Draft Minutes
Automation/Drive Interface (ADI) Working Group
Ad Hoc Meeting
T10/03-378r0
3-4 November 2003
9:00 AM – 7:00 PM (3 November)
9:00 AM – 11:00 AM (4 November)

1. Introductions: Group

   Paul Suhler called the meeting to order at 9:03 AM CST on 3 November 2003. He thanked Crossroads for hosting the meeting. A table of the attendees appears at the end of these minutes.

2. Approval of the agenda: 03-377r0 Paul Suhler

   Paul Suhler discussed the order of the discussion items. He re-ordered the existing discussion items. He asked Rod Wideman if we could discuss the status of ADC. Rod agreed. Michael Banther requested a discussion of Task attributes for encapsulated SCSI.

   Erich Oetting made a motion for acceptance of the modified agenda. Rod Wideman seconded the motion. The group passed the motion; no one objected or abstained.

3. Approval of previous meeting minutes: Paul Suhler

8-9 September 2003 meeting	03-295r0
22 September 2003 teleconference	03-327r0
20 October 2003 teleconference	03-341r0

   Paul Suhler requested comments for the minutes of the 8-9 September 2003 meeting, the 22 September 2003 teleconference, and the 20 October 2003 teleconference: 03-295r0, 03-327r0, and 03-341r0 respectively. No one provided comments.

   Rod Wideman made a motion for acceptance of the minutes as published. Michael Banther seconded the motion. In the absence of objections or abstentions, the group passed the motion unanimously.

4. Review of action items: Michael Banther
   a. Paul Entzel will write an appendix to ADT to describe an example login. *Carryover*
   b. Michael Banther will provide a proposal to define when exchanges open and when they close. *Closed*
   c. Susan Gray will provide state diagrams for the Port states (4.3) and the Link Negotiation states (4.4) as a proposal. *Closed*
   d. Paul Entzel will incorporate the proposals documented in discussion item (a) of 03-327r0 into ADT. *Closed*
   e. Paul Entzel will add a definition for the Bridging Manager in ADT. *Closed*

f. Kevin Butt will bring in a proposal to write the example containing figure 3 in ADT without the figure. *Carryover*

g. Susan Gray will propose diagrams showing specific instances of the new transport protocol service requests in the form of SAM3r09 figures 26, 28, and 30. *Carryover*

h. Paul Suhler will revise 03-318r0 per discussion item (c) of 03-341r0. *Closed*

i. Susan Gray will revise 03-355r0 per discussion item (d) of 03-341r0. *Closed*

j. Paul Suhler will revise 03-319r0 per discussion item (e) of 03-341r0. *Closed*

k. Kevin Butt will propose a definition of I_T Nexus Loss and the effects of an I_T Nexus Loss to ADT. *Carryover*

5. Discussion items:

a. ADT Baud Rate Fallback 03-380r0 Michael Banther

Michael Banther presented an overview of this proposal, i.e., ADT currently has no way to recover from transmittal errors (framing).

Michael Banther reviewed the cases where such recovery would be needed.

As a result of the issue, the automation and DTD will end up in a deadlock situation. The proposed change includes a threshold of framing errors before falling back to default baud rate and initiating port login again.

Michael Banther stated that the value of four appears to be a reasonable number of framing errors to threshold, based on the analysis contained in the proposal. He then walked us through the analysis details. Example shows a receiver running at twice the clock rate of the transmitter, the result being that the receiver may decode into valid values. This effect is due to the double sampling resulting in the receiver being able to detect appropriate start and stop bits.

Further into the analysis, Michael Banther showed that the minimum number of framing errors received would be four, based on the resultant frames not having valid stop bits.

Kevin Butt noted that the assumption was based on multiples of 9 600 baud. Michael Banther stated that he thought non-multiples would create more errors.

Michael Banther further walked us through the proposed changes to ADT. A definition is added; a statement regarding detection; a section stating recovery. Paul Entzel asked for what type of errors the recovery statement was based. Rod Wideman pointed out a minor edit to the definition.

Paul Suhler asked if was meaningful to state the detection of a bad start bit (extra sampling). Discussion concluded most UARTs wouldn't sample enough to accomplish such detection. Paul Suhler clarified that perhaps a better question was whether we state requiring UARTs to detect bad start bits. The group reached a consensus of no.

