STYLE 6000 SERIES CAN MESSAGING REFERENCE MANUAL
OPERATING INSTRUCTIONS

CAN MESSAGING

Prerequisite Knowledge

It is assumed that the reader has some level of familiarity with CAN networking and terminology, and the Akron Brass product. This information is not intended to be a tutorial on CAN, nor Akron Brass product operation, but rather a resource for information concerning the requirements and techniques involved in implementing a CAN control interface to the Akron Brass product line.

CAN Physical Layer

Akron Brass CAN physical layer requirements are generally as specified in SAE J1939-11 Physical Layer | SAE J1939-15 Reduced Physical Layer:

- CAN 2.0B devices
- 29 bit CAN message headers
- 250K bits per second
- Proper bus wiring, terminations, etc.

Address Claiming

Rather than using a pre-assigned CAN node address, Akron Brass controllers and display devices by default use the J1939 method of address claiming, detailed in SAE J1939-81 Network Management. Although the integrator must be aware of this, it is not necessary for them to understand or use this technique. Integrator considerations regarding this are summarized in Appendix A.

General Message Characteristics

To allow Akron Brass devices to cooperatively communicate on a vehicle bus shared with other message traffic, all Akron Brass messages utilize the point-to-point and broadcast ranges of message IDs set aside in SAE J1939-71 Vehicle Application Layer for vendors to use to transmit data not otherwise defined in the J1939 specification. The messages are structured as follows:

```
Header 2   2 2      1 1      0 0      0
   Bit 9...5 4......7 6......9 8......1
       [.]|| [......] [......] [......]
       [.]|| [......] [......] ↑ 8 bits, Source node
       [.]|| [......] ↑ 8 bits, PDU specific, Destination node ( FF(hex) if Broadcast )
       [.]|| ↑ 8 bits, PDU format, 11101111 = EF(hex) = 239 = PGN EF00(hex) = 61184
       [.]|↑ 1 bit, Data Page, 0
       [.]↑ 1 bit, Extended Data Page, 0
       ↑ 3 bits, Priority, 111 = 7 = lowest, 000 = 0 = highest
```

Data length - 1 to 8 bytes

Data byte 1 = Message type
Data bytes 2-8 = Data for message type (0 to 7 bytes)

Priority field is used only for physical CAN bus arbitration. All messages are parsed based on the low order 26 bits of the header.

No logical messages are spanned across multiple physical CAN messages.

Input Messages

J1939 PGN 64982 – Basic Joystick Message

```
Header 2   2 2      1 1      0 0      0
   Bit 9...5 4......7 6......9 8......1
       [.]|| [.......  .......] [......]
       [.]|| [.......  .......] ↑ 8 bits, Source node address( joystick, 33hex – 38hex )
       [.]|| ↑ 16 bits, PGN, 11111101 11010110 = FDD6hex = 64982
       [.]|↑ 1 bit, Data Page, 0
       [.]↑ 1 bit, Extended Data Page, 0
       ↑ 3 bits, Priority, 011 = 3

Data length – 8 bytes

Byte    1        2        3        4        5        6        7        8
 Bit 8......1 8......1 8......1 8......1 8......1 8......1 8......1 8......1
     [][][][] [......] [][][][] [......] [][][..] [][][][] [][][][] [][][][]
     [][][][] [......] [][][][] [......] [][][..] [][][][] [][][][] [][][]↑ Switch 12 status
     [][][][] [......] [][][][] [......] [][][..] [][][][] [][][][] [][]↑ Switch 11 status
     [][][][] [......] [][][][] [......] [][][..] [][][][] [][][][] []↑ Switch 10 status
     [][][][] [......] [][][][] [......] [][][..] [][][][] [][][][] ↑ Switch 9 status
     [][][][] [......] [][][][] [......] [][][..] [][][][] [][][]↑ Switch 8 status
     [][][][] [......] [][][][] [......] [][][..] [][][][] [][]↑ Switch 7 status
     [][][][] [......] [][][][] [......] [][][..] [][][][] []↑ Switch 6 status
     [][][][] [......] [][][][] [......] [][][..] [][][][] ↑ Switch 5 status
     [][][][] [......] [][][][] [......] [][][..] [][][]↑ Switch 4 status
     [][][][] [......] [][][][] [......] [][][..] [][]↑ Switch 3 status
     [][][][] [......] [][][][] [......] [][][..] []↑ Switch 2 status
     [][][][] [......] [][][][] [......] [][][..] ↑ Switch 1 status
     [][][][] [......] [][][][] [......] [][]↑ 4 bits ( unused )
     [][][][] [......] [][][][] [......] []↑ Y detent status ( ignored by Akron Brass devices )
     [][][][] [......] [][][][] [......] ↑ X detent status ( ignored by Akron Brass devices )
     [][][][] [......] [][][][] ↑ 8 bits, Y deflection ( high ) unsigned, 0.1%
     [][][][] [......] [][][]↑ Y neutral status
     [][][][] [......] [][]↑ Y back ( - ) status
     [][][][] [......] []↑ Y forward ( + ) status
     [][][][] [......] ↑ 2 bits, Y deflection ( low ) unsigned, 0.1%
     [][][][] ↑ 8 bits, X deflection ( high ) unsigned, 0.1%
     [][][]↑ X neutral status
     [][]↑ X left ( - ) status
     []↑ X right ( + ) status
     ↑ 2 bits, X deflection ( low ) unsigned, 0.1%
```

