	-	-	-	ECID change status
373	Set1 Impedance	4-byte f	0	99999999	0	
374	Set2 Impedance	4-byte f	0	99999999	0	
375	Set1 Bond Freq	2-byte int(us)	0	3	0	
376	Set2 Bond Freq	2-byte int(us)	0	3	0	
381	PR- Coaxial Light	2-byte int(s)	0	255	0	
382	PR-Side Light	2-byte int(s)	0	255	0	
383	Panel Coaxial Light	2-byte int(us)	0	255	0	
384	Panel Side Light	2-byte int(us)	0	255	0	
385	PR-Low Mag Coaxial Light	2-byte int(s)	0	255	0	
386	PR-Low Mag Side Light	2-byte int(s)	0	255	0	
387	Coaxial Blue Light-Hi Mag	2-byte int(s)	0	255	0	
388	Coaxial Blue Light-Low Mag	2-byte int(s)	0	255	0	
470	Defect Out Strip ID	32 byte ASCII	-	-	-	(For ASM IDEALine)
471	Defect out strip status	2 byte int(s)	-	-	-	(For ASM IDEALine)
472	Defect Out Count	2 byte int(us)	-	-	-	(For ASM IDEALine)
473	Defect out Last block	Boolean	FALSE	-	-	(For ASM IDEALine)
474	Defect out X1	2 byte int (us)	-	-	-	(For ASM IDEALine)
475	Defect Out Y1	2 byte int (us)	-	-	-	(For ASM IDEALine)
476	Defect Out Code1	2 byte int(s)	-	-	-	(For ASM IDEALine)
477	Defect out X2	2 byte int (us)	-	-	-	(For ASM IDEALine)
478	Defect Out Y2	2 byte int (us)	-	-	-	(For ASM IDEALine)
479	Defect Out Code2	2 byte int(s)	-	-	-	(For ASM IDEALine)
480	Defect out X3	2 byte int (us)	-	-	-	(For ASM IDEALine)
481	Defect Out Y3	2 byte int (us)	-	-	-	(For ASM IDEALine)
482	Defect Out Code3	2 byte int(s)	-	-	-	(For ASM IDEALine)
483	Defect out X4	2 byte int (us)	-	-	-	(For ASM IDEALine)
484	Defect Out Y4	2 byte int (us)	-	-	-	(For ASM IDEALine)
485	Defect Out Code4	2 byte int(s)	-	-	-	(For ASM IDEALine)
486	Defect out X5	2 byte int (us)	-	-	-	(For ASM IDEALine)
487	Defect Out Y5	2 byte int (us)	-	-	-	(For ASM IDEALine)

488	Defect Out Code5	2 byte int(s)	-	-	-	(For ASM IDEALine)
489	Defect out X6	2 byte int (us)	-	-	-	(For ASM IDEALine)
490	Defect Out Y6	2 byte int (us)	-	-	-	(For ASM IDEALine)
491	Defect Out Code6	2 byte int(s)	-	-	-	(For ASM IDEALine)
492	Defect out X7	2 byte int (us)	-	-	-	(For ASM IDEALine)
493	Defect Out Y7	2 byte int (us)	-	-	-	(For ASM IDEALine)
494	Defect Out Code7	2 byte int(s)	-	-	-	(For ASM IDEALine)
495	Defect out X8	2 byte int (us)	-	-	-	(For ASM IDEALine)
496	Defect Out Y8	2 byte int (us)	-	-	-	(For ASM IDEALine)
497	Defect Out Code8	2 byte int(s)	-	-	-	(For ASM IDEALine)
498	Defect out X9	2 byte int (us)	-	-	-	(For ASM IDEALine)
499	Defect Out Y9	2 byte int (us)	-	-	-	(For ASM IDEALine)
500	Defect Out Code9	2 byte int(s)	-	-	-	(For ASM IDEALine)
501	Defect out X10	2 byte int (us)	-	-	-	(For ASM IDEALine)
502	Defect Out Y10	2 byte int (us)	-	-	-	(For ASM IDEALine)
503	Defect Out Code10	2 byte int(s)	-	-	-	(For ASM IDEALine)
504	Defect out X11	2 byte int (us)	-	-	-	(For ASM IDEALine)
505	Defect Out Y11	2 byte int (us)	-	-	-	(For ASM IDEALine)
506	Defect Out Code11	2 byte int(s)	-	-	-	(For ASM IDEALine)
507	Defect out X12	2 byte int (us)	-	-	-	(For ASM IDEALine)
508	Defect Out Y12	2 byte int (us)	-	-	-	(For ASM IDEALine)
509	Defect Out Code12	2 byte int(s)	-	-	-	(For ASM IDEALine)
510	Defect out X13	2 byte int (us)	-	-	-	(For ASM IDEALine)
511	Defect Out Y13	2 byte int (us)	-	-	-	(For ASM IDEALine)
512	Defect Out Code13	2 byte int(s)	-	-	-	(For ASM IDEALine)
513	Defect out X14	2 byte int (us)	-	-	-	(For ASM IDEALine)
514	Defect Out Y14	2 byte int (us)	-	-	-	(For ASM IDEALine)
515	Defect Out Code14	2 byte int(s)	-	-	-	(For ASM IDEALine)
516	Defect out X15	2 byte int (us)	-	-	-	(For ASM IDEALine)
517	Defect Out Y15	2 byte int (us)	-	-	-	(For ASM IDEALine)
518	Defect Out Code15	2 byte int(s)	-	-	-	(For ASM IDEALine)
519	Defect out X16	2 byte int (us)	-	-	-	(For ASM IDEALine)
520	Defect Out Y16	2 byte int (us)	-	-	-	(For ASM IDEALine)
521	Defect Out Code16	2 byte int(s)	-	-	-	(For ASM IDEALine)
522	Defect out X17	2 byte int (us)	-	-	-	(For ASM IDEALine)
523	Defect Out Y17	2 byte int (us)	-	-	-	(For ASM IDEALine)
524	Defect Out Code17	2 byte int(s)	-	-	-	(For ASM IDEALine)
525	Defect out X18	2 byte int (us)	-	-	-	(For ASM IDEALine)
526	Defect Out Y18	2 byte int (us)	-	-	-	(For ASM IDEALine)
527	Defect Out Code18	2 byte int(s)	-	-	-	(For ASM IDEALine)
528	Defect out X19	2 byte int (us)	-	-	-	(For ASM IDEALine)
529	Defect Out Y19	2 byte int (us)	-	-	-	(For ASM IDEALine)
530	Defect Out Code19	2 byte int(s)	-	-	-	(For ASM IDEALine)
531	Defect out X20	2 byte int (us)	-	-	-	(For ASM IDEALine)
532	Defect Out Y20	2 byte int (us)	-	-	-	(For ASM IDEALine)
533	Defect Out Code20	2 byte int(s)	-	-	-	(For ASM IDEALine)
560	Current Lot Id	32 byte ASCII				
561	Current Lot Product Id	32 Byte ASCII				
562	Current Lot Quantity	4-byte int(us)	0	1410065407	0	
563	Current Lot unit per substrate	2-byte int(us)	0	34463	0	(For ASM IDEALine)
564	Current Lot PPID	32 byte ASCII	-	-	-	(For ASM IDEALine)
565	Current lot Gold Wire	32 byte ASCII	-	-	-	(For ASM IDEALine)

566	Current Lot Capillary	32 byte ASCII	-	-	-	(For ASM IDEALine)
567	Current Lot Window Clamp	32 byte ASCII				(For ASM IDEALine)
568	Current Lot Top Plate	32 byte ASCII	-	-	-	(For ASM IDEALine)
569	Current Lot Status	32 byte ASCII	-	-	-	(For ASM IDEALine)
570	Personnel ID	32 byte ASCII	-	-	-	(For ASM IDEALine)
571	Strip Process Status	Boolean	False	-	-	(For ASM IDEALine)
574	Cell Controller Lot Mgmt Status	1 byte int(us)	0	1	1	0: DISABLE 1: ENABLE
575	Stripmap Option	1 byte int(us)	0	1	1	0: DISABLE 1: ENABLE
576	Recipe Target	2 byte int (us)	0	2	0	0: HDD only 1: RAM only 2: HDD & RAM
577	Host Recipe Extension	1 byte int (us)	0	1	0	0: DISABLE 1: ENABLE
578	Magazine ID	32 byte ASCII	-	-	-	
579	Magazine Activity	32 byte ASCII	-	-	-	
580	Magazine Location	32 byte ASCII	-	-	-	
581	Stripmap Error Registration Option	1 byte US	0	2	0	(For ASM IDEALine) 0: No 1: Manual 2: Auto
700	Loop Mode 0 – Validity	Boolean	False			0-Invalid, 1-Valid
701	Loop Mode 1 – Validity	Boolean	False			0-Invalid, 1-Valid
702	Loop Mode 2 – Validity	Boolean	False			0-Invalid, 1-Valid
703	Loop Mode 3 – Validity	Boolean	False			0-Invalid, 1-Valid
704	Loop Mode 4 – Validity	Boolean	False			0-Invalid, 1-Valid
705	Loop Mode 5 – Validity	Boolean	False			0-Invalid, 1-Valid
706	Loop Mode 6 – Validity	Boolean	False			0-Invalid, 1-Valid
707	Loop Mode 7 – Validity	Boolean	False			0-Invalid, 1-Valid
708	Loop Mode 8 – Validity	Boolean	False			0-Invalid, 1-Valid
709	Loop Mode 9 – Validity	Boolean	False			0-Invalid, 1-Valid
710	Loop Mode 10 – Validity	Boolean	False			0-Invalid, 1-Valid
711	Loop Mode 11 – Validity	Boolean	False			0-Invalid, 1-Valid
712	Loop Mode 12 – Validity	Boolean	False			0-Invalid, 1-Valid
713	Loop Mode 13 – Validity	Boolean	False			0-Invalid, 1-Valid
714	Loop Mode 14 – Validity	Boolean	False			0-Invalid, 1-Valid
715	Loop Mode 15 – Validity	Boolean	False			0-Invalid, 1-Valid
716	Loop Mode 16 – Validity	Boolean	False			0-Invalid, 1-Valid
717	Loop Mode 17 – Validity	Boolean	False			0-Invalid, 1-Valid
718	Loop Mode 18 – Validity	Boolean	False			0-Invalid, 1-Valid
719	Loop Mode 19 – Validity	Boolean	False			0-Invalid, 1-Valid
720	Loop Mode 20 – Validity	Boolean	False			0-Invalid, 1-Valid
721	Loop Mode 21 – Validity	Boolean	False			0-Invalid, 1-Valid
722	Loop Mode 22 – Validity	Boolean	False			0-Invalid, 1-Valid
723	Loop Mode 23 – Validity	Boolean	False			0-Invalid, 1-Valid
724	Loop Mode 24 – Validity	Boolean	False			0-Invalid, 1-Valid
725	Loop Mode 25 – Validity	Boolean	False			0-Invalid, 1-Valid
726	Loop Mode 26 – Validity	Boolean	False			0-Invalid, 1-Valid
727	Loop Mode 27 – Validity	Boolean	False			0-Invalid, 1-Valid
728	Loop Mode 28 – Validity	Boolean	False			0-Invalid, 1-Valid
729	Loop Mode 29 – Validity	Boolean	False			0-Invalid, 1-Valid
730	Loop Mode 30 – Validity	Boolean	False			0-Invalid, 1-Valid

731	Loop Mode 31 – Validity	Boolean	False			0-Invalid, 1-Valid
732	Loop Mode 32 – Validity	Boolean	False			0-Invalid, 1-Valid
733	Loop Mode 33 – Validity	Boolean	False			0-Invalid, 1-Valid
740	ALC- Loop Mode	2-byte int(s)	0	33	0	
741	ALC – Loop Height	2-byte int(s)	75	1500	75	z-step
742	ALC- Reverse Height	2-byte int(s)	6	100	6	z-step
743	ALC- Reverse Distance ratio	2-byte int(s)	0	150	0	
744	ALC- Reverse Distance Angle	2-byte int(s)	0	85	-85	Deg
745	ALC-Loop Height Correction	2-byte int(s)	0	127	-127	z-step
746	ALC-Span Length	2-byte int(s)	10	90	10	%
747	ALC-Span Length Unit	2-byte int(s)	0	100	0	%
748	ALC-Span Angle Correction	2-byte int(s)	0	150	-150	Deg
749	ALC-Span Angle Correction Type	1-byte int(us)	0	1	0	Corr Type
750	ALC-Wire Length Factor	2 bytes int(s)	400	550	400	
751	ALC-Loop Height Scale Factor %	2-byte int(s)	30	200	30	
752	ALC-2ND Kink Smoothness %	2-byte int(s)	0	100	-100	
753	ALC- 2nd Kink Height Factor %	2-byte int(s)	0	100	0	
754	ALC-2nd Kink Bump Factor %	2-byte int(us)	0	100	0	
755	ALC-Slope straightness %	1-byte int(s)	0	100	-127	
756	ALC-Loop Auto Tune Prof	2-byte int(s)	0	3	0	Prof Type
757	ALC-Search Delay Base	2-byte int(s)	0	20	0	ms
758	ALC-Search Delay Slope	2-byte int(s)	0	58	-58	%
759	ALC-Sync_Offset_Base	2-byte int(s)	0	16	-16	xy step
760	ALC-Sync_Offset_Slope	2-byte int(s)	0	50	-50	%
761	ALC-DEC_Sample_OS	2-byte int(s)	20	90	20	DEC
762	ALC-Y-Comp	2-byte int(s)	0	100	0	%
763	ALC-Y-Comp_Pre_delay	2-byte int(s)	0	100	0	ms
764	ALC-Loop_Top_ABV_1st Bond	1-byte int(s)	0	2	0	Abv Type
765	ALC-Loop Top DEC Samples	2-byte int(s)	0	90	0	DEC
766	ALC-Loop Top Sync Offset	2-byte int(s)	0	40	-40	ms
767	ALC-Search Speed-2	2-byte int(s)	0	5	0	Srch Type
781	G84 stripmap – Origin Location	1 byte (us)	3	9	1	(For ASM IDEALine) 1:Upper Right topside 2:Upper Left topside 3:Lower Left topside 4:Lower right topside
782	G84 stripmap – No of rows	2 byte (us)	1	34463	0	(For ASM IDEALine)
783	G84 stripmap – No of columns	2 byte (us)	1	34463	0	(For ASM IDEALine)
784	G84 stripmap – Cell Status	Array of 1 byte (us)	0	2	0	(For ASM IDEALine) 0: Functionally good 1:Unknown CellStatus

						2: Defective Cell
785	G84 stripmap – Defect Code status	Array of 2 byte(us)	0	1000	0	
801	Assembly Stage Notification (ASN) – Username	30 bytes – ASCII	-	-	-	(for ASM internal QA)
802	ASN-UserID	30 bytes ASCII	-	-	-	(for ASM internal QA)
803	ASN-STAGE	30 bytes ASCII	-	-	-	(for ASM internal QA)
804	ASN-SUBSTAGE	30 bytes ASCII	-	-	-	(for ASM internal QA)
805	ASN-ACTIVITY	30 bytes ASCII	-	-	-	(for ASM internal QA)
806	ASN-TIMESTAMP	30 bytes ASCII	-	-	-	(for ASM internal QA)
826	Wirebonder System Software Base Version	16 bytes ASCII	-	-	-	Eg: 09.05.96
827	Software Ref number	16 bytes ASCII	-	-	-	
828	Site in Software Version	16 bytes ASCII	-	-	-	
829	WH software Version	16 bytes ASCII	-	-	-	
830	PR Software version	16 bytes ASCII	-	-	-	
831	BQM Version	16 bytes ASCII	-	-	-	
832	Servo Version	16 bytes ASCII	-	-	-	
833	Heater Version	16 bytes ASCII	-	-	-	
834	Software Date&Time	32 bytes ASCII	-	-	-	
835	Software Rev number	16 bytes ASCII	-	-	-	
836	Equipment System Type	16 bytes ASCII	-	-	-	
840	BSOB Ctrl – Ball Formation Direction	2 byte int (s)	0	2	0	
841	BSOB Ctrl – Ball Thickness	2 bytes int(s)	0	255	0	
842	BSOB Ctrl – Tail Length	2 bytes int(s)	0	255	0	
843	BSOB Ctrl – Time Base1	2 bytes int(s)	0	255	0	
844	BSOB Ctrl – Time Base2	2 bytes int(s)	0	255	0	
845	BSOB Ctrl – Power Base1	2 bytes int(s)	0	255	0	
846	BSOB Ctrl – Power Base2	2 bytes int(s)	0	255	0	
847	BSOB Ctrl – Force Base1	2 bytes int(s)	0	255	0	
848	BSOB Ctrl – Force Base2	2 bytes int(s)	0	255	0	
849	BSOB Ctrl – Standby Power1	2 bytes int(s)	0	255	0	
850	BSOB Ctrl – Standby Power2	2 bytes int(s)	0	255	0	
851	BSOB Ctrl – Contact Time1	2 bytes int(s)	0	255	0	
852	BSOB Ctrl – Contact Time2	2 bytes int(s)	0	255	0	
853	BSOB Ctrl – Contact power1	2 bytes int(s)	0	255	0	
854	BSOB Ctrl – Contact Power2	2 bytes int(s)	0	255	0	
855	BSOB Ctrl – Contact Force1	2 bytes int(s)	0	255	0	
856	BSOB Ctrl – Contact Force2	2 bytes int(s)	0	255	0	
857	Twin Ball BSOB Ctrl – Time Base1	2 bytes int(s)	0	255	0	
858	Twin Ball BSOB Ctrl – Time Base2	2 bytes int(s)	0	255	0	
859	Twin Ball BSOB Ctrl –	2 bytes int(s)	0	255	0	