Susan Gray asked whether this recovery would cause an oscillation between recovering (fallback, negotiate up, fallback, etc.). The group stated the general implementation might be to limit the recovery attempts and then stop negotiating upwards.

Michael Banther described an analysis of the amount of clock drift needed to create the issue. He thought a significant drift was required.

Group discussed the alternate method of sending a No Op after completion of link negotiation. We concluded that a No Op isn't sufficient, and the proposal covered more general cases.

Rod Wideman made a motion for incorporation of 03-380r0 as revised into ADT. Erich Oetting seconded the motion. In the absence of objections or abstentions, the motion passed unanimously.

b. ADT Section 4.7.1.3 03-355r2 Susan Gray

Susan Gray walked the group through the changes from revision zero of this proposal. This proposal incorporates Paul Suhler's 03-318 proposal as well.

Paul Suhler described the need for an Initiating Recovery port state.

We discussed the possibility of the receiver of an Initiate Recovery IU sending a NAK IU. At present the receiver will not send a NAK IU on a Checksum error. Some discussion occurred between Paul Entzel, Susan Gray, Paul Suhler, and Michael Banther resulting in agreement that the existing ACK timeout will eventually cause fallback to default baud rate and hence recovery.

Susan Gray talked us through the new text for sub-clause 4.7.1.2 and 4.7.1.3.

Rod Wideman pointed out that the proposed text specifies errors that can be retried and those that cannot in two separate places. He suggested limiting the specification to a single place, a suggestion that Paul Entzel and Kevin Butt supported. Consequently the lists in 4.7.1.3.2 and 4.7.1.3.3 will disappear.

We debated the proposed text specifying that the receiver enters P3: Pending Recovery port state after sending the NAK IU. Paul Entzel gave his opinion that the paragraph that describes the sending of the NAK IU should belong in 4.7.2, Error Recovery for Transmission Error. We debated various strategies for how to present the text, in what order and combination.

We wordsmithed the remaining text in 4.7.1.3

And then we moved on to 4.7.2, Error Recovery and then began wordsmithing it.

Rod Wideman pointed out a possible problem: can the receiver of a Port Login IU in port state P2: Active detect an error, and what does it do if it detects one? The existing text assumes that the receiver is always in P1: Login state. To prepare ourselves to answer these questions, we agreed to review the proposed table showing which errors can be retried and which cannot.

A series of comments by Paul Entzel, Paul Suhler, Michael Banther, and Kevin Butt resulted in the realisation that the setting of the REGEXP bit is tied to the port state as well as the specific error returned. A port in P3: Pending Recovery has to set REGEXP for every subsequent NAK IU regardless of the error in order to deal with lost NAK IU's. That realisation led Paul Entzel to point out that the new state definitions can lead a port to being in two states at once. A port can have initiated recovery for traffic it initiated and it also can have sent a NAK IU for traffic it received independently by the other port.

Hence a port can be in P3: Pending Recovery and P4: Initiating Recovery at the same time. We discussed whether we should break the port states into two state machines.

Rod Wideman led us back to the main topic, how does the REGEXP bit relate to port states and NAK Status Codes. We agreed that REGEXP equals one when the receiver sends a NAK IU from port state P3: Pending Recovery. This decision has a knock-on effect on the proposed definition of P3: Pending Recovery: the port should enter it after detecting the error but before transmitting the NAK IU.

Finally we moved back to reviewing the Retryable column of table x+1. A discussion of the Invalid or Illegal Pause IU Received entry caused Michael Banther to question whether disallowing a Pause IU during error recovery is a good idea. Paul Suhler agreed. Some rewording of the definition of P3: Pending Recovery may be in order. We also agreed to make the error code generic to all illegal IU's.

Kevin Butt questioned the title of the 'Retryable' column. He pointed out that several times people have stated that a 'Y' indicates recovery via the Initiate Recovery IU. However we agreed to leave the title as is.

Rod Wideman returned us to his original question: can the receiver of a Port Login IU in port state P2: Active detect any errors with the information unit? A discussion of the revised table x+1 revealed that only one error code, Unexpected Frame Number, still has a 'Y' in the Retryable column. Based on this fact, Rod argued to leave the table in its present location, remove the Retryable column, and point out in 4.7.1 that only an Unexpected Frame Number causes the receiver to enter port state P3: Pending Recovery and send a NAK IU.