(2 bit status field values: 00 = Off, 01 = On, 10 = Error, 11 = N/A)

Example – [header] $0CFDD633_{hex}$ [8 data bytes] $107D046400100000_{hex}$

This message from the joystick at CAN node address 33_{hex} indicates its current state:

X deflection is 50% to the right

[X deflection = (low – high) 00..... 01111101 =
(unsigned 10 bit) .....01 11110100 = 1F4hex = 500 = 50.0%]

[X right (+) status = ..01.... = On]

Y deflection is 40% back

[Y deflection = (low – high) 00..... 01100100 =
(unsigned 10 bit) .....01 10010000 = 190_{hex} = 400 = 40.0%]

[Y back (-) status = ...01.. = On]

Switch 2 is active [Switch 2 status = ..01.... = On]

Refer to Appendix B for detailed information on how joystick input is handled by Akron Brass devices.

Input Messages (continued)

J1939 PGN 64983 – Extended Joystick Message

```
Header 2   2 2      1 1      0 0      0
   Bit 9...5 4......7 6......9 8......1
       [.]|| [.......  .......] [......]
       [.]|| [.......  .......] ↑ 8 bits, Source node address( joystick, 33hex – 38hex )
       [.]|| ↑ 16 bits, PGN, 11111101 11010111 = FDD7hex = 64983
       [.]|↑ 1 bit, Data Page, 0
       [.]↑ 1 bit, Extended Data Page, 0
       ↑ 3 bits, Priority, 011 = 3
```

Data length – 8 bytes

```
Byte    1        2        3        4        5        6        7        8
 Bit 8......1 8......1 8......1 8......1 8......1 8......1 8......1 8......1
     [][][][] [......] [][][][] [......] [][][][] [......] [][][][] [][][][]
     [][][][] [......] [][][][] [......] [][][][] [......] [][][][] [][][]↑ Switch 16 status
     [][][][] [......] [][][][] [......] [][][][] [......] [][][][] [][]↑ Switch 15 status
     [][][][] [......] [][][][] [......] [][][][] [......] [][][][] []↑ Switch 14 status
     [][][][] [......] [][][][] [......] [][][][] [......] [][][][] ↑ Switch 13 status
     [][][][] [......] [][][][] [......] [][][][] [......] [][][][]↑ ( reserved )
     [][][][] [......] [][][][] [......] [][][][] [......] [][]↑ Grip Theta detent status
     [][][][] [......] [][][][] [......] [][][][] [......] []↑ Grip Y detent status
     [][][][] [......] [][][][] [......] [][][][] [......] ↑ Grip X detent status
     [][][][] [......] [][][][] [......] [][][][] ↑ 8 bits,Grip Theta deflection (high)unsigned, 0.1%
     [][][][] [......] [][][][] [......] [][][][]↑ Grip Theta neutral status
     [][][][] [......] [][][][] [......] [][][]↑ Grip Theta back ( - ) status
     [][][][] [......] [][][][] [......] []↑ Grip Theta forward ( + ) status
     [][][][] [......] [][][][] [......] ↑ 2 bits, Grip Theta deflection (low) unsigned, 0.1%
     [][][][] [......] [][][][] ↑ 8 bits, Grip Y deflection (high) unsigned, 0.1%
     [][][][] [......] [][][][]↑ Grip Y neutral status
     [][][][] [......] [][][]↑ Grip Y back ( - ) status
     [][][][] [......] []↑ Grip Y forward ( + ) status
     [][][][] [......] ↑ 2 bits, Grip Y deflection (low) unsigned, 0.1%
     [][][][] ↑ 8 bits, Grip X deflection (high) unsigned, 0.1%
     [][][]↑ Grip X neutral status
     [][]↑ Grip X left ( - ) status
     []↑ Grip X right ( + ) status
     ↑ 2 bits, Grip X deflection (low) unsigned, 0.1%
```