	Power Base1				
860	Twin Ball BSOB Ctrl – Power Base2	2 bytes int(s)	0	255	0
861	Twin Ball BSOB Ctrl – Force Base1	2 bytes int(s)	0	255	0
862	Twin Ball BSOB Ctrl – Force Base2	2 bytes int(s)	0	255	0
863	BSOB Ctrl – Loop Base	2 byte int (s)	0	100	11
864	BSOB Ctrl – Ball Offset	2 byte int (s)	-500	500	25
866	Contact Search Threshold 1	2 bytes int(us)	0	255	0
867	Contact Search Threshold 2	2 bytes int(us)	0	255	0
868	BSOB Wire Ctrl – Base Time1	2 bytes int(us)	0	80	5
869	BSOB Wire Ctrl – Base Time2	2 bytes int(us)	0	80	5
870	BSOB Wire Ctrl – Base Power1	2 bytes int(us)	0	255	75
871	BSOB Wire Ctrl – Base Power2	2 bytes int(us)	0	255	40
872	BSOB Wire Ctrl – Base Force1	2 bytes int(us)	0	255	30
873	BSOB Wire Ctrl – Base Force2	2 bytes int(us)	0	255	15
874	BSOB Ball Ctrl – Ball Formation Shape	2 bytes int(us)	0	3	0
875	BSOB Wire Ctrl – Search Speed 2	2 bytes int(us)	0	32767	0
882	Scheduled Down Time	8 byte float	0	999999999	0
883	Unscheduled Down Time	8 byte float	0	999999999	0
884	Nonscheduled Down Time	8 byte float	0	999999999	0
885	Standby Time	8 byte float	0	999999999	0
886	Adv Pre-Trigger Mode	2 byte int (s)	0	1	0
887	Rev Height Pre-Trigger Time	2 byte int (s)	0	100	0
888	Rev Dist Pre-Trigger Time	2 byte int (s)	0	100	0
889	Rev HT2 Pre-Trigger Time	2 byte int (s)	0	100	0
890	Rev Dist2 Pre-Trigger time	2 byte int (s)	0	100	0
891	Rev HT3 Pre-trigger Time	2 byte int (s)	0	100	0
892	Rev Dist3 Pre-Trigger Time	2 byte int (s)	0	100	0
893	Bump Pre-trigger Time	2 byte int (s)	0	100	0
894	Twist Fwd Pre-Trigger Time	2 byte int (s)	0	100	0
895	Twist Rev Pre-Trigger Time	2 byte int (s)	0	100	0
896	Loop Top – Pre-Trigger Time	2 byte int (s)	0	100	0
901	Traveling Card ID	32 bytes ASCII	-	-	-
902	Unit ID	32 bytes ASCII	-	-	-
903	Unit Starting Time	32 bytes ASCII	-	-	-
904	Unit Finishing Time	32 bytes ASCII	-	-	-

905	Etorch Information	32 bytes ASCII	-	-	-	
906	Gold Wire Information	32 bytes ASCII	-	-	-	
907	Capillary Information	32 bytes ASCII	-	-	-	
908	Transducer Information	32 bytes ASCII	-	-	-	
909	Unit Status	1 byte int (us)	0	1	0	0: Partially bonded / not processed 1: Fully bonded
911	eCheck by Host	1 byte int (us)	0	1	0	0: No 1: Yes
912	eCheck Sampling unit	2 bytes int (us)	0	30000	0	
913	Allow Bonding upon eCheck	1 byte int (us)	1	2	0	0: No 1: Yes 2: Comparing
914	eCheck Upon pgm modification	1 byte int (us)	0	1	0	0: No 1: Yes
915	Verify eCheck	1 byte int (us)	0	3	0	0: No echeck need 1: eCheck as pgm modified 2: eCheck during bonding 3: Manual eCheck request by user
918	Statistics Last Reset Date	17 bytes ASCII	-	-	-	
919	PBS FB Ball Oversize count	4 bytes int (s)	0	999999999	0	
920	PBS FB Ball Undersize count	4 bytes int (s)	0	999999999	0	
921	PBS FB Missing Bond count	4 bytes int (s)	0	999999999	0	
922	PBS FB Broken Wire count	4 bytes int (s)	0	999999999	0	
923	PBS FB Ball Placement count	4 bytes int (s)	0	999999999	0	
924	PBS FB Ball Offset count	4 bytes int (s)	0	999999999	0	
925	PB Data Last Reset Date	17 bytes ASCII	-	-	-	
926	Wire Base Group List	2 bytes int (s)	0	60	0	
927	Wire Loop Mode List	1 bytes int (s)	0	52	0	
961	Barcode Manual Key-in Item	1 bytes int (us)	0	9	0	
962	Scanned Item	1 bytes int (us)	0	9	0	
967	PBS FB Unit Pass	4 bytes int (s)	0	999999999	0	
970	Capillary Part Number	32 bytes ASCII	-	-	-	
980	Calibration Message	32 bytes ASCII	-	-	-	
981	LF bonded number in lot	2 bytes int (us)	0	34463	0	(For ASM IDEALine)
982	Magazine bonded number in lot	2 bytes int (us)	0	34463	0	(For ASM IDEALine)
984	Processed unit number in lot	4 bytes int (us)	0	999999999	0	(For ASM IDEALine)
985	Process Yield in lot	4 bytes float	0.0	100.0	0.0	(For ASM IDEALine)
986	Input Magazine Activity	1 bytes int (us)	0	1	1	
987	Output Magazine Activity	1 bytes int (us)	0	1	1	
990	Strip Out Status	1 bytes int (us)	0	10	0	
991	Tool Install/Uninstall Message	32 bytes ASCII	-	-	-	
992	User Login Name	32 bytes ASCII	-	-	-	(For ASM IDEALine)

1000	Magazine length	4 bytes int (s)	15000	26500	21500
1001	Magazine Width	4 bytes int (s)	8500	1200	5400
1002	Magazine level	4 bytes int (s)	0	50	40
1003	Magazine Base	4 bytes int (s)	1	10000	1250
1004	Magazine pitch	4 bytes int (s)	30	3000	250
1005	Magazine top	4 bytes int (s)	1	10000	250
1006	LF unit number	2 bytes int (us)	1	300	8
1007	LF width	4 bytes int (s)	787	8500	3500
1008	LF device pitch	4 bytes int (s)	150	6600	2650
1010	LF length	4 bytes int (s)	150	99999	21200
1011	Wire spool radius	4 bytes int (s)	1000	6000	2540
1012	Delta step size	2 bytes int (us)	1	125	10
1013	Indexer offset	2 bytes int (s)	-32000	32000	0
1014	Left indexer offset	2 bytes int (s)	-32670	32670	0
1015	Right indexer offset	2 bytes int (s)	-32670	32670	0
1016	Device name	32 bytes ASCII	-	-	-
1017	Device Scale	2 bytes int (s)	0	1	0
1050	Zoom lens delay	2 bytes int (s)	0	2000	0
1051	Row Column PR	1 bytes int (s)	0	1	0
1052	Local Lead	1 bytes int (s)	0	1	0
1053	Backup PR	1 bytes int (s)	0	4	0
1054	Auto index flag	1 bytes int (s)	0	1	0
1055	Ball detect	1 bytes int (s)	0	1	0
1056	Enable PR	1 bytes int (s)	0	1	0
1057	Number of Lead PR alignment	1 bytes int (s)	0	2	2
1061	Efo unit type	1 bytes int (s)	0	1	1
1062	GAP wide warning voltage	2 bytes int (s)	0	9999	4500
1063	Trajectory profile tune	2 bytes int (s)	0	2	1
1064	Speed percentage	2 bytes int (s)	30	100	30
1065	1st bond release force	2 bytes int (s)	0	1000	0
1066	2nd bond release force	2 bytes int (s)	0	1000	0
1067	1st Wire power offset	2 bytes int (s)	-127	127	0
1068	1st Wire time offset	2 bytes int (s)	0	20	0
1069	1st bond release power	2 bytes int (s)	0	255	0
1070	2nd bond release power	2 bytes int (s)	0	255	0
1071	2nd bond PT offset (BSOB Wire)	2 bytes int (s)	-100	100	28
1072	Lead Search Range	2 bytes int (s)	1	13	7
1073	Die Search Range	2 bytes int (s)	1	13	7
1085	Software Name	32 bytes ASCII	-	-	-
1086	Software File Type	32 bytes ASCII	-	-	-
1087	Software Install Status	32 bytes ASCII	-	-	-
1088	Terminal Message	32 bytes ASCII	-	-	-
1091	EFO Time	4 bytes float	0.100	15.00	0.881
1097	1st bond Amplitude Ratio	2 bytes int (us)	0	100	0
1098	1st bond Y Scrub Delay	2 bytes int (us)	0	1000	0
1106	2nd bond Scrub Z Pos	2 bytes int (us)	0	1000	0
1167	Total Wire Length Limit Setting	8 bytes int (s)	0	99999	0
1168	Capillary Warning Limit Setting	4 bytes int (s)	0	99999	0
1169	Capillary Stop Limit Setting	4 bytes int (s)	0	99999	0

1170	Wire Clamp Stop Limit	4 bytes int (s)	0	99999	0	
1171	Etorch Stop Limit	4 bytes int (s)	0	99999	0	
1172	Missing Ball Stop Limit	4 bytes int (s)	0	99999	0	
1173	Skip Die Total Stop Limit	4 bytes int (s)	0	99999	0	
1174	Skip Die Count Stop Limit	4 bytes int (s)	0	99999	0	
1175	Statistic hit limit event type	2 bytes int (s)	0	999999999	0	1: Wire End Detected. 2: Capillary count hit stop limit. 3: Wire clamp count hit stop limit. 4: Etorch count hit stop limit. 5. Missing ball count hit warn limit. 6. Skip die Total hit stop limit. 7: Skip die count hit stop limit.
1400	UPH Gross(Search Result)	4 bytes f	0	99999999	0	
1402	MTBA(Search Result)	4 bytes f	0	99999999	0	
1403	MTBF(Search Result)	4 bytes f	0	99999999	0	
1404	Production Time(Search Result)	4 bytes f	0	99999999	0	
1405	Idle Time (Search Result)	4 bytes f	0	99999999	0	
1406	Output(Search Result)	4 bytes int (us)	0	99999999	0	
1407	NSOP(Search Result)	4 bytes int (us)	0	99999999	0	
1408	Short Tail(Search Result)	4 bytes int (us)	0	99999999	0	
1409	Lead PR Fail(Search Result)	4 bytes int (us)	0	99999999	0	
1410	Lead PR Tol(Search Result)	4 bytes int (us)	0	99999999	0	
1411	Die PR Fail(Search Result)	4 bytes int (us)	0	99999999	0	
1412	Die PR Tol(Search Result)	4 bytes int (us)	0	99999999	0	
1413	VLL Fail(Search Result)	4 bytes int (us)	0	99999999	0	
1414	NSOL(Search Result)	4 bytes int (us)	0	99999999	0	
1500	Capillary count warning limit	4 bytes int (s)	0	999999999	0	
1501	Capillary count stop limit	4 bytes int (s)	0	999999999	0	

Tips: To Monitor Adv Looping Parameter

The SVIDs 700 to 733 stands for 34 Loop Mode, These SVIDs will return the value either T (TRUE) or F(FALSE). True means that particular Loop mode is active and False means it is not active. Before retrieving these parameter values from bonder, host need to send ECID 399 with ECV=255

To monitor which all are the Adv.Loop mode are active
Step 1: Host need to send S2F15 with ECID = 399 and ECV =255
Then Equipment will acknowledge S2F16 Ack=0 to host
Step 2: Host need to send S1F3 with SVIDs 700 to 733
Then Eqpt will return either 0 or 1 for each SVID 700 to 733, depends on the which all are the loop mode active

For each Loop mode there are lot of Loop parameters and the SVIDs 740 to 767 stands for these parameters. For details of individual parameter please refer the Host manual document SVID section.

To monitor advanced loop parameters of one particular Loop Mode, first host need to send S2F15 with ECID 399 and ECV = Loop Mode (0 to 33). If host receives S2F16 Ack=0 from irebonder, host can send SVIDs 740 to 767 (one or more) to get the updated Loop parameter of that particular Loop Mode

eg:
Host want to monitor Advanced Loop parameters (Loop Height, Reverse Height and Reverse Distance Angle) for Loop mode 5
Step 1: Host need to send S2F15 with ECID = 399 and ECV = 5
Then Equipment will acknowledge S2F16 Ack=0 to host
Step 2: Then Host need to send S1F3 with SVIDs 741, 742 and 744
(SVID 741 – Loop Height, SVID 742 – Reverse Height, SVID 744 – Rev Distance Angle)
Then Eqpt will return S1F4 with values for SVIDs 741, 742 and 744

Next Host want to monitor Advanced Loop parameters (Loop Height, Reverse Height and Reverse Distance Angle) for Loop mode 8
Step 1: Host need to send S2F15 with ECID = 399 and ECV = 8
Then Equipment will acknowledge S2F16 Ack=0 to host
Step 2: Then Host need to send S1F3 with SVIDs 741, 742 and 744
(SVID 741 – Loop Height, SVID 742 – Reverse Height, SVID 744 – Rev Distance Angle)
Then Eqpt will return S1F4 with values for SVIDs 741, 742 and 744

8. REPORT ID

These are default reports defined by ASM. Some of them are linked as event report. User may redefine or delete them if necessary. The wire bonder is able to handle one hundred and one reports with RPTID ranges from 0 to 100. Please note that some are only valid if the Eagle Xtreme is connected to the ASM Cell Controller.

RPTID	Description	VID	Variable name
0	General Equipment Info	0	Equipment Name
		1	Software Revision
1	Control State Change	5	Clock
		6	ControlState
2	Operator Command	16	Operator Command
3	Previous Process State	5	Clock
		7	Previous Process State
4	Current Process State	5	Clock
		8	Process State
5	Pause Reason	15	Pause Reason
6	Alarm Info	13	Alarm ID
		14	Alarm Set
		23	Alarm Clock
		24	Secs_Alarmset
7	Process Program	17	PPChangeName
	Upload/Download info	18	PPChangeStatus
8	Current Process Program Name	564	Current Lot PPID
9	Current Time & Date	5	Clock
10	Spool Info	12	Spool Start Time
11	Spool Info	10	Spool Count Total
12	Spool Info	5	Clock
		9	Spool Count Actual
13	Current Process Program Info	203	Number of wire
		204	Number of reference
		261	Device name
14	Current Process Program's Bond	205	Feed Power
	Power Control	207	Standby Power
		217	Standby Power2
		245	Power Factor
		252	Power Base 1
		253	Power Base 2
15	Current Process Program's Bond	214	Contact Force 1
	Force Control	215	Contact Force 2
		250	Force Base 1
		251	Force Base 2
16	Current Process Program's Bond	216	Power Delay 1
	Time Control	218	Power Delay 2
		246	Force Factor
		248	Time Base 1
		249	Time Base 2
17	Current Process Program's Step	256	Number of Rows
	& Repeat Info	257	Number of Columns
		258	Step & Repeat Flag
18	Current Operator Info	265	Operator ID
		560	Current Lot ID
		561	Current Product ID

RPTID	Description	VID	Variable name
19	Process Statistics	265	Operator ID
		281 – 300	Statistical Information
		306	Number of Failure
		560	Cur Lot ID
20	Lot Transition Information	261	Device Name
		265	Operator ID
		302	ACK Code
		560	Current Lot ID
21	SMI Transition Report	272	SMI OPTION
		560	Current Lot ID
22	Process Info	320	Processed Quantity
		321	Rejected Quantity
		322	Yield
23	Semi E10 Statistics	30	Performance Data Last Reset
		303	MTBA
		304	Num Of Assists
		305	MTBF
		306	Num Of Failures
		307	Total Idle Time
		308	Productive Time
		309	Down Time
24	Equipment Status	323	Equipment State
25	Lot ID	560	Current Lot ID
26	Lot Information	560	Current Lot ID
		564	Current Lot PPID
		569	Current Lot Status
27	Strip Map Defect codes	470 – 533	Strip Map Defect Codes
		571	Strip process status
28	Lot Info & Personal Info	561	Current Lot Product ID
		564	Current Lot PPID
		570	Personal ID
29	Machine ID and PM date	25	Machine ID
		26	Last PM Date
		27	Installation Date
30	Completion of Unit Bonding	281	Number of Unit bonded
35	Soft Start Activation	35	Soft Start Activation
37	Update Lot Mgmt status	574	Cell Controller Lot Mgmt Status
38	Update strip mgmt option	575	Strip map option
		576	Recipe Target
		577	Host Recipe Extension
		581	Stripmap Error Reg Option
40	Last Equipment Constant Changed	372	ECChangeStatus
41	Lead frame Indexing-in	37	Barcode String(2D label)
		470	Defect Out Strip Id(Input Magazine 1D Label)
42	Lead Frame Indexing-out	37	Barcode String(2D label)
		470	Defect Out Strip Id(Output Magazine 1D Label)
43	Magazine Management (For ASM IDEALine only)	560	Lot ID
		578	Magazine ID
		589	Magazine Activity
		580	Magazine Location
45	Lot Start Information	265	Operator ID