We debated what restrictions, if any, exist on the Frame Number of a Port Login IU. Paul Entzel stated that no restriction exists. Michael Banther pointed out that 4.6.2 requires the initiator of a Port Login negotiation sequence to set the Next Frame To Send counter, and hence the first Frame Number, to zero. We discussed the extent of the implications of leaving this restriction in the standard or removing it. Rod Wideman would like to architect Port Login IU's such that the receiver does not have to worry about unexpected frame numbers. That way, the receiver's handling of a Port Login IU is identical regardless of the port state of the receiver. However, Paul Entzel and Michael Banther raised concerns about the consequences of this move. We agreed that we could not receive consensus on this issue until we review a revision of this proposal that incorporates the previously discussed comments.

Susan Gray talked us through the proposed changes to sub-clause 4.7.2.3. She received a few comments.

Rod Wideman suggested changing the name of the REGEXP bit to Pending Recovery (PR).

Susan Gray will revise 03-355r2 based on the comments received.

Rod Wideman, Susan Gray, and Paul Entzel pointed out the three outstanding items revealed by the discussion:

1. The need for separate transmit and receive state machines (see 03-369),

2. The restrictions, or lack thereof, on the Frame Number of a Port Login IU. Specifically should the restriction remain for a Frame Number of zero when initiating negotiation, or should we set aside frame number zero for Port Login IU's only?

3. Text is needed to require acknowledgement of frames in the order received.

c. ADT Informative Annex: Error detection and recovery action examples
03-367r2 Paul Suhler

Paul Suhler reviewed the drawing conventions. Rod Wideman suggested defining the acronym NFTS. Paul Suhler agreed to do so.

Paul Suhler walked us through the first ladder diagram. It raised a few questions but little disagreement. Paul Entzel stated that this diagram matched the existing ADT text.

Paul Suhler presented the second ladder diagram. It shows an ACK Offset of two or greater and a lost ACK IU on the first frame sent. Kevin Butt pointed out that the receipt of the ACK IU for the second frame sent before exhaustion of the ACK timer for the first frame presented an opportunity to detect the problem earlier. Paul Entzel and Paul Suhler pointed out that Kevin's comment relies on the, as yet un-proposed, change requiring Acknowledgements in the order of reception of their associated frames. Susan Gray suggested leaving the diagram alone, as the implementer has the option of using the out-of-order frame or not.

Paul Suhler and Paul Entzel agreed that the existence of a Recovery Frame Number on this diagram is not a good idea.

Paul Suhler presented the third ladder diagram, an ACK timeout due to a lost frame. Paul Entzel and Kevin Butt suggested removing IU B as it confuses the issue.

Paul Suhler presented the fourth ladder diagram, a lost frame followed by additional traffic from the same port. No one commented on this diagram.

Paul Suhler presented the fifth ladder diagram, a frame with a non-retryable error detected by the receiver. Kevin Butt raised an objection to the apparent linkage between the failing frame and the subsequent successful frame. Paul Suhler and Michael Banther suggested adding arrows from and to the ULP's in both devices.

Having presented all of the diagrams in the annex, Paul Suhler asked if a diagram showing a Port Login re-negotiation would be useful. Paul Entzel stated it would be great if Paul Suhler could sort out all of the interactions.

Paul Suhler showed us a diagram under development. This diagram shows a lost ACK IU followed by a lost NAK IU. Kevin Butt suggested adding text to the diagrams showing the ACK offset value. We debated whether the normative text should make any statement about the receiver checking for the transmitter exceeding the Maximum ACK Offset. Paul Entzel argued strongly against such text stating that he expects difficulty in the receiver recovering from an out-of-resources condition.

Susan Gray discovered text in sub-clause 4.5 that requires acknowledgement of frames in the order received, resolving one of the outstanding worries of the previous discussion.

Paul Suhler shows us another ladder diagram under development. This one shows a receiver-detected error with the receiver's NAK IU causing the transmitter to cancel its ACK timer. Kevin Butt asked what happens if the non-retryable NAK IU of the frame numbered k is lost, the transmitter send frames k+1 and k+2, and then the receiver sends a NAK IU for frame k+1.