(2 bit status field values: 00 = Off, 01 = On, 10 = Error, 11 = N/A)

Example – [header] $0CFDD733_{hex}$ [8 data bytes] $107D0464FFFF0004_{hex}$

This message from the joystick at CAN node address 33_{hex} indicates its current state:

Grip X deflection is 50% to the right

[Grip X deflection = (low – high) 00..... 01111101 =
(unsigned 10 bit) .....01 11110100 = $1F4_{hex}$ = 500 = 50.0%]

[Grip X right (+) status = ..01.... = On]

Grip Y deflection is 40% back

[Grip Y deflection = (low – high) 00..... 01100100 =
(unsigned 10 bit) .....01 10010000 = 190_{hex} = 400 = 40.0%]

[Grip Y back (-) status = ...01.. = On]

Grip Theta deflection is not applicable (FFFF).

Switch 15 is active [Switch 15 status = ….01.. = On]

Refer to Appendix B for detailed information on how joystick input is handled by Akron Brass devices.

Status Messages – Broadcast

Message type FF_{hex} – Address Announcement

This message is issued to inform the integrator what CAN node address has been claimed for use by the Akron Brass device having the indicated PIN (15 digit Product Identification Number, similar to Ethernet Mac ID in that it uniquely identifies any Akron Brass device).

Header per General Message Characteristics, Destination node = FF_{hex} (Broadcast)

Data length – 8 bytes

```
Byte    1        2        3        4        5        6        7        8
 Bit 8......1 8......1 8......1 8......1 8......1 8......1 8......1 8......1
     [......] [.....][ .....][. ........ .][..... ........ ........ .......]
     [......] [.....][ .....][. ........ .]↑ 30 bits ( high - low ) unsigned, PIN digits 7-15
     [......] [.....][ .....]↑ 12 bits ( high - low ) unsigned, PIN digits 1-4
     [......] [.....]↑ 7 bits, PIN digit 6 ( ASCII character )
     [......] ↑ 7 bits, PIN digit 5 ( ASCII character )
     ↑ 8 bits, message type, 11111111 = FF hex
```

Example – [header] $18EFFF80_{hex}$ [8 data bytes] FF83150226021A0Bhex

This message indicates that the 6032 Universal II device with PIN '1032AE100801035' has claimed CAN node address 80_{hex} and is ready to issue and accept CAN messages.

[PIN digits 1-4 = (high – low) .....01 00000010 00..... =
(unsigned 12 bit) ....0100 00001000 = 408_{hex} = 1032]
[PIN digit 5 = 1000001. = (7 bit ASCII) .1000001 = 41_{hex} = 'A']
[PIN digit 6 = ...1 000101.. = (7 bit ASCII) .1000101 = 45_{hex} = 'E']
[PIN digits 7-15 = (high – low) ..100110 00000010 00011010 00001011 =
(unsigned 30 bit) ..100110 00000010 00011010 00001011 =
$26021A0B_{hex}$ = 100801035]

Refer to Appendix A for more detailed information regarding how this message should be processed.

Message type FD_{hex} – Diagnostic Data (Akron Brass internal / diagnostic use)

These messages are issued when a device is placed into Diagnostic mode, and are processed by the Akroview diagnostic program. The format and content are specific to an individual device type.

Header per General Message Characteristics, Destination node = FF_{hex} (Broadcast)

Data length – 3 to 8 bytes

```
Byte    1        2        3        4        5        6        7        8
 Bit 8......1 8......1 8......1 8......1 8......1 8......1 8......1 8......1
     [......] [......] [ ...... ........ ........ ........ ........ .......]
     [......] [......] ↑ Format and content vary by device
     [......] ↑ 8 bits, Diagnostic record type
     ↑ 8 bits, message type, 11111101 = FD hex
```

Example – [header] $18EFFF80_{hex}$ [5 data bytes] $FD01000000_{hex}$

Format and content vary by device.

Status Messages – Broadcast (continued)

Message type 01_{hex} – Position / Limit Status

This message is issued periodically (100ms) by the 6032 Universal II, 6040 Forestry and 6042 Global Platform controllers to indicate axis positions (for units with analog feedback), travel limit status, et.al.