		560	Lot ID
		562	Lot Quantity
46	Lot Finish Information	265	Operator ID
		320	Processed Quantity
		321	Rejected Quantity
		560	Lot ID
		562	Lot Quantity
47	Lot Start – TI	265	Operator ID
		560	Lot ID
		562	Lot Quantity
		565	Lot Gold Wire
		566	Lot Capillary
		567	Lot Window Clamp
		568	Lot Top Plate
48	Lot End – TI	265	Operator ID
		560	Lot ID
		562	Lot Quantity
		565	Lot Gold Wire
		566	Lot Capillary
		567	Lot Window Clamp
		568	Lot Top Plate
51	Change in control parameter	195	VLL Lead Width Tolerance
		196	VLL Search Delay
		197	WCL Open Force
		198	WCL Close Force
		241	Die Alignment Tolerance
		242	Lead Alignment Tolerance
		245	Power Factor
		246	Force Factor
		247	Pre Heat Time
		248	Bond Time 1
		249	Bond Time 2
		250	Bond Force 1
		251	Bond Force 2
		252	Bond Power 1
		253	Bond Power 2
		255	Lead Tool Offset
		259	PR Search Delay for Die
		260	PR search delay for Lead
		273	Pre Heat All Units
		866	Contact Search Threshold 1
		867	Contact Search Threshold 2
52	Change in Global Parameter	207	Standby Power 1
		214	Contact Force 1
		215	Contact Force 2
		216	Power Delay 1
		217	Standby Power 2
		218	Power Delay 2
		236	Contact Time1
		237	Contact Time 2
		238	Contact Power 1
		239	Contact Power 2
53	Change in Wire Parameter	36	Type of Wire Param modified
54	Change in EFO Parameter	208	Fire Level Factor
		209	Tail Length

		224	EFO FAB Block number
		226	Wire Size – EFO Parameter
		227	FAB size – EFO parameter
		274	EFO Current FAB Block #0
		275	EFO Current FAB Block #1
		276	EFO – Wire Type
		277	EFO Control Mode
		278	EFO Enable Dual FAB
		279	Auto Calc EFO Time
		1091	EFO Time
56	Change in Capillary	283	Capillary Count
57	Wire Turn count Info	285	Wire Turn Count
58	Barcode Scan	37	Barcode String
		962	Scanned Item
59	Change in REF RECORD	-	
61	Assembly Stage Notification	801	ASN Username
	(ASN)	802	ASN User-ID
	(only for ASM Internal QA)	803	ASN Stage
		804	ASN Substage
		805	ASN Activity
		806	ASN Timestamp
62	ASN – User Name request	802	ASN User-ID
	(only for ASM Internal QA)		
65	Wirebonder Software Revision	826	Software Version
	Information	827	Software Ref Number
		828	Site in Software Version
		829	WH Software version
		830	PR software version
		831	BQM version
		832	Servo Version
		833	Heater Version
		834	Software Date & Time
		835	Software Rev Number
67	G84 – Stripmap data	470	Strip ID (indexing out)
	(only for ASM IDEALine)	571	Strip process status – G84
		781	Origin Location of strip – G84
		782	No of rows in strip – G84
		783	No. of columns in strip – G84
		784	Cell Status in strip – G84
		785	Defect Code – G84
68	Calibration starts	980	Calibration message
69	Calibration finishes	980	Calibration message
71	Barcode – Unit start report	265	Operator ID
		901	Traveling card ID
		902	Unit ID
		903	Unit Starting Time
		906	Gold wire information
		907	Capillary Information
		908	Transducer Information
72	Barcode – Unit finish report	265	Operator ID
		901	Traveling card ID
		902	Unit ID
		904	Unit Finishing Time
		906	Gold wire information
		907	Capillary Information

		908	Transducer Information
		909	Unit Status
73	Manual key in barcode	49	Barcode String
		961	Barcode manual key it item
76	1st Bond Scrub Modification	361	1st Bond Scrub Control
		362	1st Bond Scrub Direction
		363	1st Bond Scrub Force
		364	1st Bond Scrub Power
77	2nd Bond Scrub Modification	365	2nd Bond Scrub Control
		366	2nd Bond Scrub Direction
		367	2nd Bond Scrub Force
		368	2nd Bond Scrub Power
78	BSOB Ctrl Modification	840	BSOB Ctrl- Ball Formation direction
		841	BSOB Ctrl-Ball Thickness
		842	BSOB Ctrl – Tail Length
		843	BSOB Ctrl – Time Base1
		844	BSOB Ctrl – Time Base2
		845	BSOB CTRL – Power Base1
		846	BSOB Ctrl – Power Base2
		847	BSOB Ctrl – Force Base 1
		848	BSOB Ctrl-Force Base2
		849	BSOB Ctrl – Standby Power1
		850	BSOB Ctrl – Standby Power2
		851	BSOB Ctrl – Contact Time 1
		852	BSOB Ctrl – Contact Time 2
		853	BSOB Ctrl - Contact Power1
		854	BSOB Ctrl – Contact Power 2
		855	BSOB Ctrl – Contact Force 1
		856	BSOB Ctrl –Contact Force 2
		863	BSOB Ctrl – Loop Base
		864	BSOB Ctrl – Ball Offset
		868	BSOB Wire- Time Base 1
		869	BSOB Wire- Time Base 2
		870	BSOB Wire- Power Base 1
		871	BSOB Wire- Power Base 2
		872	BSOB Wire- Force Base 1
		873	BSOB Wire- Force Base 2
79	Twin Ball BSOB Ctrl	857	Twin Ball BSOB Ctrl – Time Base1
		858	Twin Ball BSOB Ctrl – Time Base2
		859	Twin Ball BSOB Ctrl –Power Base1
		860	Twin Ball BSOB Ctrl – Power Base 2
		861	Twin Ball BSOB Ctrl – Force Base1
		862	Twin Ball BSOB Ctrl – Force Base2
86	Verify eCheck	915	Verify eCheck
87	eCheck flag status	911	eCheck by Host
		912	eCheck Sampling unit
		913	Allow Bonding upon eCheck
		914	eCheck Upon pgm modification
89	User Login information	897	User Login ID
		898	Login Password
90	Active Group information	564	Current lot PPID
		926	Wire Base Group List
		927	Wire Loop Mode List
91	Bond Parameter Information	926	Wire Base Group
		930	Bond Time1

		931	Bond Time2
		932	Bond Power1
		933	Bond Power2
		934	Bond Force1
		935	Bond Force2
		936	Standby Power 1
		937	Standby Power2
		939	Power Delay 2
		940	Contact Time1
		941	Contact Time2
		942	Contact Power1
		943	Contact Power2
		944	Contact Force1
		945	Contact Force2
92	Loop Parameter Information	927	Wire Loop Mode List
		952	Loop Height
		953	Reserve Height
		954	Reserve Distance
		955	Reserve Distance Angle
		956	Loop Height Correction
		958	Span Length
		959	Span Length Correction
93	Input Magazine Activity	986	Input Magazine Activity
95	Current PPID Dos Name	975	Current PPID Dos Name
96	Output Magazine Activity	987	Output Magazine Activity
104	System Software Status	1085	Software Name
		1086	Software File Type
		1087	Software Install Status
105	Clear Terminal Message	1088	Terminal Message
111	Statistic data hit limit	1175	Statistic data hit limit event type
117	Capillary count hit warning limit	1500	Capillary count warning limit
118	Capillary count hit stop limit	1501	Capillary count stop limit
120	Strip id inform	470	Defect Out Strip ID
121	QA finishing	470	Defect Out Strip ID
122	User Login	992	User Login Name
125	Retrieve Statistic(TI)	1400	UPH Gross(Search result)
		1402	MTBA(Search result)
		1403	MTBF(Search result)
		1404	Production Time(Search result)
		1405	Idle Time(Search result)
		1406	Output(Search result)
		1407	NSOP(Search result)
		1408	Short Tail(Search result)
		1409	Lead PR fail(Search result)
		1410	Lead PR Tol(Search result)
		1411	Die PR fail(Search result)
		1412	Die PR Tol(Search result)
		1413	VLL Fail(Search result)
		1414	NSOL(Search result)
150	Upload 2D label	37	Barcode String(2D label)
		470	Defect Out Strip Id(2D label)
151	Input Magazine Unload	37	Barcode String(1D label)
		470	Defect Out Strip Id(1D label)
		986	Input Magazine Activity
152	Output Magazine Unload	37	Barcode String(1D label)

	470	Defect Out Strip Id(1D label)
	987	Output Magazine Activity

9. EVENT ID

The following list contains of events defined in Eagle Xtreme Gold Wire Bonder and the default reports linked to each of them. User may modify and re-link the reports. Please note that some events are not generated by the equipment but are provided to the Host to query important data.

Event ID	Description	Reports Linked	Trigger
0	Enable SECS or Power On	0	Equipment
1	Equipment OFF-LINE	1	Equipment
2	Operator Command	2	Equipment
3	Previous Process State	3	Host
4	Process Stopped	3,4	Host
5	Process State Change	3, 4, 5	Equipment
6	Alarm Detected	6	Equipment
7	Previous Process State	4	Equipment
8	Process Program Change	7	Equipment
9	Process Program Selected	8	Equipment
10	Current Time	9	Host
20	Statistical Data Collection	19,23	Equipment
22	Equipment State Changed	24	Equipment
23	Lot Info	26	Equipment
25	Defect Out Strip Status	27	Equipment
26	Lot Info	28	Equipment
27	Production info	22	Equipment
28	Inst Date, Machine ID	29	Host
29	Rej quantity	22,26	Host
30	Processed Quantity	22,25	Host
31	Soft Start Timeout	1	Equipment
33	Unit Bond completion	30	Equipment
36	Soft Start Activation	35	Equipment
37	CC Lot Mgmt Status	37	Equipment
38	Strip Mgmt status	38	Equipment
41	Lead frame index-in	41	Equipment
42	Lead frame Index-out	42	Equipment
43	Lot Mgmt Status	43	Equipment
45	Lot Start Information	45	Equipment
46	Lot Finish Information	46	Equipment
47	Lot Start for TI	47	Equipment
48	Lot End for TI	48	Equipment
51	Change in Control Parameter	51	Equipment
52	Change in Global Parameter	52	Equipment
53	Change in Wire Parameter	53	Equipment
54	Change in EFO Parameter	54	Equipment
56	Replace Capillary	56	Equipment
57	Wire usage Info	57	Equipment
58	Auto Save Finished	8	Equipment
59	Change in REF RECORD	59	Equipment
60	Barcode Event	58	Equipment
61	Assembly Stage Notification	61	Equipment
62	ASN – UserID	62	Equipment

65	Wirebonder Software Info	65	Equipment
67	G84- Stripmap Data	67	Equipment
68	Calibration starts	68	Equipment
69	Calibration finishes	69	Equipment
71	Barcode – Unit start event	71	Equipment
72	Barcode – Unit finish event	72	Equipment
73	Barcode manually key in	73	Equipment
76	1st Bond Scrub Modification	76	Equipment
77	2nd Bond Scrub Modification	77	Equipment
78	BSOB Ctrl Modification	78	Equipment
79	Twin Ball BSOB Ctrl Modification	79	Equipment
80	NSOP or NSOL increment	(no RPTID)	Equipment
86	Verify eCheck	86	Equipment
87	eCheck flag Status	87	Equipment
89	User Login information	89	Equipment
90	Active Group information	90	Equipment
91	Bond Parameter Information	91	Equipment
92	Loop Parameter Information	92	Equipment
93	Input Magazine Activity	93	Equipment
95	Current PPID Dos Name	95	Equipment
96	Output Magazine Activity	96	Equipment
97	Lot Ready	8	Equipment
104	Host Install Software Complete	104	Equipment
105	MC Ready To Backup	104	Equipment
106	Host Restore Data	104	Equipment
107	System Shutdown	24	Equipment
108	System Restart	24	Equipment
109	System Initialization	24	Equipment
110	Clear Terminal Message	105	Equipment
111	Statistic data hit limit	111	Equipment
117	Capillary count hit warning limit	117	Equipment
118	Capillary count hit stop limit	118	Equipment
120	Strip id inform	120	Equipment
121	QA Finishing	121	Equipment
122	User Login	122	Equipment
125	Statistic Retrieve(TI)	125	Equipment
150	Leadframe 2D Label Upload	150	Equipment
151	Input Magazine Unload	151	Equipment
152	Output Magazine Unload	152	Equipment

10. REMOTE COMMAND CODE

Remote Commands Recognized (S2,F21)

Remote Command Codes, which are defined in Eagle Xtreme Gold Wire Bonder and used in S2F41 messages, are as follows. These are accepted only if the wire bonder is under remote control state.

Remote Command	Command Code	Valid Processing States
START BOND	13	IDLE, SETUP
STOP BOND	24 or 109	EXECUTING, PAUSE
PAUSE	33 or 117	EXECUTING, PAUSE, IDLE, SETUP
RESUME	35 or 111	IDLE, SETUP
SOFT START	37 or 118	IDLE,SETUP
To Upload wire record data file (WIRE_RCD.DAT) **	50 or 119	IDLE,SETUP,PAUSE (Should have a valid bond program in Memory)
To Upload Bond point coordinate file (BONDPT.TXT) **	51 or 120	IDLE,SETUP,PAUSE (Should have a valid bond program in Memory)
To upload Reference record data file (REF_RCD.DAT) **	52 or 122	IDLE,SETUP,PAUSE (Should have a valid bond program in Memory)
REQ CHANGE LOT (Only for ASM IDEALINE)	100	IDLE, SETUP
Reset Production count (Only for ASM IDEALINE)	106	IDLE,SETUP,PAUSE
Reset EPD (Equipment Performance Data)	107	IDLE,SETUP,PAUSE
Pause after current strip bonding (only for ASM IDEALine)	108	EXECUTING, PAUSE
Pause after current wire bonding (only for ASM IDEALine)	110	EXECUTING, PAUSE
Reset Capillary and False-alarm (only for ASM IDEALINE)	114	IDLE,SETUP,PAUSE
AUTO SAVE BOND PROGRAM	121	IDLE,SETUP,PAUSE
Reset Statistics count	133	IDLE,SETUP,PAUSE
Upload Wire Layout Data	134	IDLE,SETUP,PAUSE
Upload loop data file	135	IDLE,SETUP,PAUSE
Reset PBI data	136	IDLE,SETUP,PAUSE
Reset statistics Tools usage	137	IDLE,SETUP,PAUSE

** Note: When Upload REF_RCD.DAT or WIRE_RCD.DAT or BONDPT.TXT using the corresponding remote command, the file will be uploaded using S7F1, S7F2, S7F3 and S7F4

Host Commands Accepted (S2,F41)

Description	<A Host Command>	<A Parameters>	Valid Processing States
Stop the bonding	"STOP"		EXECUTING, PAUSE
Pause the bonding	"PAUSE"		EXECUTING, PAUSE, IDLE, SETUP
Start the bonding	"START"		IDLE, SETUP
Resume the bonding	"RESUME"		IDLE, SETUP
Soft Start Bonding	"SOFTSTART"		IDLE,SETUP
Change the device	"PP_SELECT"	1. PPID <A "FILENAME.WB"> 2. PROMPT <B00LEAN T/F>	IDLE, SETUP
Enable Lot Mgmt (For IDEALine only)	"ELM"	13. CPNAME <A "CCLOTMANAGEMENT" > CPVALUE <A "ENABLE" / "DISABLE">	IDLE, SETUP
CC Settings (For IDEALine only)	"CCSETTING"	1. CPNAME <A "STRIPMANAGEMENT" > CPVALUE <A "ENABLE" / "DISABLE"> 14. CPNAME <A "RECIPEDESTINATION" > CPVALUE <U1 0, 1 or 2> 0:HDDonly, 1:RAMonly, 2:HDD&RAM 15. CPNAME <A "HOSTRECIPEEXT"> CPVALUE <A "ENABLE" / "DISABLE">	IDLE, SETUP
Magazine Management (For IDEALine only)	"UNLOADMAGAZINE"	1. CPNAME <A "LOCATION" > CPVALUE <A "INPUT" / "OUTPUT"> 2. CPNAME <A "LOTID"> CPVALUE <A 32 bytes> 3. CPNAME <A "STRIPID"> CPVALUE <A 32 bytes> 16. CPNAME <A "CONDITION"> CPVALUE <A 32 bytes">	All States
Reset Lot Information (For IDEALine only)	"RESETLOTINFO"	17. CPNAME <A "LOTID" > CPVALUE <A 32 bytes>	IDLE , SETUP
Assembly Stage Notification (For ASM in-house usage)	"ASN"	1. CPNAME <A "STAGE" > CPVALUE <A > 2. CPNAME <A "SUBSTAGE"> CPVALUE <A > 3. CPNAME <A "ACTIVITY"> CPVALUE <A 32 bytes> 18. CPNAME <A "TIMESTAMP"> CPVALUE <A >	IDLE, SETUP
StripID Verify Command	"STRIPRECIPECOMMAND"	1. CPNAME <A "STRIPID" > CPVALUE <A > 2. CPNAME <A "RECIPEID"> CPVALUE <A >	
Central Password	"APPLYSECURITY"	1. CPNAME <A "FULLPATH" > CPVALUE <A >	Using Share folder download password file from host side
Retrieve Statistic	"COLLECTINFO"		All States
Leadframe Resume	"LF_RESUMECMD"	CPNAME <A "LF_RESUMECMD" > CPVALUE <A "0" / "1">	All States

Magazine Resume	"MAG_RESUMECMD"	CPNAME <A "MAG_RESUMECMD" > CPVALUE <A "0" / "1">	All States