Paul Suhler presented a rather difficult case. It consists of a lost ACK IU for frame k followed by a NAK IU for frame k+1 that causes the transmitter to send an Initiate recovery IU followed by expiration of the ACK timer for frame k. The exhaustion of the ACK timer causes the transmitter to send an Initiate Recovery IU for frame k after the Initiate Recovery IU for frame k+1. We resolved this problem by having the reception of the NAK IU for frame k+1 cancel the ACK timer for frame k.

Paul Entzel pointed out that the solution described above will not work if the originally lost acknowledgement is a NAK IU for a non-retryable error. The receiver must ensure that it responds to the re-transmission of the frame with the lost NAK IU with a duplicate NAK IU, not an ACK IU. This requirement implies substantial change to the existing normative text.

Paul Suhler will revise the proposal based on the comments received.

d. ADT State Machine 03-369r0 Susan Gray

Susan Gray began to take us through the proposal.

Rod Wideman noted the absence of port state P4. Susan replied that P4 is a new state; hence she did not include it.

Kevin Butt objected to the statement that the ADT transport layer contains a single state machine. We know from the previous discussion that each port will contain a separate state machine for transmitting and receiving. Susan agreed to change the text to indicate two state machines. Rod Wideman pointed out that some states apply to both state machines, e.g., P0: Initial, P1: Login, and some will only apply to one of the state machines, e.g., P3: Recovery Pending (receiver) and P4: Initiating Recovery (transmitter).

Paul Entzel suggested using 'T' and 'R' sub-states to P2: Active to define recovery. The 'T' sub-states will apply to the transmitter's state machine only. The 'R' sub-states will apply to the receiver's state machine only. This idea received general acceptance. The transmitter sub-state machine contains two states: normal and initiating recovery. The receiving sub-state machine contains three states: normal, recovery pending, and recovering. After some discussion, Paul Entzel and Susan Gray agreed to describe the one upper-level and three sub-state machines as four separate state machines.

We returned to a consideration of the proposed text. Susan received several comments for each state clarifying and restricting the allowed behaviour. In general we added missing text defining the allowed Information Units and conditions for entering and leaving each state.

The group ran into some confusion when considering the description of the N0: Initiate state description. To what frames shall the port reply with a NAK IU when in this state? Eventually Paul Entzel, Rod Wideman, and Susan Gray agreed to respond with a NAK IU

to any frame other than a Port Login IU or an acknowledgement IU. Michael Banther pointed out that this list should include a Port Logout IU to maintain consistency with the description of port state P1: Login. Kevin Butt summarised the effect of these changes on the associated state diagram. Rod Wideman requested a transition from Negotiation state N0: Initiate to itself in the case of a timeout waiting for the receipt of a Port Login from the other port.

Paul Entzel and Kevin Butt debated whether the protocol should allow a Port Logout IU in any state. Previously we had agreed that a Port Logout IU would not be valid from port state P0: Initial. Michael Banther asked what would happen if a port in P0 received a Port Logout. Erich Oetting replied that the port would respond with a NAK IU and then log out. After some brief chuckles, Paul Entzel pointed out that receipt of a Port Logout IU by a DTD in P0: Initial state does add some value because it transitions the DTD to port state P7: Logged Out. From this state, the DTD cannot initiate link negotiation.

Having achieved an agreement that a port can process a Port Logout IU from any state, we debated how to specify the operation of it. Do we specify it once globally or in each state description? We agreed to use the text to capture the actions taken whilst in the state. We also agreed to use a state machine figure to specify the transitions from state to state and the events that cause them. Finally, we agreed to define actions taken during the transition in the text.

Susan Gray will revise the proposal based on the comments received.

e. ADT Transfer Ready IU / Response IU Race Condition

03-319r1 Paul Suhler

Paul Suhler described the changes in this revision. He and Paul Entzel agreed that the proposed Invalid Exchange ID NAK status code should not be retryable.

Paul Entzel investigated the ‘single-IU exchange’ concept and discovered that it’s used in both the Link Service and Fast Access protocols. He requested that the definition of it reside in the model or definition sub-clause. Paul Suhler asked for a better name than ‘single-IU exchange’. After some discussion, Paul Entzel suggested ‘simple exchange’ as a good moniker. Several others agreed.