Header per General Message Characteristics, Destination node = FF_{hex} (Broadcast)

Data length – 8 bytes

```
Byte    1        2        3        4        5        6        7        8
 Bit 8......1 8......1 8......1 8......1 8......1 8......1 8......1 8......1
     [......] [.......  ...][][] [.......  ...][][] [][][][] [][][][] [][][][]
     [......] [.......  ...][][] [.......  ...][][] [][][][] [][][][] [][][]↑ Calibration status
     [......] [.......  ...][][] [.......  ...][][] [][][][] [][][][] [][]↑ 2 bits ( unused )
     [......] [.......  ...][][] [.......  ...][][] [][][][] [][][][] []↑ Axis 5(-) hard stop status
     [......] [.......  ...][][] [.......  ...][][] [][][][] [][][][] ↑ Axis 5(+) hard stop status
     [......] [.......  ...][][] [.......  ...][][] [][][][] [][][][]↑ Axis 4(-) hard stop status
     [......] [.......  ...][][] [.......  ...][][] [][][][] [][]↑ Axis 4(+) hard stop status
     [......] [.......  ...][][] [.......  ...][][] [][][][] []↑ Axis 3(-) hard stop status
     [......] [.......  ...][][] [.......  ...][][] [][][][] ↑ Axis 3(+) hard stop status
     [......] [.......  ...][][] [.......  ...][][] [][][]↑ Axis 2(-) hard stop status
     [......] [.......  ...][][] [.......  ...][][] [][]↑ Axis 2(+) hard stop status
     [......] [.......  ...][][] [.......  ...][][] []↑ Axis 1(-) hard stop status
     [......] [.......  ...][][] [.......  ...][][] ↑ Axis 1(+) hard stop status
     [......] [.......  ...][][] [.......  ...][]↑ Elevation Down(-) position status
     [......] [.......  ...][][] [.......  ...]↑ Elevation Up(+) position status
     [......] [.......  ...][][] ↑ 12 bits ( high - low ) unsigned, Elevation position, 0.1°
     [......] [.......  ...][]↑ Rotation Left(-) position status
     [......] [.......  ...]↑ Rotation Right(+) position status
     [......] ↑ 12 bits ( high - low ) unsigned, Rotation position, 0.1°
     ↑ 8 bits, message type, 00000001 = 01hex
```

(2 bit status field values: 00 = Off, 01 = On, 10 = Error, 11 = N/A)

Example – [header] $18EFFF80_{hex}$ [8 data bytes] $011D940371000041_{hex}$

This message from the device at CAN node address 80_{hex} indicates the status of the unit:

Rotation position is +47.3°

[Rotation position = (high – low) 00011101 1001.... =
(unsigned 12 bit) ....0001 11011001 = $1D9_{hex}$ = 473 = 47.3°]

[Rotation Right (+) position status = ...01.. = On]

Elevation position is -5.5°

[Elevation position = (high – low) 00000011 0111.... =
(unsigned 12 bit) ....0000 00110111 = 037_{hex} = 55 = 5.5°]

[Elevation Down (-) position status = ...01 = On]

Axis 5 is at its maximum possible positive position

[Axis 5 (+) hard stop status = 01..... = On]

Unit has been calibrated

[Calibration status = ...01 = On]

Refer to Appendix C for default axis usage for Akron Brass devices.

If Operational error status is 01 (On), the user should perform a Fault Status poll (02_{hex}) and refer to pCode error code to determine required action.

Status Messages – Broadcast (continued)

Message type 81_{hex} – Rotation Soft Limits

This message is issued periodically (100ms) by the 6032 Universal II for monitors with analog position feedback to allow the 6036 Direction Indicator to display programmed soft limits or oscillation limits.

Header per General Message Characteristics, Destination node = FF_{hex} (Broadcast)

Data length – 6 bytes

```
Byte    1        2        3        4        5        6
 Bit 8......1 8......1 8......1 8......1 8......1 8......1
     [......] [....... .......] [....... .......] |[.....]
     [......] [....... .......] [....... .......] |↑ 7 bits ( unused )
     [......] [....... .......] [....... .......] ↑ Oscillation status
     [......] [....... .......] ↑ 16 bits (high – low) signed, Rotation Right Soft Limit, 0.1°
     [......] ↑ 16 bits (high - low) signed, Rotation Left Soft Limit, 0.1°
     ↑ 8 bits, message type, 10000001 = 81hex
```

Example – [header] $18EFFF80_{hex}$ [6 data bytes] $81FED401C280_{hex}$

This message from the Universal II at CAN node address 80_{hex} indicates the status of the unit:

Oscillation Left Limit is -30.0°

[Rotation Left Soft Limit = (high – low) 11111110 11010100 =
(signed 16 bit) 11111110 11010100 = $FED4_{hex}$ = -300 = -30.0°]

Oscillation Right Limit is +45.0°

[Rotation Right Soft Limit = (high – low) = 00000001 11000010 =
(signed 16 bit) 00000001 11000010 = $01C2_{hex}$ = +450 = +45.0°]

Unit is oscillating

[Oscillation status = 1.....]