11. ALARM ID

Alarm Code Byte stands for
0- Not Used
1- Personal Safety
2- Equipment Safety
3- Parameter Control warning
4- Parameter Control error
5- Irrecoverable error
6- Equipment Status warning
7- Attention Flag
8- Data Integrity

Alarm ID	Alarm Code Byte	Description
0	0	Net Comm Error;
1	5	Motor board Error;
2	6	Window Clamp Error;
3	6	Kicker Error;
4	6	Ejector Error;
5	6	Indexer Position Error;
6	6	Input Sensor Error;
7	6	Output Sensor Error;
8	3	No Vacuum;
9	4	No Pressure;
10	7	No Leadframe Input;
11	7	Input Magazine Empty;
12	7	No Output mag or mag orient err;
13	7	Platform Full;
14	6	WH Wait For Input;
15	7	WH Input Hit Limit;
16	6	WH Wait For Output;
17	7	WH Output Hit Limit;
21	4	1st PR Not Loaded;
22	4	2nd PR Not Loaded;
23	6	PR Lead Tol Failure;
24	6	PR Die Tol Failure;
25	6	Manual Align Lead Tol Failure;
26	7	Manual Align Die Tol Failure;
27	7	PR Lead Quality Rejected;
28	7	PR Die Quality Rejected;
29	7	This unit not taught;
30	7	Invalid Reference;
31	6	PR Setup Err;
32	6	Index PR not taught;
33	6	Index PR Fail;
34	6	Index PR Not Loaded;
41	6	Contact Search Failed;
42	6	Contact Search 1 Failed;
43	6	Contact Search 2 Failed;
44	7	1st Bond Non Stick;

45	7	2nd Bond Non Stick;
46	7	Tail too Short;
47	7	Missing Ball;
48	6	Double Bond;
49	6	EFO No High Voltage;
50	6	EFO No Current;
51	6	EFO Gap Wide;
52	6	EFO Short Tail;
60	6	Wire End Warning;
61	6	Wire Used Up Replace;
62	6	BND WH Comm Err;
63	6	BND TCH Comm Err;
64	6	BND EFO Comm Err;
65	6	BND buf overflow;
66	6	BND comm number err;
67	6	BND not req msg;
68	6	BND no msg;
69	6	BND invalid Argument;
70	6	BND oper failed;
71	6	BND timeout;
72	6	BND SI Timeout;
73	6	B18 This Unit Is Bonded;
74	6	Missed sync signal;
75	6	Heater Open Circuit;
76	6	B23 Ctact At Srch Ht;
77	6	B24 Check 2nd Ctact Lvl;
78	6	BQM Calibration Failure;
79	6	BQM Timeout;
80	6	Capillary Hit Stop Limit;
81	6	Temperature is not Even;
82	6	No Program exist;
83	6	This Unit Not Taught;
84	6	Warning EFO Gap Wide;
85	6	Warning EFO Short Tail;
86	6	Ball Detect Disabled;
87	6	Stick Detect Disable;
88	6	Ball Non Stick;
89	6	Capillary Hit Warn Limit;
100	6	Other Error;
101	6	Process Program Not Found;
110	6	VLL Pos Out Of Tolerance;
111	6	VLL Load Lead Fail;
112	3	VLL Search Lead Fail;
113	3	VLL Cannot Find Lead Tip;
114	3	VLL Load Angle Too Large;
115	4	VLL No Signal Or Timeout;
116	4	VLL Set Scan Mode Fail;
117	3	VLL Cannot Find Lead Edge;
118	3	VLL Out of Lead Width Tolerance;
119	3	VLL Check Clearance Fail;
120	3	VLL Load Template Out of Range;

121	3	VLL Load Bin Lead Fail;
122	3	VLL Tip Offset Too Large;
123	3	VLL Grab Image Error;
124	3	VLL Unknown Error;
130	3	Postbond success;
131	3	PB FB Ball Oversize;
132	3	PB FB Ball Undersize;
133	3	PB FB Exceed ball Place;
134	3	PB FB missing Bond;
135	7	PB FB Ball Broken Wire;
136	7	PB FB Exceed balls Offset;
137	6	PB FB Exceed balls Distance;
138	6	PB FB Fail 2 Wire Dist;
139	6	PB FB Fail Ball Roundness;
180	6	PB SB Weld Missing Bond;
181	6	PB SB Broken Wire;
182	6	PB SB Fail Weld Clearance;
183	6	PB SB Exceed Weld Width;
184	6	PB SB Weld Accuracy;
185	6	PB SB Weld Width Undersize;
186	6	PB SB Weld Length Oversize;
187	6	PB SB Weld Length Undersize;
188	6	Strip Map Dimension Error;
230	3	PB WT Missing or Sway Wire;
231	6	BFM Error;
232	6	Wire loop error;
233	6	EFO Error;
234	6	Calibration Memory Alloc failure;
235	6	Illegal LF Adjust Error;
236	6	No unit on bond side;
237	6	LF Adjust offset out of range;
238	6	LF Adjust not finished;
239	6	LF Position calculation error;
240	6	Current bond head inactive;
241	6	WH Cover not closed;
242	6	PB Bond Tip offset Error;
260	6	Host Reject LF 2D Label;
261	6	Upload LF 2D Label Fail;
262	6	Read LF 2D Label Fail;
263	6	Host Reject Mag 1D Label;
264	6	Upload Mag 1D Label Fail;
265	6	Read Input Mag 1D Label Fail;
266	6	Read Output Mag 1D Label Fail;

12. PROTOCOL BASED ON SECS I

12.1. Physical Link

Electrical Interface
The connection includes a serial interface according to **EIA** Standard **RS-232-C** for interface Type **E**, full duplex communication.

Connector
The standard 25-pin connector described in **ISO 2110-1980** should be used. The screw locks should be contained.

Signal Pins
Pins on connector have functions and defined in Table 1

Table 1
Signal Connections

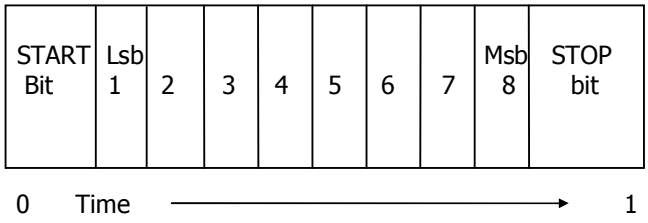
Pin Num	RS-232C Circuit	Description
1	AA	Shield
2	BA	Data From Equipment
3	BB	Data To Equipment
7	AB	Signal Ground

Data rate
The supported data rates on signal pins are 19200, 9600, 4800, 2400, 1200, 300 baud.
NB: Though the AB339E does display higher baud rates (up to 115200) and can communicate at these higher speeds, the maximum throughput is limited by how busy the machine CPU is at the time of communications.

12.2. Character Structure

Data is transmitted or received in a serial bit stream of 10 bits per character at one of the specified data rates.

Start bit 1 bit
Data bit 8 bits (= 1 byte)
Stop bit 1 bit



12.3. Block Transfer Protocol

Handshake Bytes

Name	Code (*b8 b7 .. b1*)	Function
ENQ	00000101	Request to send
EOT	00000100	Ready to receive
ACK	00000110	Correct reception
NAK	00010101	Incorrect reception

Header Structure in a Block

	8	7 6 5 4 3 2 1	
1	**R**	upper device	
2		lower device	
3	**W**	upper message	-- *Stream ID*
4		lower message	-- *Function ID*
5	**E**	upper block num	
6		lower block num	
7		system bytes	
8		system bytes	
9		system bytes	
10		system bytes	

R -bit -- Reverse Bit signifies the direction of a message. The R-bit is set to 0 for messages to the equipment and set to 1 for messages to the host. The R-bit is included in the header so that the direction of the message is contained in every block.

W-bit -- Wait Bit is used to indicate that the sender of a primary message expects a reply. A value of one in W-bit means that a reply is expected. A value of zero in the W-bit means no reply is expected. The W-bit must be set to zero in all secondary messages. For multi-block messages, the sender must ensure that the W-bit is the same in every block of the message. In ASMGEM all primary messages must expect a reply.

E-bit -- End bit is used to determine if a block is the last block of a message. A value of one in the E-bit means that the block is the last block. A value of zero means that more blocks are to follow.

Block Number -- A message sent as more than one block is called a multi-block message. The first block number is given a block number of one, and the block number is incremented by one for each subsequent block until the entire message is sent.
Multi-block message are supported when ASMGEM is receiver or sender.

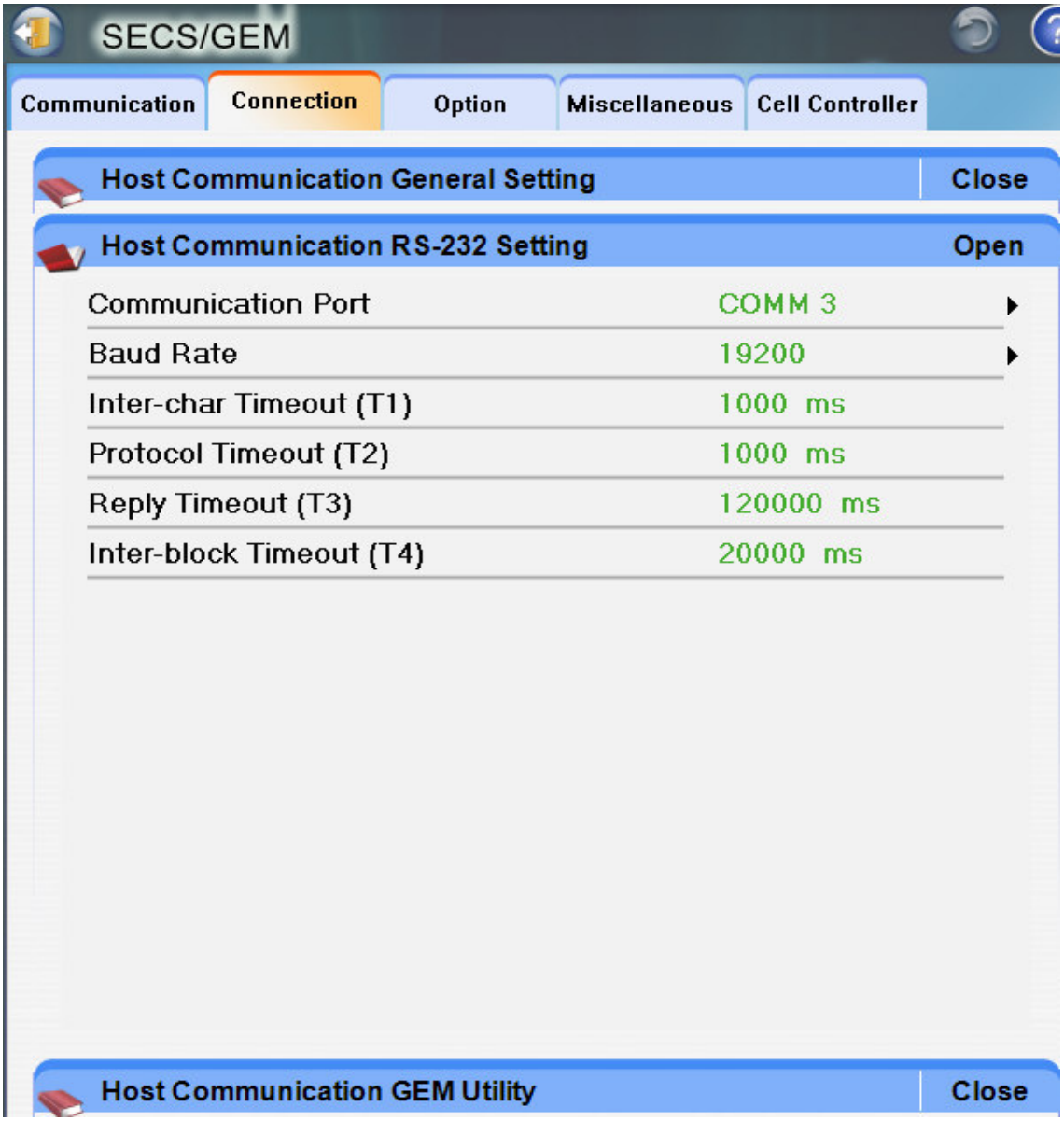
System Byte -- The secondary message is a copy of the data sent from the host. and is transmitted to the host when **W-bit** is 1.

12.4. Message Protocol

Message Blocking
Message blocking is the division of the message data into blocks to be sent to the ***Block Transfer Protocol***. Equipment fill all blocks of a multi-block message, except possibly the last block, with maximum 254 bytes.

Message Length
The maximum data length in a single block of a message is 244 bytes. The maximum number of of block is 256, and so the maximum data length allowed in one message is 244 x 256.

Header
The message protocol must establish the header in each block of the message according to **12.3.**



Protocol Parameters

Symbol	Description	Range	Default
BAUD	Baud Rate	300 –115200	19200
DEVID	Device ID	0 – 32767	0
T1	Inter-character Timeout	0.1 - 10 sec	1 sec
T2	Protocol Timeout	0.2 - 25 sec	3 sec
T3	Reply Timeout	1 – 120 sec	30 sec
T4	Inter-block Timeout	1 – 120 sec	10 sec
RTY	(Retry Limit	1 – 31	3

These parameters can be adjusted on the Eagle Xtreme through the SECS/GEM Configuration page found in the [Configuration] [Factory Automation][SECS/GEM Communication] menu. The changes become effective when the operator exits the Configuration page. Please note that the timing parameters are in ms.

13. GEM Compliance Statement

AB339E GEM Compliance Statement (as at 28 Aug. 00)

GEM COMPLIANCE STATEMENT		
FUNDAMENTAL GEM REQUIREMENTS	**IMPLEMENTED**	**GEM-COMPLIANT**
State Models	☑ Yes ❑ No	☑ Yes
Equipment Processing States	☑ Yes ❑ No	❑ No
Host-Initiated S1=F13/F14 Scenario	☑ Yes ❑ No	
Event Notification	☑ Yes ❑ No	
On-Line Identification	☑ Yes ❑ No	
Error Messages	☑ Yes ❑ No	
Documentation	☑ Yes ❑ No	
Control (Operator Initiated)	☑ Yes ❑ No	
ADDITIONAL CAPABILITIES	**IMPLEMENTED**	**GEM-COMPLIANT**
Establish Communications	☑ Yes ❑ No	☑ Yes ❑ No
Dynamic Event Report Configuration	☑ Yes ❑ No	☑ Yes ❑ No
Variable Data Collection	☑ Yes ❑ No	☑ Yes ❑ No
Trace Data Collection	❑ Yes ☑ No	❑ Yes ☑ No
Status Data Collection	☑ Yes ❑ No	☑ Yes ❑ No
Alarm Management	☑ Yes ❑ No	❑ Yes ☑ No (1)
Remote Control	☑ Yes ❑ No	☑ Yes ❑ No
Equipment Constants	☑ Yes ❑ No	☑ Yes ❑ No
Process Program Management	☑ Yes ❑ No	☑ Yes ❑ No
Material Movement	❑ Yes ☑ No	❑ Yes ☑ No
Equipment Terminal Services	☑ Yes ❑ No	❑ Yes ☑ No (2)
Clock	☑ Yes ❑ No	☑ Yes ❑ No
Limits Monitoring	❑ Yes ☑ No	❑ Yes ☑ No
Spooling	❑ Yes ☑ No	❑ Yes ☑ No
Control (Host-Initiated)	☑ Yes ❑ No	☑ Yes ❑ No

Note:
(1) Events indicating an Alarm Set and Alarm Clear are shared among all alarms. GEM requires a separate event for every alarm.
(2) Only the Single Line terminal service is supported.

Application Notes

Following are some simple tips on configuring the Host software to communicate with the Eagle Xtreme.

Earlier versions of the ASMGEM library (pre June – 2000) did not correctly support the System Bytes. For these versions the Host had to have an option to Ignore System Bytes enabled.

The ASMGEM library used by ASM equipment does not support Message Interleaving or Multiple Open Transactions.

ASMGEM has an internal receive buffer size of 32K. Any multi-block message larger than this will be rejected with S9, F11 Data Too Long (DLN) message. The exception to the rule is during Process Program Upload / Dnload, where a special buffer will be dynamically created to handle the long message (if permission to send has been granted).

If the machine is bonding and receives a request for upload or download, there may be a few seconds pause to the bonding process while the file is compresses / decompressed. The delay will be seen as an inter-wire delay and does NOT affect the bonding quality. If possible, do not perform upload / download operations when the Processing state is EXECUTING.

Note that all the message descriptions are using the SML Notation. This makes it easier to find the data type of the variables in the message and removes the need for a Data Dictionary section in the document.

SML Format

Item Format	SML Item Format Mnemonic
LIST	L[Length]
Binary	B
Boolean	BOOLEAN
ASCII	A[Length]
JIS-8	J[Length]
8-byte integer (signed)	I8
1-byte integer (signed)	I1
2-byte integer (signed)	I2
4-byte integer (signed)	I4
8-byte floating point	F8
4-byte floating point	F4
8-byte integer (unsigned)	U8
1-byte integer (unsigned)	U1
2-byte integer (unsigned)	U2
4-byte integer (unsigned)	U4
* Any Type (Variant)	V

* (V=variant) is not a "true" SML Item Format, but is added to inform that the format for an item is not fixed until the moment the message is sent (eg. Host Command Parameters S2, F41).

For factory floor implementation, linking to a network of Terminal Servers will extend the distance between the Host and the Equipment in a simple and efficient manner. This also removes the WinSECS RS-232 implementation limit of a max baud of 19200. This is because the data is sent over a network to the Terminal Server which can be configured to communicate at speeds up to 115200 baud.