Rod Wideman raised concerns with the use of the terms ‘active’ and ‘invalid’ to describe the state of an exchange. Paul Suhler suggested changing to ‘begin’ and ‘end’. Paul Entzel countered with reusing the terms associated with task lifetimes in SAM-x. Paul Suhler checked and found that SAM-x uses ‘begin’ and ‘end’ also.

Susan Gray objected to the lack of information regarding when a negotiation exchange begins and ends in the receiver. The proposed text covers only the sender of the initial Port Login IU.

Rod Wideman raised an objection to some ambiguous language in the proposed text in the SCSI Exchange Lifetime sub-clause. His objection led to a discussion of how a Task Management Function that aborts a task in another exchange will operate. We digressed into a discussion of whether an Abort Task will use the exchange of the task it’s aborting or not. Paul Entzel stated that using the same exchange is dangerous. The response to the task management function may relate to the aborted task or to the task management

function Abort Task, and the initiator cannot tell the difference. The same concern applies to the task management function Query Task. Michael Banther asked how parallel SCSI handles this problem. Paul Entzel replied that parallel SCSI goes to Bus Free phase; hence no need exists in this transport. Kevin Butt explored SAS and found that this protocol provides a separate information unit for task management functions and this information unit contains a field to identify the exchange being managed. Paul Entzel listed a series of possible solutions. Of them, Paul Suhler suggested creating a separate information unit for task management functions. This idea received general support. Paul Entzel agreed to propose it, although Rod Wideman raised some reservations about the usefulness of supporting the task management function Abort Task.

Michael Banther asked why we do not simply refer to the definition of task lifetimes given by SAM. Paul Entzel and Paul Suhler stated that we still need to describe what happens when a port receives an encapsulated SCSI IU outside of an existing exchange. Paul Suhler's proposal already includes text to cover this case. Paul Entzel asked how a Port Logout will affect an open encapsulated SCSI exchange. Kevin Butt suggested an I_T Nexus Loss, and his proposal defining I_T_ Nexus loss will cover this concern.

Susan Gray raised a concern that the proposed text doesn't cover the race condition initially identified as the problem this proposal seeks to resolve. Paul Suhler believes that the problem has been fixed. However talking through the sequences for one port sending a SCSI Response IU and the other sending a Transfer Ready IU, it appears that a window of vulnerability still exists. However Paul Suhler believes that the transport level can still sort the problem out. He will provide clarifying text as part of the revision to reference SAM task lifetimes.

We moved on to the section on Fast Access exchanges. Rod Wideman raised several editorial comments.

Paul Suhler will revise based on the comments received.

f. ADT Fast Access Errors 03-382r0 Michael Banther

Straight out of the chute, Rod Wideman made a motion for acceptance. Michael Banther seconded the motion.

Rod Wideman described the proposal (kind of nice for the secretary). Paul Entzel objected to using a link-level response (a NAK IU) to indicate a problem detected at a layer above the transport. He agrees that a problem exists, but doesn't like the solution because it violates layering. Rod Wideman subsequently withdrew his motion.

Paul Entzel suggested that a DTD receiving a multi-bit field containing at least one bit equal to zero treat the entire field as equal to zero. He suggested a usage model for the AER Control IU:

1. The automation port sends it with all bits equal to one,

2. The data transfer device resets all reserved bits and fields/bits for which it does not support asynchronous reporting to zero,

3. The automation port uses the AER Control IU returned by the DTD as the basis for configuring the DTD's asynchronous reporting. The automation port does not change any of the zero bits to one. The automation port may change any of the one bits to zero to turn off the asynchronous reporting that it does not want.

When the DTD returns the AER Control IU, it will set all of the bits in the multi-bit field to zero. The automation port can use this change to conclude that it sent an invalid value for the multi-bit field. Michael Banther agreed to make this change.

Michael Banther raised the need for reporting reserved bits set by the automation port. After some debate, he requested that we table the discussion so that he could discuss the situation with other HP engineers.

g. Status of ADC Rod Wideman

Rod Wideman reported on the status of ADC letter ballot comments. He received 1080 editorial, 93 technical, and 29 other comments.

Rod Wideman highlighted StorageTek's comment requesting removal of Annex A. Adoption of it will affect several tens of comments. Erich Oetting explained that StorageTeks's concern centred on the appropriateness of this type of material in a standard. The group agreed to leave the annex in the draft standard.