Message type 82_{hex} – Elevation Soft Limits

This message is issued periodically (100ms) by the 6032 Universal II for monitors with analog position feedback to allow the 6036 Direction Indicator to display programmed soft limits.

Header per General Message Characteristics, Destination node = FF_{hex} (Broadcast)

Data length – 6 bytes

```
Byte    1        2        3        4        5        6
 Bit 8......1 8......1 8......1 8......1 8......1 8......1
     [......] [....... .......] [....... .......] [......]
     [......] [....... .......] [....... .......] ↑ 8 bits ( unused )
     [......] [....... .......] ↑ 16 bits (high - low) signed, Elevation Up Soft Limit, 0.1°
     [......] ↑ 16 bits (high - low) signed, Elevation Down Soft Limit, 0.1°
     ↑ 8 bits, message type, 10000010 = 82hex
```

Example – [header] $18EFFF80_{hex}$ [6 data bytes] $82FF38038400_{hex}$

This message from the Universal II at CAN node address 80_{hex} indicates the status of the unit:

Elevation Down Soft Limit is -20.0°

[Elevation Down Soft Limit = (high – low) 11111111 00111000 =
(signed 16 bit) 11111111 00111000 = $FF38_{hex}$ = -200 = -20.0°]

Elevation Up Soft Limit is +90.0°

[Elevation Up Soft Limit = (high – low) = 00000011 10000100 =
(signed 16 bit) 00000011 10000100 = 0384_{hex} = +900 = +90.0°]

Status Messages – Broadcast (continued)

Message type 83_{hex} – CAN Digital Outputs

This message is issued periodically (100ms) by 6032 Universal II, 6040 Forestry, 6042 Global Platform and 6052 Stream-Master II controllers to make selected digital output status available to other devices.

Header per *General Message Characteristics*, Destination node = FF_{hex} (Broadcast)

Data length – 3 bytes

```
Byte    1        2        3
 Bit 8......1 8......1 8......1
     [......] ........ ........
     [......] ........ .......↑ Status (panel) LED
     [......] ........ ......↑ Status (point-aim) LED
     [......] ........ .....↑ Monitor not in exclusion zone (ladder avoidance)
     [......] ........ ....↑ Remote mode (portable battery powered monitor)
     [......] ........ ...↑ Status (wireless hand-held) LED
     [......] ........ ..↑ Monitor high-flow status
     [......] ........ .↑ Discharge valve status
     [......] ........ ↑ Aux agent dispense status
     [......] ......↑↑ Electric riser command bits
     [......] .....↑ ( reserved )
     [......] ....↑ SOE Mode Active (If 1, Permissive Req'd For Motion)
     [......] ...↑ Point/Aim Mode Active
     [......] ..↑ Monitor Stowed
     [......] .↑ Monitor Active status (Not Stowed)
     [......] ↑ Enable input status
     ↑ 8 bits, message type, 10000011 = 83hex
```

Example – [header] $18EFFF80_{hex}$ [3 data bytes] $83C051_{hex}$

This message from the device at CAN node address 80_{hex} indicates the status of the unit:

CAN Device Status LED is on
[CAN Device Status LED output = ...1]
Wireless HHRC LED is on
[Wireless HHRC LED output = ...1....]
Discharge valve is open
[Discharge Status = ...1.....]
Monitor is active (not stowed)
[Monitor Active = .1..... .....]
Enable is asserted (allowing motion)
[Enable Asserted = 1..... .....]

Message type 84hex – CAN Digital Outputs

This message is issued periodically (100ms) by the 6033 Mini Universal controller to make selected digital output status available to other devices.

Header per *General Message Characteristics,* Destination node = FF_{hex} (Broadcast)

Data length – 3 bytes

```
Byte    1        2        3
 Bit 8......1 8......1 8......0
     [......] |||||||| |||||||
     [......] |||||||| |||||||↑ Riser Retracted status output
     [......] |||||||| ||||||↑ Riser Extended status output
     [......] |||||||| |||||↑ ( reserved )
     [......] |||||||| ||||↑ ( reserved )
     [......] |||||||| |||↑ ( reserved )
     [......] |||||||| ||↑ ( reserved )
     [......] |||||||| |↑ ( reserved )
     [......] |||||||| ↑ ( reserved )
     [......] |||||||↑ ( reserved )
     [......] ||||||↑ ( reserved )
     [......] |||||↑ ( reserved )
     [......] ||||↑ ( reserved )
     [......] |||↑ ( reserved )
     [......] ||↑ ( reserved )
     [......] |↑ Device Faulted status output
     [......] ↑ Device Active status output
     ↑ 8 bits, message type, 10000100 = 84hex
```

Example – [header] $18EFFF81_{hex}$ [3 data bytes] 848002_{hex}

This message from the 6033 Mini Universal at CAN node address 81_{hex} indicates the status of the unit:

Riser controller is active

[Device Active status output = 1..... .....]