Many Terminal Servers have a Type-Ahead buffer. Ensure this is set to a large value (1024) if communication faults are detected during large messages (especially process program upload / download).

An Example:

WinSECS Settings (Direct Connection)

General Tab

AutoDevice	☑
MonitorEnable	☐
MultipleOpen	☐
SimulateMode	☐
PortType	RS-232
DefaultDeviceID	0

RS-232 Tab

SerialPort	COM1/COM2
AcceptDuplicateBlocks	☐
AutoBaud	☑
IgnoreSystemBytes	☐
Interleave	☐
SecsHost	☑
Baud	19200
RetryLimit	3
T1	1
T2	1
T3	45
T4	10

14. IDEALine Feature – Wire bonder with Cell Controller

14.1. Introduction

In IDEALINE one or more ASM AB339E wire bonder(s) connected and controlled by a Cell Controller. Here Cell Controller is a remote intelligent computer, controlling all the wire bonders. In IDEALINE most of the wire bonder features are automatically controlled by cell controller which include recipe management, statistic management, Alarm management , Lot management , strip management, etc.

Wire bonder can communicate with cell controller either by using RS-232 communication or HSMS (TCP\IP) communication. As soon as the communication enabled with cell controller, the following information will be sent from bonder to cell controller.

1) MDLN : Model No of the equipment. For Eagle Xtreme wire bonder this string will be "Eg XT"
2) SoftRev: Software version of the wire bonder (eg: 090592)
3) Equip ID: The Equipment ID for identification of individual wire bonder
4) Last PM Date: Last Preventive Maintenance Date
5) Installation Date:
6) Operator ID: This is to identify which operator is in-charge of production side.

14.2. Recipe Management

If the wire bonder is working as IDEALINE mode operator can not load the bond program manually using the bonder menu item "Load Bond Program". Because this menu item will be dim as protection. Here Cell Controller will give the recipe name (PPID) to the wire bonder and wire bonder will automatically load that particular recipe from the bonder hard disk to memory. Cell controller can see which are the bond programs (PPIDs) available in the bonder hard disk by directly search. Also it can delete the bond program (PPID) from bonder hard disk.

14.3. Alarm Management

All the errors and warnings happened in wire bonder during different state will be automatically sent to cell controller. During the bonding if any error happens that error should be registered to Cell controller according to that error type.

14.4. Statistic Management

Cell controller can monitor all the statistical from the wire bonder. Statistical data contain the information of UPH, Productive time, Non Scheduled Time, Scheduled Time, Idle Time, Number of good bonded unit, number of rejected unit, etc

Every time wire bonder's state will be reflected to cell controller dynamically. The different states of wire bonder are as follows

1) PRODUCTION
   a) Production (Wire Bonder tower Light is GREEN)
   b) StandBy (Wire Bonder tower Light is YELLOW)
   c) Unscheduled Down (Wire Bonder tower Light is RED)
2) ENGINEERING (Wire Bonder tower Light is YELLOW)
3) SCHEDULED DOWN (Wire Bonder tower Light is YELLOW)

14.5. Stripmap Management

In IDEALINE mode each strip require a strip defect map (PRE-BOND-STRIPMAP) from Cell controller , so that bonder will handle (bond or skip) each subunit according to strip defect code of PRE-BOND-STRIPMAP. Before start bonding of each strip , bonder will verify whether the correct PRE-BOND-STRIPMAP has reached from cell controller for that particular strip. If strip map has not reached from the cell controller, bonder will display a message "Waiting for Strip map from Host" before start bonding and only after receiving the PRE-BOND-STRIPMAP from cell controller boner will start bonding.

The PRE-BOND-STRIPMAP from cell controller contains information of X-coordinate, Y-coordinate and the Error Code for every sub unit of that strip. According to Error Code bonder will take the appropriate action on a particular sub unit.

After completion of the strip bonding , strip map result, i.e POST-BOND-STRIPMAP, will send to the cell controller.

In strip map management boder has mainly 3 types of strip index array

1) **Host Strip Index** : This is the PRE-BOND-STRIPMAP, new strip map received from Cell Controller. This will receive during strip indexing into leadframe track. Just before bonding start, PRE-BOND-STRIPMAP will transfer to "**Bond Strip Index"**

2) **Bond Strip Index** : This contains the stripmap data of currently bonding strip. Once start bonding wirebonder will process each subunit according to PRE-BOND-STRIPMAP data and will update it upon completion of each sub unit. After finish the bonding of a particular strip all information, say POST-BOND-STRIPMAP will be transferred to "**Output Strip Index"** . Also the same information will be transferred to Cell Controller during leadframe index out.

3) **Output Strip Index**: After bonding a strip, all the strip map information (POST-BOND-STRIPMAP) will be passed from Bond Strip Index to Output Strip Index. And POST-BOND-STRIPMAP will be stored here up to the arrival of POST-BOND-STRIPMAP from Bond Strip Index. As the new POST-BOND-STRIPMAP comes from the Bond Strip Index the new one will be stored after erasing old one.

There are two types of strip map format available in ASM Eagle/Eagle60/Eagle Xtreme wire bonder called

a) SEMI G84 stripmap format and
b) ASM stripmap format

This classification is mainly based on the content of stripmap data and method of stripmap data download/upload

14.5.1. SEMI G84 Stripmap Standard

In this method wir bonder will request for PRE-BOND-STRIPMAP data to Cell controller using S14F1 (GAR) during strip indexing into track and Cell Controller will download PRE-BOND-STRIPMAP data using S14F2 (GAD) . This is a single transaction and more faster

Here PRE-BOND-STRIPMAP data contains

- Strip ID
- Strip status (First strip, Intermediate strip, last strip, Test strip, etc)
- Number of Rows and Columns in strip
- Origin Location of first subunit
- Cell status of strip
- Error Code of each subunit
- Whether the Block is last block

The uploading of POST-BOND-STRIPMAP is done by S6F11/S6F12 and it is equipment initiated transaction.
The POST-BOND-STRIPMAP data contains

- Strip ID
- Strip Process status (whether fully bonded or partially bonded)
- Number of Rows and Columns in strip
- Error code for each subunit after process
- Whether the Block is last block

S14, F1 GetAttr Request (GAR) **S, H ⇔ E, reply**

Description : This message is used to request a set of specified attributes for one or more objects. It consists of an "object specifier" for the owner of the target objects (the objects of interest), the target object type, a list of identifiers of the target objects, a filter (a list of qualifying relationships) that limits the target objects of interest to those that meet all of the qualifications in and the specific attributes whose values are requested.

Structure : L, 5

1. <OBJSPEC>
2. <OBJTYPE>
3. L, i i = identifiers of the object instances requested
    1. <OBJID1>
    .
    .
    i. <OBJIDi>
4. L, q q = # object qualifiers to match
    1. L, 3
        1. <ATTRID1>
        2. <ATTRDATA1>
        3. <ATTRRELN1>
    .
    .
    q. L, 3
        1. <ATTRIDq>
        2. <ATTRDATAq>

3. <ATTRRELNq>
5. L, a a = # attributes requested
1. <ATTRID1>
.
.
a. <ATTRIDa>

Exception :
If OBJSPEC is a zero-length item, no object specifier is provided. If i = 0, only the filter is to be applied. If q = 0, no filter is specified. If both i and q = 0, information for all instances of the objects are requested. If a = 0, all attributes are requested.

S14, F2 GetAttr Data (GAD) **S, H ⇔ E, reply**

Description : This message is used to transfer the set of requested attributes of the specified object(s). The order of attributes is retained from the primary message.

Structure : L, 2
1. L, n n = number of objects
1. L, 2
1. <OBJID1>
2. L, a a = number of attributes
1. L, 2
1. <ATTRID1>
2. <ATTRDATA1>
.
.
a. L, 2
1. <ATTRIDa>
2. <ATTRDATAa>
.
.
n. L, 2
1. <OBJIDn>
2. L, b b = number of attributes
1. L, 2
1. <ATTRID1>
2. <ATTRDATA1>
.
.
b. L, 2
1. <ATTRIDb>
2. <ATTRDATAb>
2. L, 2
1. <OBJACK>
2. L, p p = number of errors reports
1. L, 2
1. <ERRCODE1>
2. <ERRTEXT1>
.

.
p.L, 2
1. <ERRCODEp>
2. <ERRTEXTp>

Exception :

If OBJSPEC is a zero-length item, no object specifier is provided. If n = 0, no objects matched the specified filter. If p = 0, no errors were detected.
If n = 0, no objects matched the specified filter. In this case, equipment will automatically create the default stripmap. If p = 0, no errors were detected.

14.5.2. ASM Stripmap Standard

In this method Cell controller will initiate the PRE-BOND-STRIPMAP download with S2F15/S6F16 during strip indexing process and here S2F15/S6F16 is a multi-block transaction. Each block can contain information up to 20 subunits.

The PRE-BOND-STRIPMAP data contains

- Strip ID
- Strip status (First strip, Intermediate strip, last strip, Test strip, etc)
- X-coordinate and Y-coordinate of each subunit
- Error code for each subunit
- Whether the Block is last block

The uploading of POST-BOND-STRIPMAP is done by S6F11/S6F12 and it is equipment initiated transaction.
The POST-BOND-STRIPMAP data contains

- Strip ID
- Strip Process status (whether fully bonded or partially bonded)
- X-coordinate and Y-coordinate of each subunit
- Error code for each subunit after process
- Whether the Block is last block(since data is sending block by block)

14.5.3. Create Default Stripmap

For some customers' requirement, our software supports create default map in WB from Revision 11 Reference 47 onwards. If user enables the flag of "Create default strip map" in menu page [10A12], machine will auto create default map instead of sending S14F1 event to request strip map. The default stripmap data contains:

- Strip ID: Read from 2D camera
- Strip Process status: 0 (fully bonded)
- Original location: 3
- G84 strip map number of row: set same as row number in bond program
- G84 strip map number of column: set same as column number in bond program
- Cell Status: 0
- Defect code: 200

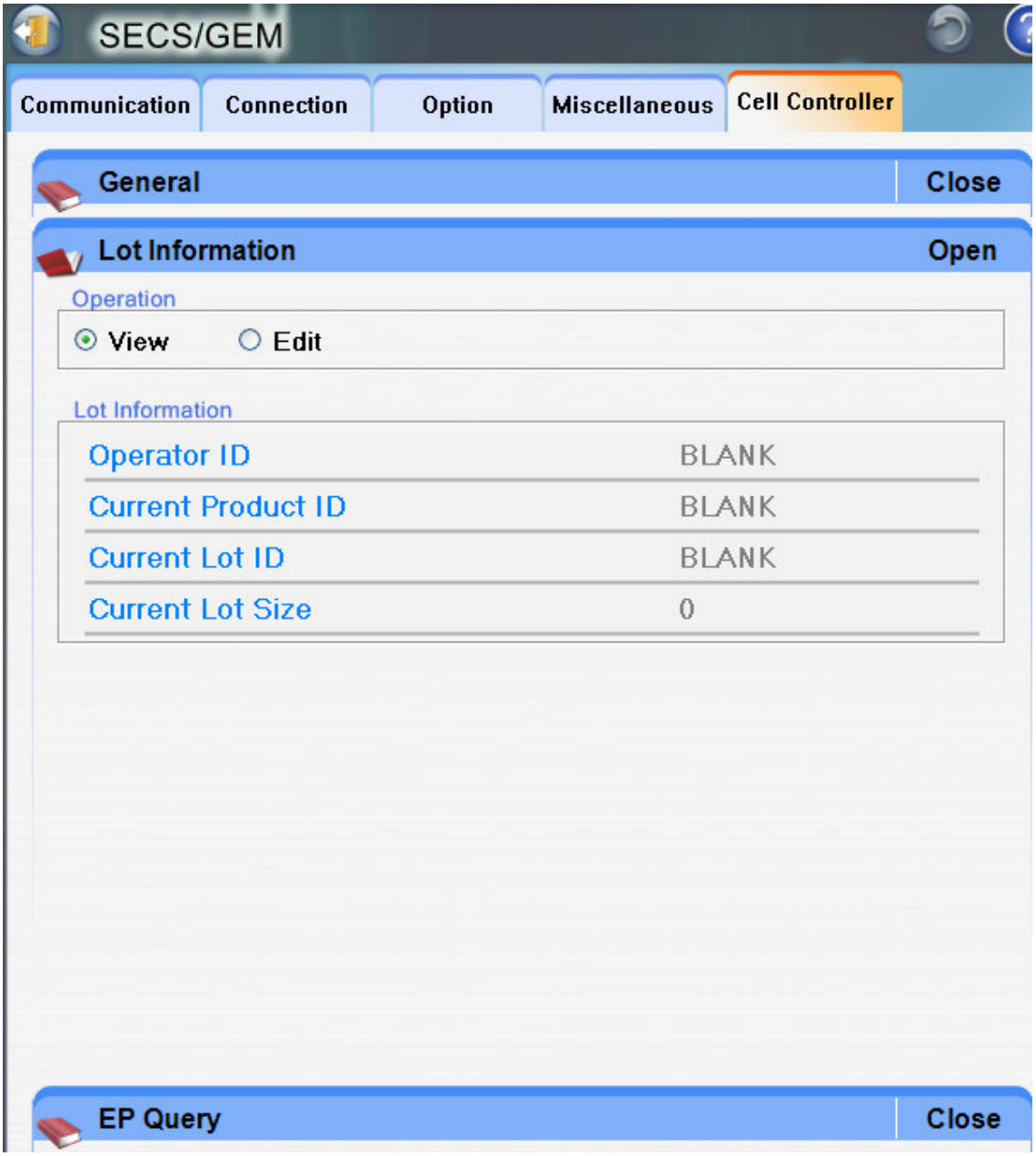
14.6. Lot Management

14.6.1. Lot Management with Cell Controller

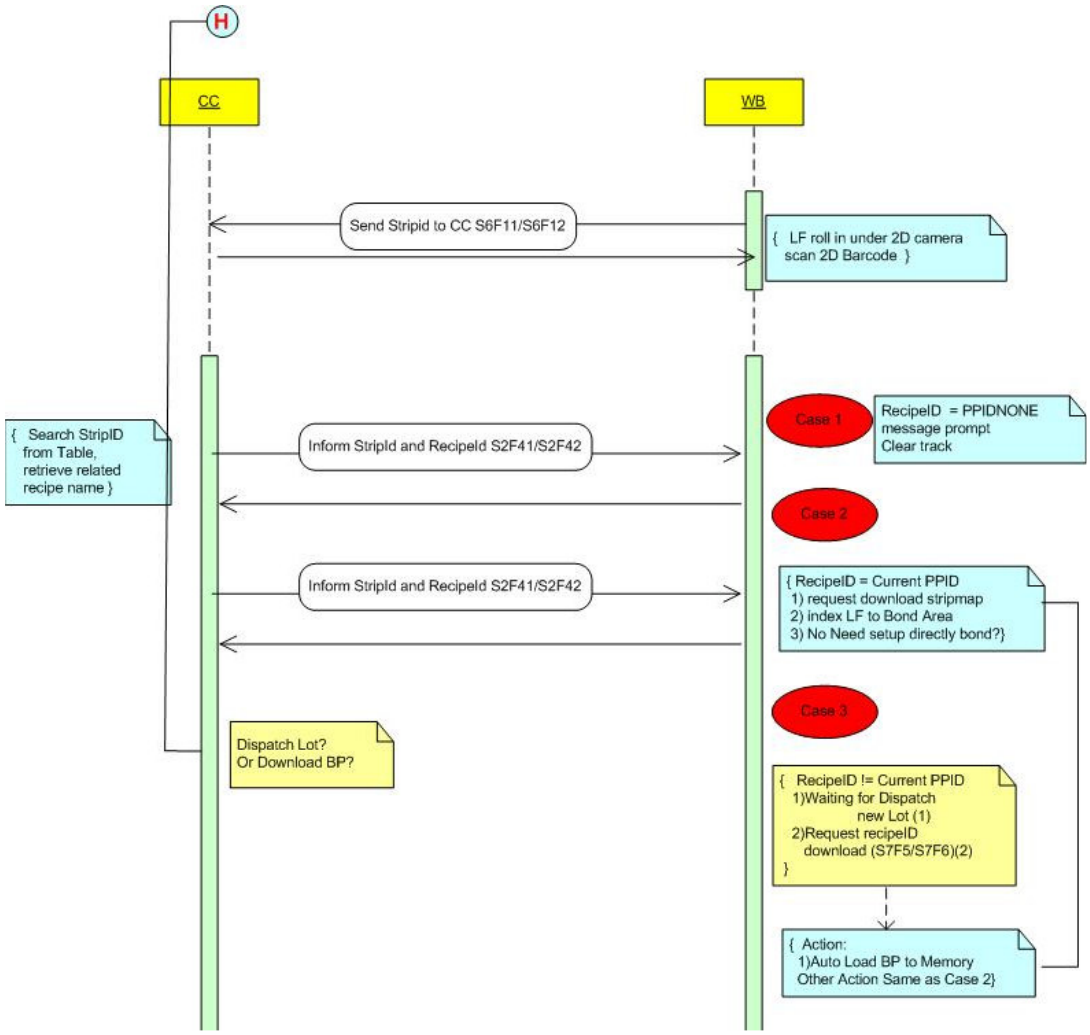
Before starting the production of a new lot, Cell controller will send the Lot Information to the wire bonder. This lot information contains