Rod Wideman expects to post the list of comments to the T10 web site.

h. Task Attributes for SCSI Encapsulation Michael Banther

Paul Entzel asked if anyone needs to support untagged tasks once we add a Task Attributes field. No one claimed a need to support untagged tasks.

Michael Banther raised a concern about the non-existence of 'simple task queuing' in SAM-2. He stated his preference that we render ADT silent on the tasking model and add a Task Attribute field in the encapsulated SCSI Request IU.

Paul Entzel intends to add a Task Attribute field to the encapsulated SCSI Request IU modelled on SAS. He has agreed to remove the sentence requiring a tasking model.

Paul Entzel will provide a proposal to accomplish the changes described above.

6. Unscheduled business:

a. Port Login IU Frame Number Restrictions

We recapped the concern that led to this issue. Paul Entzel explained that the issue revolves around using Port Login in the middle of frame recovery to renegotiate the baud rate in an attempt to attain reliable communication between the ports.

Paul Suhler recalled the suggestion that we set aside frame number zero for Port Login IU's. Paul Entzel suggested that we extend the change to cover Port Logout, Pause, and No Op IU's as well. All of these IU's operate in an interlocked manner, i.e., a port cannot have more than one outstanding at a time.

Kevin Butt pointed out that we will have to change the largest available Maximum ACK offset. Paul Entzel agreed that it will have to be reduced to a two bit field.

Paul Entzel will prepare a proposal regarding these changes.

b. Task Management IU for ADT 03-391r0 Paul Entzel

Paul Entzel walked us through the proposal. He fielded various questions. He brought up several potential editorial changes and received comments on them.

Paul Entzel made a motion for acceptance of 03-391r0 into ADT. Michael Banther seconded the motion.

Kevin Butt requested a review of the existing Response Codes in ADT. Kevin asked exactly what Response Code Paul is adding. Paul Entzel suggested clarifying the name of the new Response Code to make it specific to task management functions. Paul Suhler and Kevin Butt agreed with this change.

In the absence of objections or abstentions, the motion passed unanimously.

c. Formatting of Response Codes

Paul Suhler asked George Penokie on the preferred formatting of transport layer response codes. George suggested using all CAPS.

d. Review of ADC comments Rod Wideman

As George Penokie attended the Tuesday morning portion of the meeting, Rod Wideman walked us through a review of the comments submitted by George (IBM/Tivoli) and marked as technical.

7. Next meeting requirements: Paul Suhler

The group will hold teleconferences on 17 November 2003 and 15 December 2003. These teleconferences will begin at 8:00 AM PST and conclude at 10:00 AM PST.

The group will hold a meeting 12-13 January 2004 during T10 plenary week in Chandler, AZ. Subject to approval by T10, the meeting will begin on the 12th at 9:00 AM and conclude at 7:00 PM MST. The meeting will reconvene on the 13th at 9:00 AM and adjourn at 11:00 AM MST.

8. Review new action items: Michael Banther

a. Paul Entzel will incorporate 03-380r0 into ADT.

b. Susan Gray will revise 03-355r2 per discussion item (b).

c. Paul Suhler will revise 03-367r2 per discussion item (c).

d. Susan Gray will revise 03-369r0 per discussion item (d).

e. Paul Suhler will revise 03-319r1 per discussion item (e).

f. Paul Entzel will prepare a proposal covering frame numbers in link service frames.

g. Paul Entzel will incorporate 03-391r0 into ADT.

9. Adjournment: Group

   Erich Oetting made a motion for adjournment. Kevin Butt seconded the motion. In the absence of objections or abstentions, the group passed the motion unanimously. Paul Suhler adjourned the group at 10:52 AM CST on 4 November 2003.

Attendees:

Name	Organization	E-mail
Rod Wideman	ADIC	rod.wideman@adic.com
Paul Suhler	Certance	paul.a.suhler@certance.com
Bob Griswold	Crossroads	rgriswold@crossroads.com
Michael Banther	HP	michael.banther@hp.com
George Penokie	IBM	gop@us.ibm.com
Kevin Butt	IBM	kdbutt@us.ibm.com
Paul Entzel	Quantum	paul.entzel@quantum.com
Susan Gray	Quantum	susan.gray@quantum.com
Tuong Vu	Sony	tuong.vu@am.sony.com
Erich Oetting	StorageTek	erich_oetting@stortek.com