Riser is fully extended

[Riser Extended status output = ...1.]

Message type 85hex – CAN Digital Inputs

This message is issued periodically (100ms) by the 6039 CAN I/O module to make external digital input status available to devices on the CAN bus.

Header per *General Message Characteristics*; Destination node = FF_{hex} (Broadcast)

Data length – 2 bytes

```
Byte    1        2
 Bit 8......1 8......1
     [......] ||||||||
     [......] |||||||↑ Allowed in collision zone ( ladder avoidance )
     [......] ||||||↑ Deploy
     [......] |||||↑ Stow
     [......] ||||↑ ( reserved )
     [......] |||↑ ( reserved )
     [......] ||↑ ( reserved )
     [......] |↑ ( reserved )
     [......] ↑ ( reserved )
     ↑ 8 bits, message type, 10000101 = 85hex
```

Example – [header] $18EFFFD0_{hex}$ [2 data bytes] 8501_{hex}

This message from the 6039 CAN I/O module at CAN node address $D0_{hex}$ indicates the input status:
Monitor is allowed to be in the collision zone
[Allowed in collision zone = ...1]

Message type $A1_{hex}$ – SOE Permissive Inputs

This message must be issued periodically (100ms, 300ms timeout) by a user supplied Master Controller.

Header per General Message Characteristics, Destination = Akron Brass Monitor node, Source = Integrator node

Data length – 2 bytes

```
Byte      1         2       3
Bit    8......1 8......1 8......1
[......] ........ ........
[......] ........ .......↑ AX1+ (1 = permission to move in + direction) Right
[......] ........ ......↑ AX1- (1 = permission to move in - direction) Left
[......] ........ .....↑ AX2+ (1 = permission to move in + direction) Up
[......] ........ ....↑ AX2- (1 = permission to move in - direction) Down
[......] ........ ...↑ ( reserved )
[......] ........ ..↑ ( reserved )
[......] ........ .↑ ( reserved )
[......] ........ ↑ ( reserved )
[......] .......↑ ( reserved )
[......] ......↑ ( reserved )
[......] .....↑ ( reserved )
[......] ....↑ ( reserved )
[......] ...↑ ( reserved )
[......] ..↑ ( reserved )
[......] .↑ ( reserved )
[......] ↑ ( reserved )
↑ 8 bits, message type, 10100001 = A1hex
```

Example – [header] $18EF8005_{hex}$ [3 data bytes] $A10009_{hex}$

This message to be sent from the integrator device at CAN node address 05_{hex} to the monitor controller at CAN node address 80_{hex} to limit monitor movement:

Monitor Elevation is permitted to move down, but not up
Monitor Rotation is permitted to move right, but not left.

Status Messages – Poll / Response

J1939 PGN 59904 – Field Request

```
Header 2   2 2      1 1      0 0      0
   Bit 9...5 4......7 6......9 8......1
       [.]|| [......] [......] [......]
       [.]|| [......] [......] ↑ 8 bits, Source node address
       [.]|| [......] ↑ 8 bits, PDU specific, Destination node ( FF(hex) if Broadcast )
       [.]|| ↑ 8 bits, PDU format, 11101010 = EA(hex) = 234 = PGN EA00(hex) = 59904
       [.]|↑ 1 bit, Data Page, 0
       [.]↑ 1 bit, Extended Data Page, 0
       ↑ 3 bits, Priority, 110 = 6
```

Data length – 3 bytes

```
Byte    1        2        3
Bit 8......1 8......1 8......1
    [....... ........ .......]
    ↑ 24 bits, PGN, ( low – mid – high ) 00000000 11101110 00000000 =
                    ( unsigned 24 bit ) 00000000 11101110 00000000 = EE00hex = 60928 = Name field
```

Example – [header] 18EAFFFEhex [3 data bytes] $00EE00_{hex}$

This message requests all (capable) devices to broadcast their J1939 Name field.

Note that Akron Brass devices support this request only for the Name field, PGN 60928 ($EE00_{hex}$).