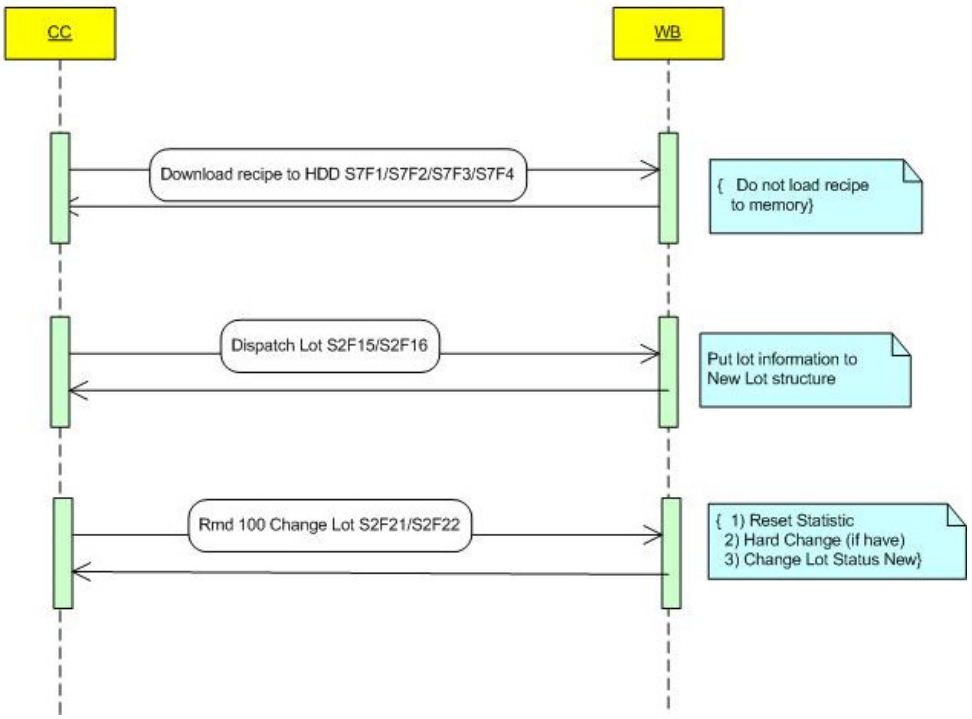
1) Lot ID
2) Product Id
3) Lot quantity
4) Unit per strip
5) Process Program ID
6) Gold Wire Type
7) Capillary Type
8) Window Clamp Type
9) Top Plate Type
10) Number of rows in Strip
11) Number of Columns in Strip

All these lot information will be stored in Wire bonder. Since cell controller can sent the new lot at any time Wire bonder has the capability to store 2 lot arrays, Current lot information and New Lot Information. Current Lot information contains all the information about the current running lot. And New Lot Information contains the information of the latest Lot sent by cell controller. Once a lot starts all the lot information from "New Lot Information" array will be passed to "Current Lot Information" array.



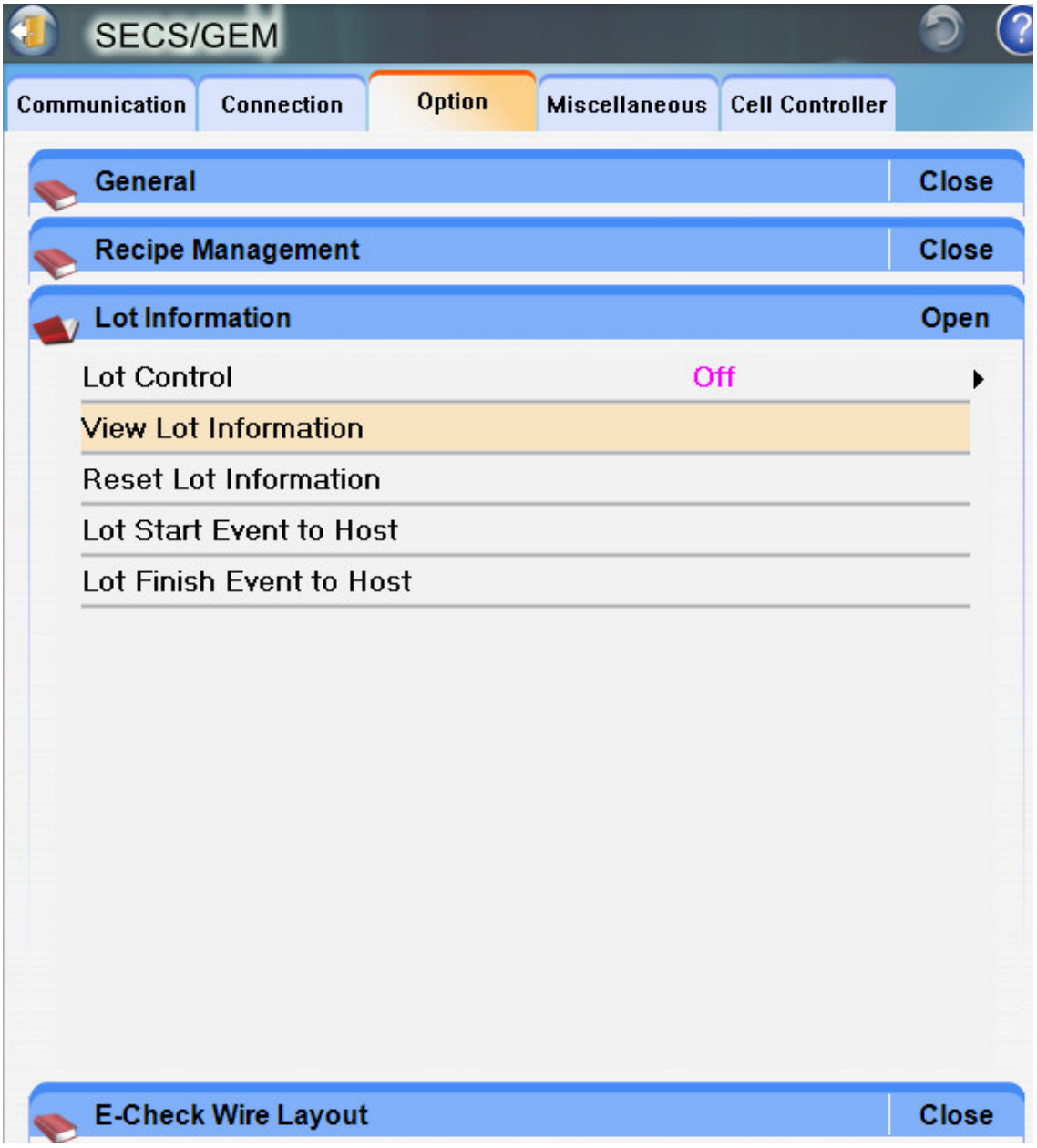
14.6.2. Multi Lot sequence





14.6.3. Advanced Lot Management

As customer required, our software support advanced lot management in Eagle Xtreme. The Lot Management feature is independent from strip mapping, tracking and validation feature. Equipment does not need to enable strip mapping option, to have lot management features functional.
The related menu items locate in menu page [Configuration]->[Factory Automation]->[SECS/GEM]->[Option]->[Lot Information]



A. Item "Lot Control"

-Disabled, machine will be able to operate as per normal.

-Enabled, machine will be able to autobond and single bond **ONLY** when lot information is received.

B. Item "View Lot Information"

-This menu item is used to view current lot information machine is using. The lot information contains:

12) Lot ID
13) PPID
14) Operator ID
15) LF Count

16) Magazine Count
17) Total Unit
18) Processed Unit
19) Num of LF Bonded
20) Num of Magazine Bonded
21) Lot Status

C. Item "Reset Lot Information"

- This menu item is used to reset current lot information to default value.

Note: To allow user to reset current processing lot.

If user enables this feature for advanced lot management, machine will auto report the related events to host during bonding. The related events for lot management is define as followed:
If user enables lot management feature, machine will auto report lot status event to host during bonding cycle. The major activity events are defined per below:

A. Lot Start Event:

CEID = 45
RPTID = 45
SVID = 265 (Operator ID)
SVID = 560 (Current Lot ID)
SVID = 562 (Current Lot Size, No of Unit in Lot)
SVID = 564 (Current Lot PPID)
SVID = 582 (Current Lot LF Count)
SVID = 583 (Current Lot Magazine Count)

Note: This equipment will report to host during the 1st Lead Frame of current lot index into the work holder.

B. Lot End Event:

CEID = 46
RPTID = 46
SVID = 265 (Operator ID)
SVID = 560 (Current Lot ID)
SVID = 562 (Current Lot Size, No of Unit in Lot)
SVID = 564 (Current Lot PPID)
SVID = 582 (Current Lot LF Count)
SVID = 583 (Current Lot Magazine Count)
SVID = 984 (Processed Unit Num in Current Lot)
SVID = 985 (Processed yield in Current Lot)

Note: This event will report to host during the last LF of current lot index out to the output magazine. After sending this event, machine will stop bonding until next lot information is received.

C. LF Index-In Event:

CEID = 41
RPTID = 41
SVID = 470 (Strip ID. Without turning on 2D code feature, this SVID should be default value "BLANK")
SVID = 981 (Index of Bonding LF)
SVID = 982 (Index of Bonding Magazine)

Note: This event will report to host during at every LF index into work holder.

D. LF Index-Out Event:

CEID = 42

RPTID = 42

SVID = 470 (Strip ID. Without turning on 2D code feature, this SVID should be default value "BLANK")

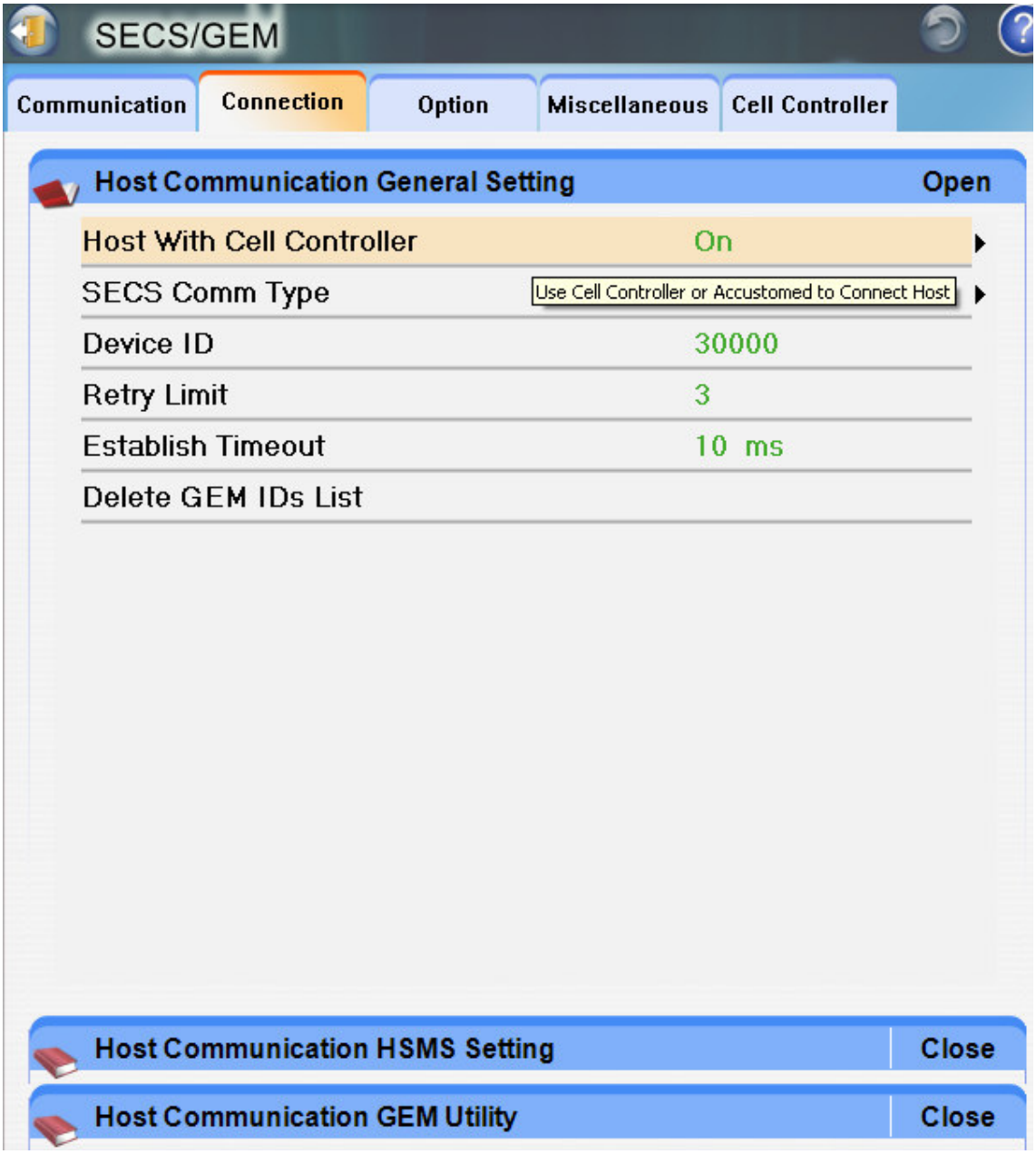
SVID = 981 (Index of Bonded LF)

SVID = 982 (Index of Bonded Magazine)

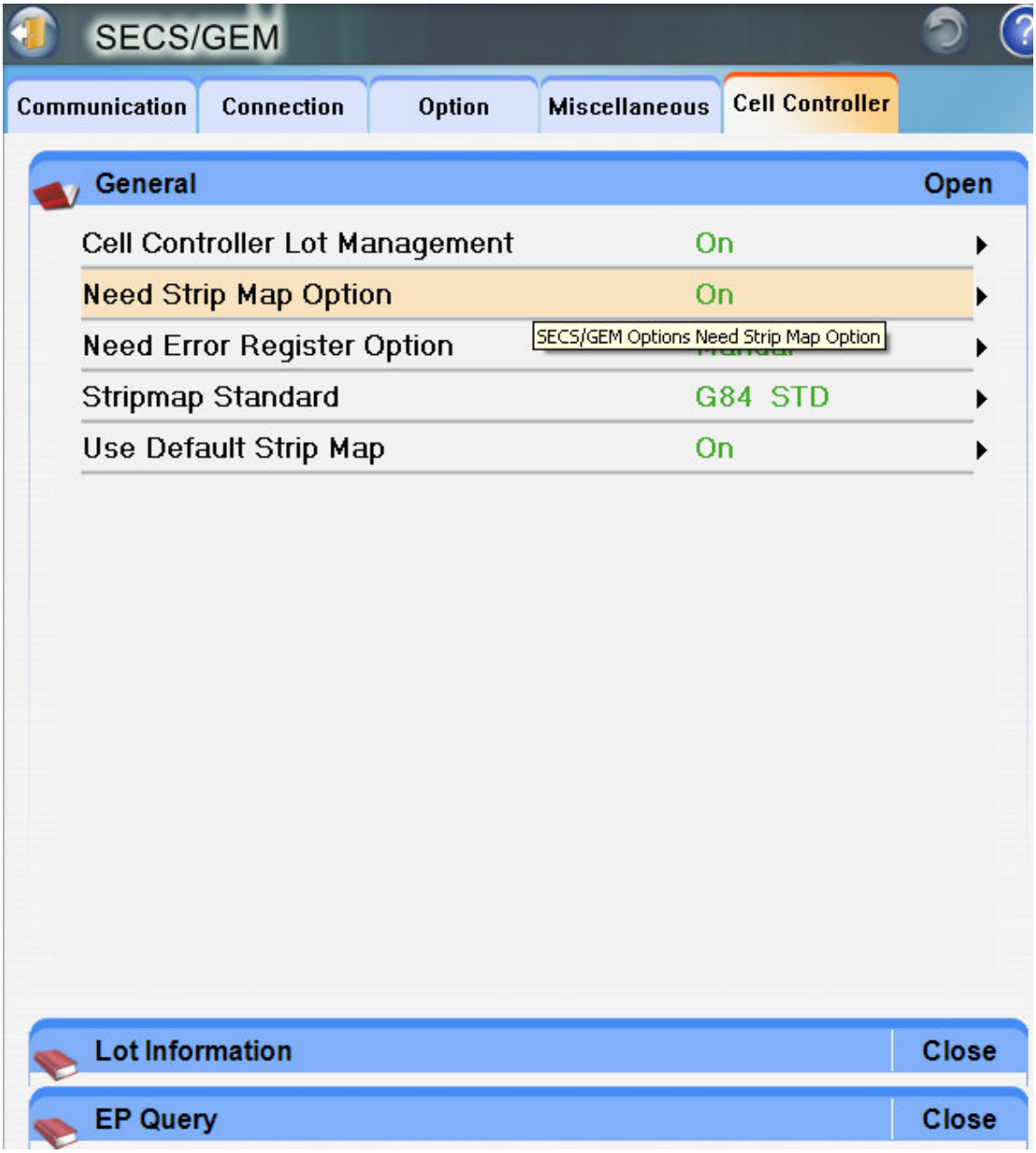
Note: This event will report to host at every LF index out to output magazine.

14.7. Wire Bonder S/W settings for Cell Controller

For a proper communication with cell controller the wire bonder menu settings should be correct.



0. **Host With Cell Controller:** This menu item should be On for IDEALINE setup.