Status Messages – Poll / Response (continued)

J1939 PGN 60928 – Name Field

```
Header 2   2 2      1 1      0 0      0
   Bit 9...5 4......7 6......9 8......1
       [.]|| [......] [......] [......]
       [.]|| [......] [......] ↑ 8 bits, Source node address
       [.]|| [......] ↑ 8 bits, PDU specific, Destination node, 11111111 = FF hex = Broadcast
       [.]|| ↑ 8 bits, PDU format, 11101110 = EE hex = 238 = PGN EE00 hex = 60928
       [.]|↑ 1 bit, Data Page, 0
       [.]↑ 1 bit, Extended Data Page, 0
       ↑ 3 bits, Priority, 110 = 6
```

Data length – 8 bytes

```
Byte    1        2        3        4        5        6        7        8
 Bit 8......1 8......1 8......1 8......1 8......1 8......1 8......1 8......1
     [......] [......] [.][...] [......] [...][.] [......] [.....]| |[.][..]
     [......] [......] [.][...] [......] [...][.] [......] [.....]| |[.]↑ 4 bits, System Instance
     [......] [......] [.][...] [......] [...][.] [......] [.....]| |↑ 3 bits, Industry Group
     [......] [......] [.][...] [......] [...][.] [......] [.....]| ↑ 1 bit, Arbitrary Address Capable
     [......] [......] [.][...] [......] [...][.] [......] [.....]↑ 1 bit, Reserved, 0
     [......] [......] [.][...] [......] [...][.] [......] ↑ 7 bits, Vehicle System
     [......] [......] [.][...] [......] [...][.] ↑ 8 bits, Function
     [......] [......] [.][...] [......] [...]↑ 3 bits, ECU Instance
     [......] [......] [.][...] [......] ↑ 5 bits, Function Instance
     [......] [......] [.][...] ↑ 8 bits, Manufacturer Code (high)
     [......] [......] [.]↑ 5 bits, Identity Number (high)
     [......] [......] ↑ 3 bits, Manufacturer Code (low)
     [......] ↑ 8 bits, Identity Number (mid)
     ↑ 8 bits, Identity Number (low)
```

Example – [header] $18EEFF80_{hex}$ [8 data bytes] $0B1AC22800420090_{hex}$

This message from the device at CAN node address 80_{hex} decodes as follows:

Arbitrary Address Capable = 1……. = 1 (device uses address claiming)

Industry Group = .001…. = …..001 = 01_{hex} = 1 (On-Highway Equipment)

System Instance = ….0000 = 00_{hex} = 0

Vehicle System = 0000000. = .000000 = 00_{hex} = 0

Function = 01000010 = 42_{hex} = 66 (I/O Controller)

Function Instance = 00000… = …00000 = 00_{hex} = 0

ECU Instance = …..000 = 00_{hex} = 0

Manufacturer Code = (low – high) = 110….. 00101000 =
(unsigned 11 bit) …..001 01000110 = 146_{hex} = 326 (Akron Brass)

Identity Number (low – high) = 00001011 00011010 …00010 =
(unsigned 21 bit) ...00010 00011010 00001011 = $021A0B_{hex}$ =
Low 21 bits of PIN digits 7-15 field of record type FF_{hex} for PIN '1032AE100801035'
(used to make NAME field as unique as possible per SAE J1939-81 Network Management)

Status Messages – Poll / Response (continued)

Message type 02_{hex} – Fault Status Poll

Header per General Message Characteristics, Destination = Akron Brass node, Source = Integrator node

Data length – 1 byte

```
Byte    1
 Bit 8......1
     [......]
     ↑ 8 bits, message type, 00000010 = 02hex
```

Example – [header] $18EF8005_{hex}$ [1 data byte] 02_{hex}

This message requests fault status from the unit at CAN node address 80_{hex} to be sent to the Integrator unit at CAN node address 05_{hex}.

Message type 02_{hex} – Fault Status Response

Header per General Message Characteristics, Destination = Integrator node, Source = Akron Brass node