1. **Cell Controller Lot Mgt:** This menu item should be YES for IDEALINE setup.
2. **Need Strip Map Option? :** In IDEALINE mode production can be done with or without Strip Map feature. If the customer is using Strip map management in IDEALINE mode, this menu item should be YES, otherwise NO.
3. **Lot Information :** Lot information menu page contains "Current Lot Information" and "New Lot Information" This menu page is just for getting information about the Current Lot and New Lot information sent from cell controller


SECS/GEM
Communication
Connection
Option
Miscellaneous
Cell Controller
General
Close
Lot Information
Open
Operation
View
Edit
Lot Information
Operator ID
BLANK
Current Product ID
BLANK
Current Lot ID
BLANK
Current Lot Size
0
EP Query
Close

15. HSMS (TCP\IP) Communication

15.1.Introduction

High Speed SECS Message Services (HSMS) defines a communication interface suitable for exchange of messages or information between computers using TCP\IP environment which provides reliable two way simultaneous transmission of streams of contiguous bytes. In the semiconductor industry HSMS communication is useful for the exchange of information between remote computer and the semiconductor equipment using TCP\IP. Where TCP\IP (Transmission Control Protocol \ Internet protocol) is a method of communication which provides reliable, connection oriented message exchange between computers within a Network.

HSMS is an alternative to SEMI E4 (SECS-1) for applications where higher speed communication is needed, instead of simple point-to-point topology. It is also an alternative to SEMI E13 (SECS Message Services) for application where TCP\IP is preferred over OSI.

15.2. Comparison between SECS-1 and HSMS

Feature	SECS-1	HSMS
Communication Protocol Base	RS-232	TCP\IP
Physical Layer	25 pin connector and 4 wire serial cable	Physical Layer not defined. HSMS allow any TCP\IP supported medium.
Communication speed	Typically about 1000bytes/seconds (assuming 9600 baud). Max Baud is 115200	Typically 10Mbits/second (assuming typical Ethernet)
Connections	One physical RS-232 cable per SECS-1 connection	One physical network can support many HSMS connections
Message Format	Message text is SECS-II Data items Transmits a SECS-II message as a series of transmittal blocks each approximately 256 bytes in size. Each block has a one byte block length, a ten byte block header, text, and a two bytes checksum	Message text is SECS-II Data items Transmits a SECS-II message as a TCP\IP byte stream. The message has a four bytes message length, ten bytes message header, and text. The TCP\IP layer may impose blocking limits which depends on the physical layer used. But this blocking is transparent to the TCP\IP API and is outside the scope of HSMS.
Header	Ten byte header on each block of the message. Header bytes 4-5 contains E-bit and block number	One ten byte header for entire message. Header byte 4-5 contain P Type and S Type. Header byte 2-3 are W-Bit, stream, and functions when Stype=0 (Data message). For S Type not equal to 0(Control message), bytes 2-3 have other uses. No R-bit
Maximum Message size	Limited to approximately 7.9 million bytes (32767 blocks times 244 text byte per block)	Message size is limited by 4 bytes message length. (approximately 4Gbytes). Local implementation of TCP\IP and HSMS may further limit

		this in practice.
Protocol parameter (common)	T3 Reply Timeout Device ID	T3 Reply Timeout Session ID (analogous to Device ID)
Protocol parameter (SECS-I only)	Baud Rate T1 Inter Character timeout T2 Block Protocol Timeout T4 Inter Block Timeout Retry Limit	Not used in HSMS (corresponding issues addressed by TCP\IP layers)
Protocol parameter (HSMS only)	Not need for SECS-I	IP address and Port of PASSIVE entity T5 Connect separation Timeout T6 Control Transact Timeout T7 NOT SELECTED Timeout T8 Network Inter Character Timeout

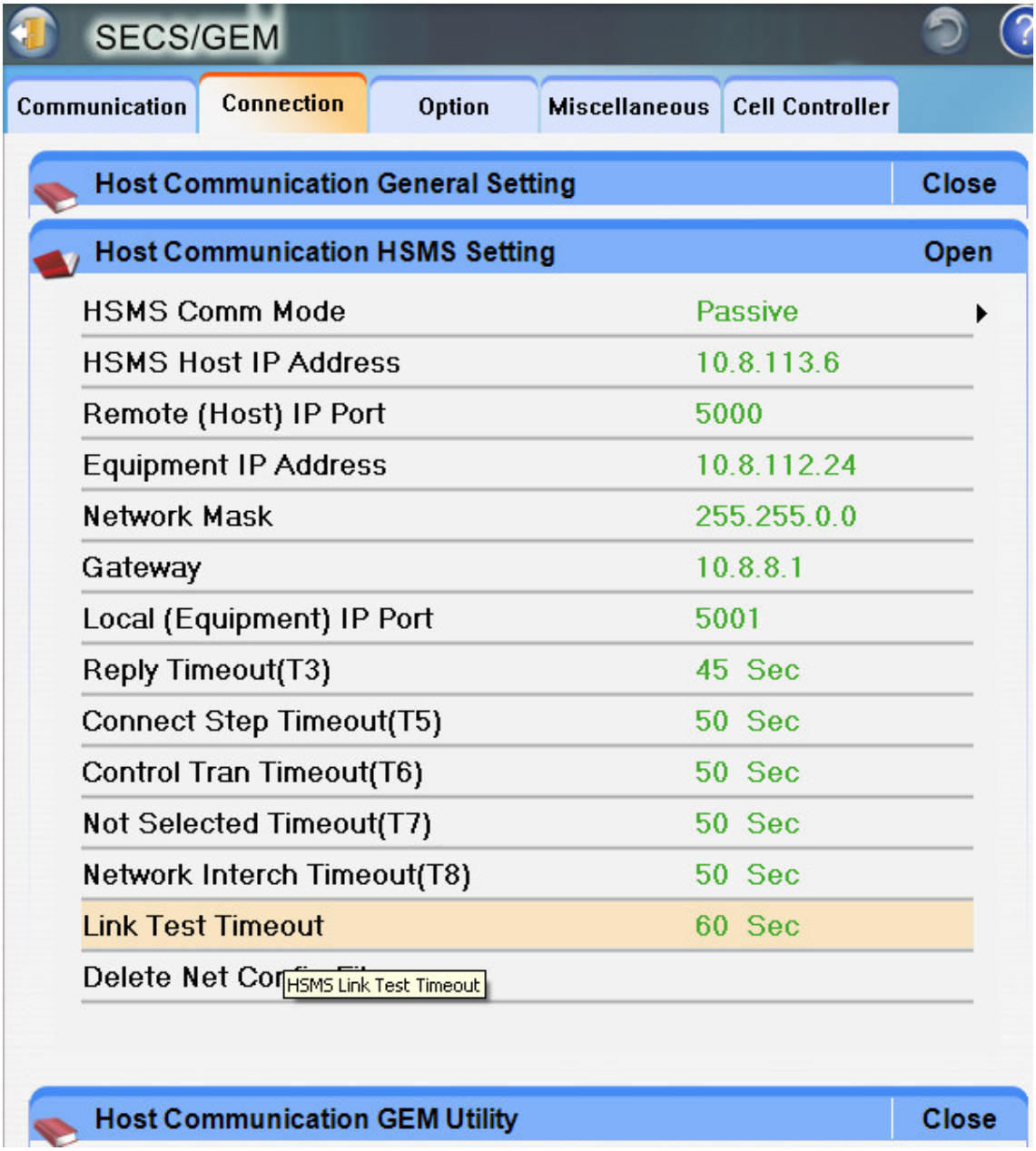
15.3. Hardware Settings for HSMS Communication

For HSMS communication ASM Eagle Xtreme wire bonder must have a Network Interface Card (NIC). Please use the PCI Ethernet card in PCI Slot 1 for Eagle Xtreme wire bonder. For more details about NIC installation and HSMS setup please refer the NIC installation document for Eagle Xtreme wire bonder.

15.4. Software Settings for HSMS Communication

"HSMS Configuration" menu page can see under "SECS\GEM configuration" menu page. This menu page will be bright only for the customers those who are using HSMS communication features. Customer should setup these menu items according to their network and host computer settings.

0. **HSMS communication mode** : According to SECS\GEM standard, 2 types of communication modes can set in the bonder (PASSIVE or ACTIVE) . Normally in equipment side this mode is PASSIVE and host side this mode is ACTIVE.
1. **Hsms Remote (HOST) IP**: This one is IP address of the Host computer. Here user has to give a valid HOST IP address . Otherwise bonder will prompt an error message "Invalid IP address". The standard format of IP is four gorup of numbers separated by dot. But here in ASM AB339E wire bonder instead of dot user has to give a blank space as the separator or delimit (eg: 100 10 25 8)
2. **Remote (Host) IP Port**: Here user has to enter the IP port value of the host computer.
3. **HSMS Local (Equip) IP**: This number is the equipment's IP address. This menu item will be dim, so that user can not give any value manually. Because during the bonder initialization, bonder will get this value from the IP configuration file . This IP configuration file is existing in the bonder hard disk. If user changes any network configuration settings this file also will be updated dynamically. The Equipment IP address value is in standard format. Four group of numbers separated by dot.(eg: 100.10.25.8)

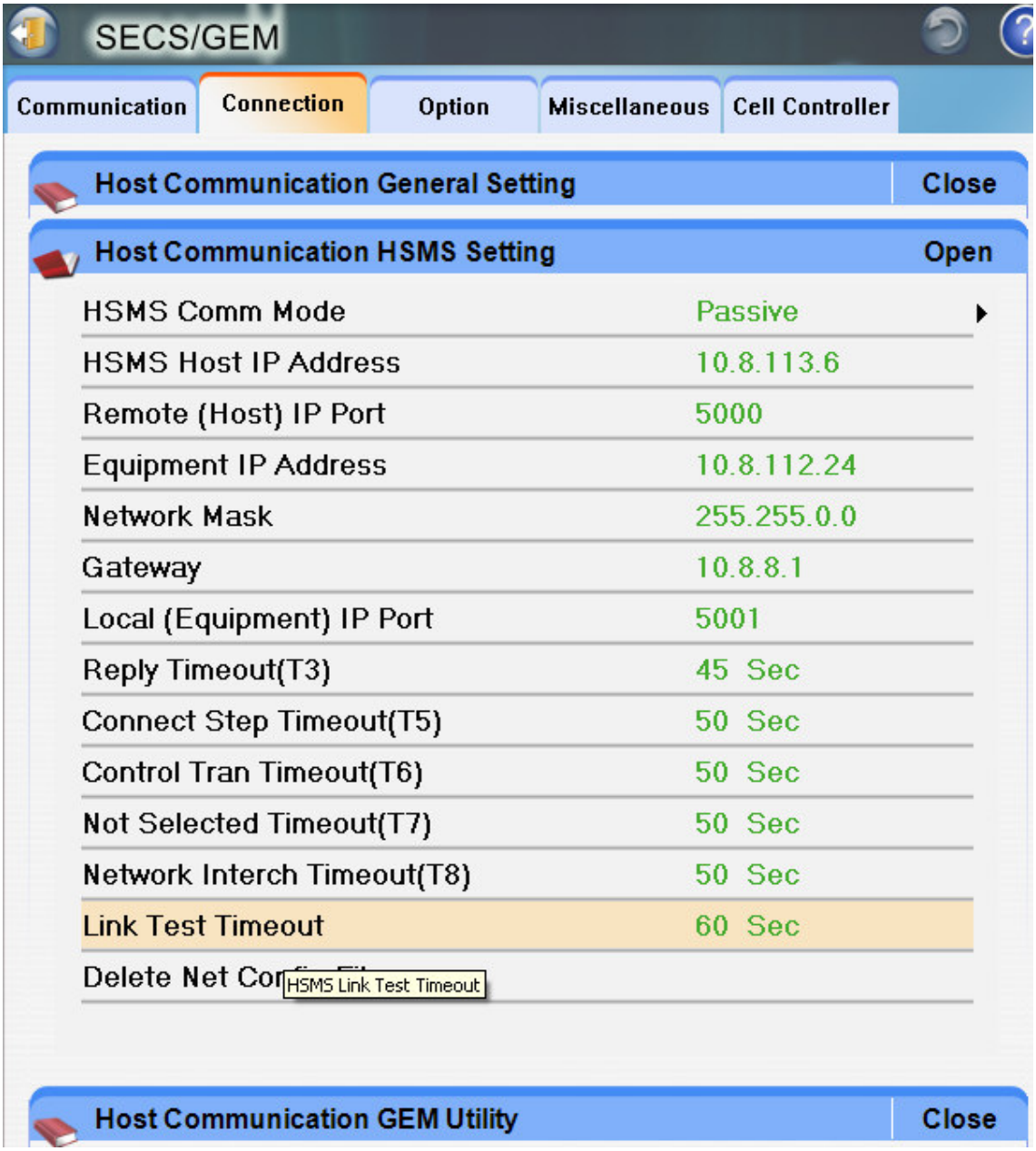


4. **Reply Timeout (T3):** Reply Timeout in HSMS protocol is a limit on the length of time that the HSMS message protocol is willing to wait for a reply message. Value range is 1 to 120 seconds
5. **Connect Separate Timeout (T5) :**This timeout is used to prevent excessive TCP\IP connect activity by providing a minimum time between the breaking, by an application program, of a TCP\IP connection or a failed attempt to establish one, and the attempt, by the same application program to initiate a new TCP\IP connection. Value range is 1 to 240 seconds

6. **Control Transact Timeout (T6) :** This defines the maximum time an HSMS control transaction can remain open before a communication failure is considered to have occurred. A transaction is considered open from the time the initiator sends the required request message until the response message is received. Value range is 1 to 240 seconds
7. **Not Selected Timeout (T7) :** This defines the time which a TCP\IP connection can remain in the NOT SELECTED state (i.e., no HSMS activity) before it is considered a communication failure. This property sets or returns the T7 "not selected" timeout value (in seconds) for the HSMS protocol. Value range is 1 to 240 seconds
8. **Network InterCh Timeout (T8)** : This is the maximum amount of time which may transpire between the receipt of any two successive bytes of a complete HSMS message before a communication failure is considered to have occurred. Value range is 1 to 120 seconds
9. **Link Test Timeout :** This value determines the amount of time between the link test control message issued by application software control. Such message are useful to determine whether or not the HSMS connection is intact , even in the absence of SECS message activity.

Please note that the unit of all HSMS timeout parameter values in ASM AB339E wire bonder menu is in seconds (s).

HSMS Network Configuration settings : This settings only one time required to do in the ASM Eagle Xtreme wire bonder..



0. **Equipment IP Addr:** This should be an unique IP address (Internet Protocol) for individual equipment. Here user has to give a valid IP address. Otherwise bonder will prompt an error message "Invalid IP address". The standard format of IP is four gorup of numbers separated by dot. But here in ASM AB339E wire bonder instead of dot user has to give a blank space as the separator or delimit (eg: 100 10 25 8)

1. **Netmask** : This value is according to the customer's LAN settings(Local Area Networking). Here user has to give a valid Netmask address. Otherwise bonder will prompt an error message "Invalid IP address". The standard format of Netmaks is four gorup of numbers separated by dot. But here in ASM AB339E wire bonder instead of dot user has to give a blank space as the separator or delimit (eg: 255 255 0 0)

2. **Gateway :** This value is also according to the customer's LAN settings(Local Area Networking). Here user has to give a valid Gateway address. Otherwise bonder will prompt an error message "Invalid IP address". The standard format of Gateway is four groups of numbers separated by dot. But here in ASM AB339E wire bonder instead of dot user has to give a blank space as the separator or delimit (eg: 100 172 18 123)

3. **Local (Equip) IP Port :** This is a part of TCP/IP communication. The complete network address is specified by IP address and IP port number. By default in ASM EAGLE wire bonder this value is 5000. Customer can change this value according to their Host settings.

Please give the above values from the bonder menu (HSMS configuration\Network configuration) with the help of network administrator. After modifying any of these three values bonder will prompt the message ***"Restart the bonder for new Setting"*** . The new values will be activated only after restarting the bonder. If any of these values modify, bonder will generate a dynamic IP configuration file inside the bonder hard disk

Note: Out of HSMS and RS-232 communications only one type communication user can configure in ASM Eagle Xtreme wire bonder. If wire bonder software is for HSMS communication, then all the RS-232 related setting parameters in bonder menu will be hided and vice versa.

15.5. HSMS Demo Version Protection

HSMS demo version is an evaluation software for customers who want to use HSMS communication feature for a particular time period for their production evaluation. In the demo version software HSMS communication feature will expire after that particular time period and so that customer can not use that feature after that time period. During the HSMS setting and Network configuration customer can identify whether the software is a demo version or the original version. In the demo version during the Network configuration settings, bonder will display the message "HSMS Demo Ver Protection". Also every time when the operator try to access the "HSMS configuration" menu page or "SECS\Gem Option" menu page the same message will be repeated.

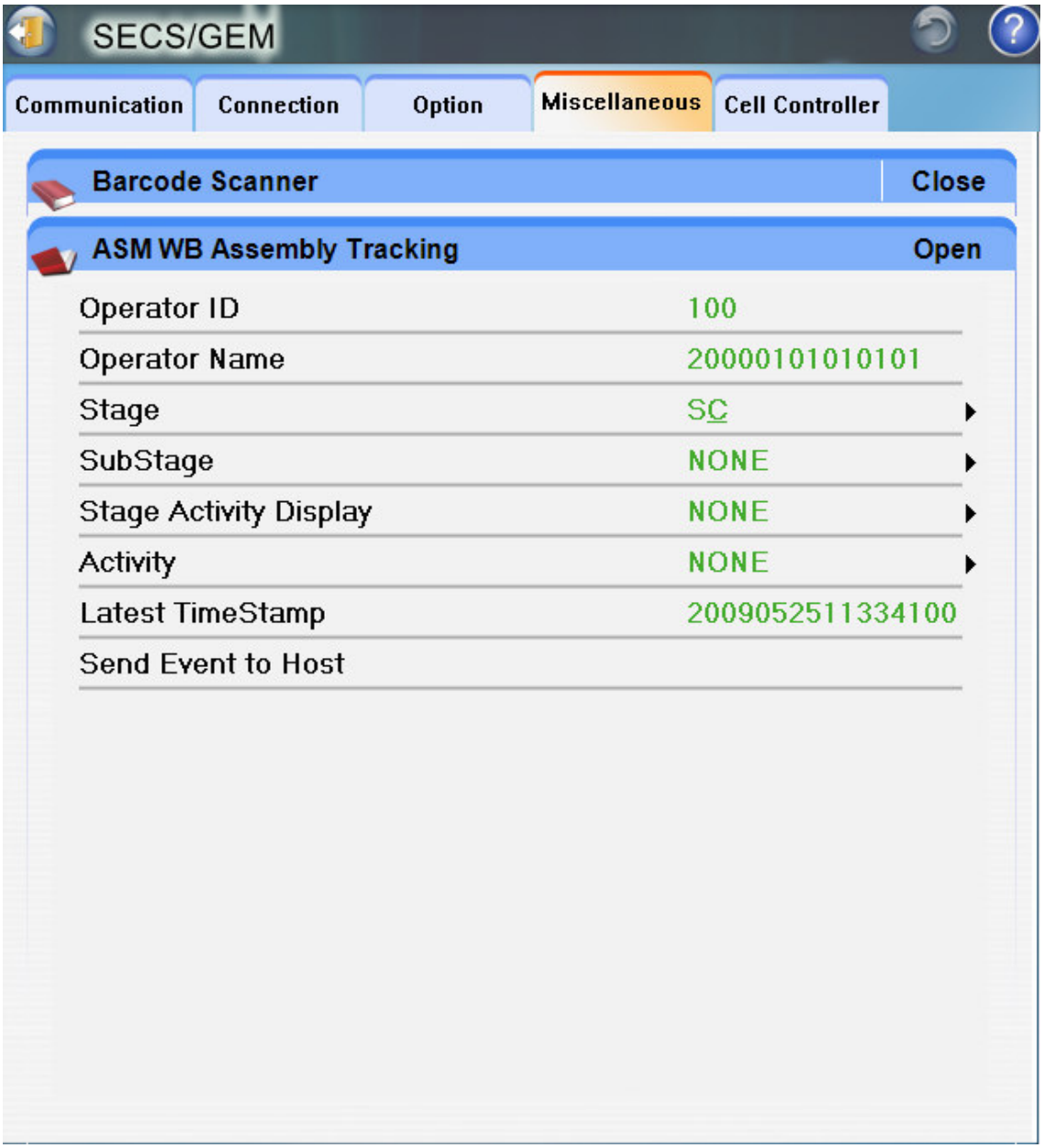
Later if customer satisfied in the demo version and they want to use HSMS feature permanently , then ASM will provide a dynamic password to make demo version into the original version ,so that they can use the same software in future also.

16. Wirebonder Assembly Tracking System

16.1. Introduction

The purpose of this feature is to monitor and control from Cell Controller different pre-shipment assembly stage/sub stage activities of wire bonder. This feature is available only for ASM in-house usage only for Quality assurance.

16.2. Menu Location



16.3.Assembly Stage Notification

Assembly Stage Notification (**ASN**) is to notify Cell Controller when there is any change of activity in stage/sub stage. When user changes the activity of a stage or sub stage, Wire Bonder Equipment shall report **ASN** to Cell Controller through event report (**S6F11/S6F12**) and to request ASN through event report request (**S6F15/16**).

	Element	Data Type	Elaboration
Assembly Stage Notification, ASN	**UserName**	**ASCII**	**D**escribes the user name of the user id that is currently attending to the equipment.
	UserID	**ASCII**	Describes the user id of the user that is currently attending to the equipment **Enforcement Rule:** Equipment has to prompt user to enter user id before allowing user to successfully start/end a stage/substage.
	Stage	**ASCII**	**D**escribes the stage that the equipment Definition of stages: "SETUP & CALIBRATION" "IN-HOUSE QUAD" "FINAL TEST & QUAD" "PRE-SHIPMENT"
	SubStage		Describes the substages of the equipment Definition of substages in **Stage 3**: "FORCE CALIBRATION" "MOTION DECOUPLING CALIBRATION" "FFC TUNING" "E-TORCH BURN IN" "INDEXING ACCURACY TEST"
	Activity	**ASCII**	Describes whether the equipment is starting/ending the stage Definition of Activity: "PENDING" "STARTED" "ENDED" "SUBMISSION-1" "SUBMISSION-2" "SUBMISSION-3" "SUBMISSION-4" "SUBMISSION-5"
	TimeStamp	**ASCII**	Start/End time for each stage/substage, as yyyymmddhhmmsscc.

Cell Controller can send Host command "ASN" with following element which helps to update the assembly Stage from remote Cell Controller without any user prompt on the equipment side.

	Element	Data Type	Elaboration
Update Assembly Stage ("ASN")	"Stage"	**ASCII**	Describes the stage that the equipment Definition of stages: "SETUP & CALIBRATION" "IN-HOUSE QUAD" "FINAL TEST & QUAD" "PRE-SHIPMENT"
	"SubStage"	**ASCII**	**D**escribes the substages of the equipment Definition of substages in **Stage 3**: "FORCE CALIBRATION" "MOTION DECOUPLING CALIBRATION" "FFC TUNING" "E-TORCH BURN IN" "INDEXING ACCURACY TEST"
	"Activity"	ASCII	Describes whether the equipment is starting/ending the stage Definition of Activity: "PENDING" "STARTED" "ENDED" "SUBMISSION-1" "SUBMISSION-2" "SUBMISSION-3" "SUBMISSION-4" "SUBMISSION-5"
	"TimeStamp"	**ASCII**	Start/End time for each stage/substage, as yyyymmddhhmmsscc.

17. eCheck – Wire Layout comparison

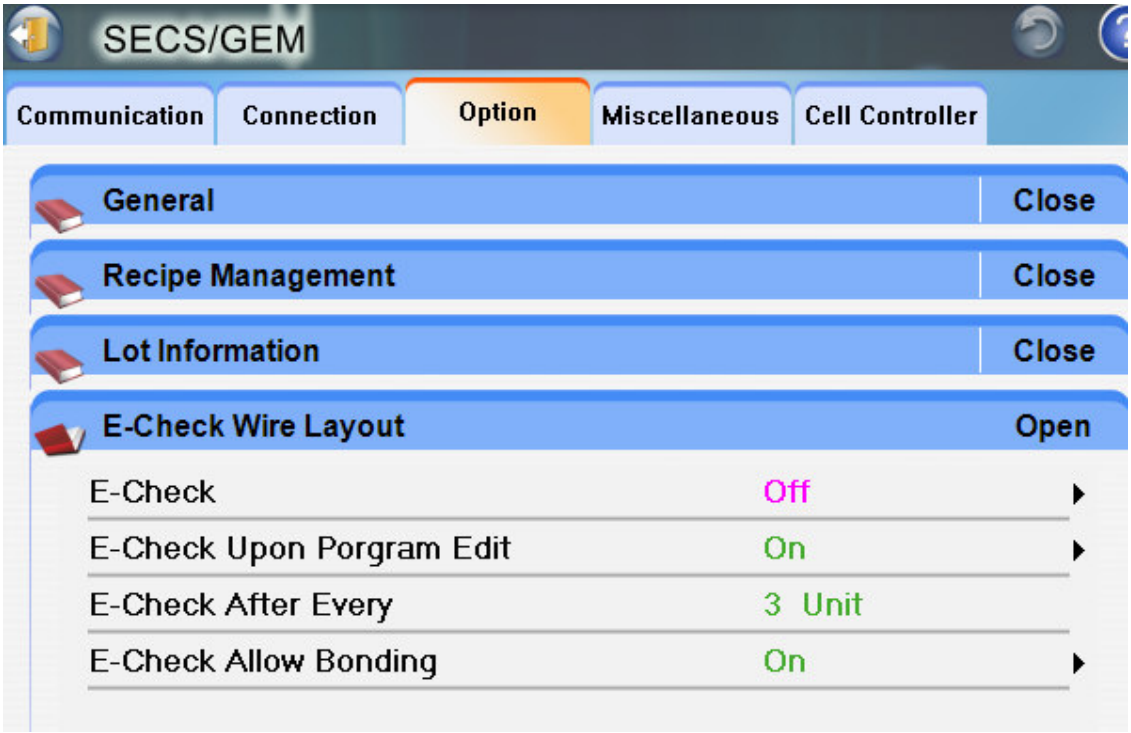
17.1. Introduction

This feature helps to compare the wire layout loaded at wire bonder with base program (or master program) stored at host, after teaching or prior to auto-bonding. Wire bonder will generate "LAYOUT.DAT" file and will upload to host and host will do the comparison between this newly generated .DAT file and Master program's .DAT file. The comparison will PASS or FAIL based on the predefined parameter tolerance limit setting on host side.

For more details about generating Master program and verification program refer "Manual of ASM Offline Verification Program (OLVP) or eCheck.

17.2. Menu location

The menu item is located under SECS/GEM communication page



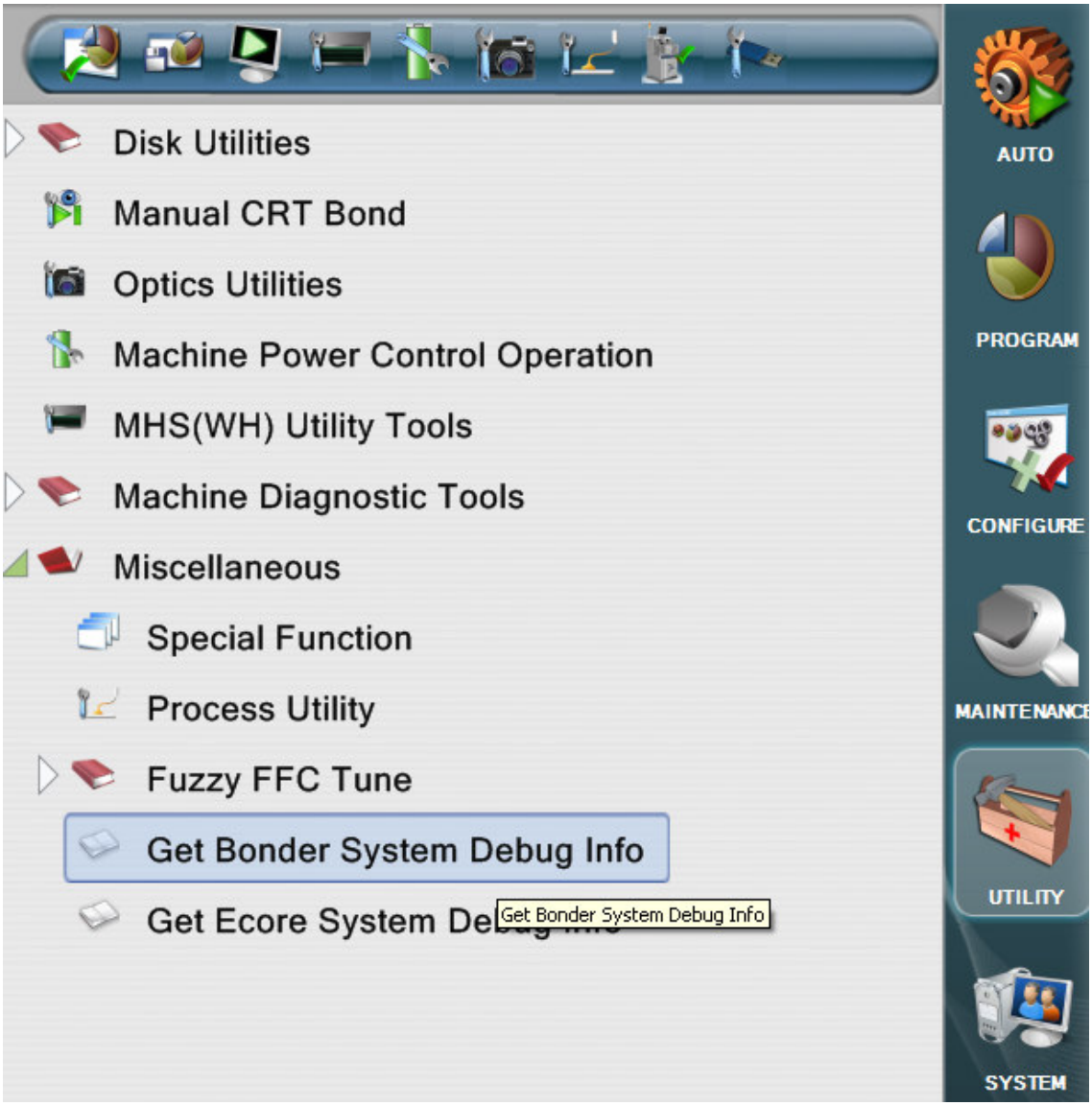
0: **"eCheck"** :- This is the master flag to support all eCheck related features. If it is disabled, none of the eCheck features will work out.

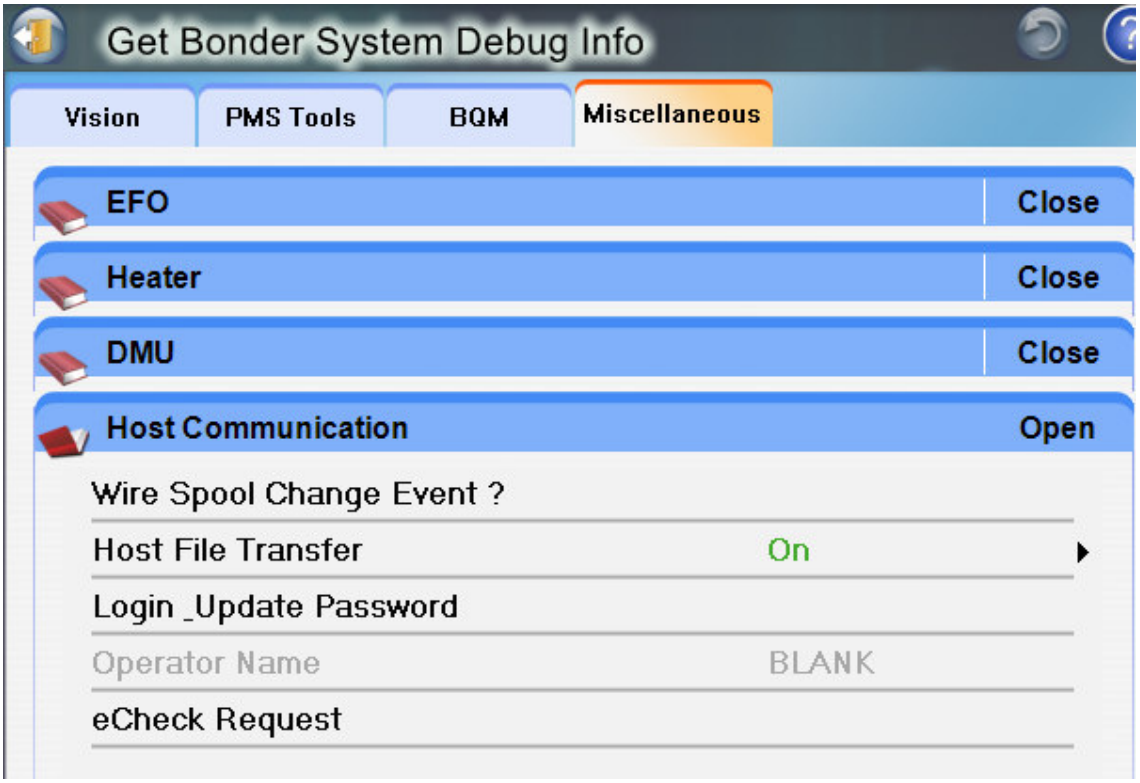
1: **"eCheck upon pgm Edit"** :- If this flag is YES , upon user edits or teaches the bond program, wirebonder will trigger for eCheck.

2: **"eCheck after every n units"** :- During autobond after every n units finish, wire bonder will trigger for eCheck. If this set as 0, then feature will disable. The value of n varies from 0 to 30000.

3: **"eCheck Allow Bonding"** :- This flag shows the status of eCheck PASS or FAILURE, based on the result from host. There are three possible values (YES (PASS), NO(FAIL), and Compare).

Besides these menu items, there is another menu item called **"eCheck Request"** inside "Utilities" → "Host Misc Operations" page. By this item user can manually trigger an eCheck request to host at any time.





17.3. eCheck Functionality

Different eCheck triggering methods are

1) **Host initiated eCheck**:-
When host wants to do eCheck wire layout
   - Host sends a remote command (RMT CMD 134) to wirebonder
   - Wirebonder will upload the file "LAYOUT.DAT" to host using S7F1, S7F2 , S7F3 & S7F4
   - Host will do the eCheck (wire layot comparison) while wirebonder waits for eCheck result.
   - Host will send the eCheck result to wirebonder , also it will allow/reject wirebonder to do bonding based on the eCheck result.

2) **Wirebonder initiated eCheck, upon teaching/modification of bond program**:-
If Operator Edits/Teaches the bond program, upon quitting the Teach menu page, wirebonder will trigger eCheck request event (CEID 86) to host .

3) **Wirebonder initiated eCheck, during autobond**:- Wirebonder will trigger eCheck request event (CEID 86) to host, just before start the bonding of first unit ; Also after every n number of units (user can key in this 'n' from menu).

4) **User can manually request eCheck:**- By this user can manually trigger an eCheck request to host at any time