Data length – 7 bytes

```
Byte    1        2        3        4        5        6        7
 Bit 8......1 8......1 8......1 8......1 8......1 8......1 8......1
     [......] |||[...] |||[..]| |||[..]| |||||||| [....... .......]
     [......] |||[...] |||[..]| |||[..]| |||||||| ↑ 16 bits (high-low) unsigned, pCode error code
     [......] |||[...] |||[..]| |||[..]| |||||||↑ Internal EEPROM fault
     [......] |||[...] |||[..]| |||[..]| ||||||↑ Motor Axis 1 fault
     [......] |||[...] |||[..]| |||[..]| |||||↑ Motor Axis 2 fault
     [......] |||[...] |||[..]| |||[..]| ||||↑ Motor Axis 3 fault
     [......] |||[...] |||[..]| |||[..]| |||↑ Motor Axis 4 fault
     [......] |||[...] |||[..]| |||[..]| ||↑ Motor Axis 5 fault
     [......] |||[...] |||[..]| |||[..]| |↑ Digital Output 1 fault
     [......] |||[...] |||[..]| |||[..]| ↑ Digital Output 2 fault
     [......] |||[...] |||[..]| |||[..]↑ Digital Output 3 fault
     [......] |||[...] |||[..]| |||↑ 4 bits ( unused )
     [......] |||[...] |||[..]| ||↑ SCI input overrun
     [......] |||[...] |||[..]| |↑ CAN input overrun
     [......] |||[...] |||[..]| ↑ XGATE processor | code fault
     [......] |||[...] |||[..]↑ pCode syntax error
     [......] |||[...] |||↑ 4 bits ( unused )
     [......] |||[...] ||↑ Internal fuse blown
     [......] |||[...] |↑ 1 bit ( unused )
     [......] |||[...] ↑ External EEPROM fault
     [......] |||↑ 5 bits ( unused )
     [......] ||↑ SCI output overrun
     [......] |↑ CAN output overrun
     [......] ↑ S12 processor | code fault
     ↑ 8 bits, message type, 00000010 = 02hex
```

Example – [header] $18EF0580_{hex}$ [7 data bytes] $0200000000000B_{hex}$

This response from the unit at CAN node address 80hex to the Integrator unit at CAN node address 05hex indicates fault status:

Rotation sensor error

[pCode error code = (high – low) 00000000 00001011 =
(unsigned 16 bit) 00000000 00001011 = $000B_{hex}$ = 11]

Values in pCode error code correspond to Error Flash Codes in the reference manual for the specific device.

Status Messages – Poll / Response (continued)

Message type 03_{hex} – Firmware Version Poll

Header per General Message Characteristics, Destination = Akron Brass node, Source = Integrator node

Data length – 1 byte

```
Byte    1
 Bit 8......1
     [......]
     ↑ 8 bits, message type, 00000011 = 03hex
```

Example – [header] $18EF8005_{hex}$ [1 data byte] 03_{hex}

This message requests firmware version information from the unit at CAN node address 80_{hex} to be sent to the Integrator unit at CAN node address 05_{hex}.

Message type 03hex – Firmware Version Response

Header per General Message Characteristics, Destination = Integrator node, Source = Akron Brass node

Data length – 7 bytes

```
Byte    1        2        3        4        5        6        7
 Bit 8......1 8......1 8......1 8......1 8......1 8......1 8......1
     [......] [......] [......] [......] [......] [......] [......]
     [......] [......] [......] [......] [......] [......] ↑ 8 bits, System minor
     [......] [......] [......] [......] [......] ↑ 8 bits, System major
     [......] [......] [......] [......] ↑ 8 bits, Bootloader minor
     [......] [......] [......] ↑ 8 bits, Bootloader major
     [......] [......] ↑ 8 bits, Fixed Boot minor
     [......] ↑ 8 bits, Fixed Boot major
     ↑ 8 bits, message type, 00000011 = 03hex
```

Example – [header] $18EF0580_{hex}$ [7 data bytes] $03010B02040206_{hex}$

This response from the unit at CAN node address 80_{hex} to the Integrator unit at CAN node address 05_{hex} indicates firmware information:

Fixed Boot version 1.11
- [Fixed Boot major = 00000001 = 01_{hex} = 1]
- [Fixed Boot minor = 00001011 = $0B_{hex}$ = 11]

Bootloader version 2.4
- [Bootloader major = 00000010 = 02_{hex} = 2]
- [Bootloader minor = 00000100 = 04_{hex} = 4]

System code version 2.6
- [System major = 00000010 = 02_{hex} = 2]
- [System minor = 00000110 = 06_{hex} = 6]

Status Messages – Poll / Response (continued)

Message type 04_{hex} – pCode Version Poll

Header per General Message Characteristics, Destination = Akron Brass node, Source = Integrator node

Data length – 1 byte

```
Byte    1
 Bit 8......1
     [......]
     ↑ 8 bits, message type, 00000100 = 04hex
```

Example – [header] $18EF8005_{hex}$ [1 data byte] 04_{hex}

This message requests pCode version information from the unit at CAN node address 80hex to be sent to the Integrator unit at CAN node address 05_{hex}.

Message type 04hex – pCode Version Response

Header per General Message Characteristics, Destination = Integrator node, Source = Akron Brass node

Data length – 7 bytes

```
Byte    1        2        3        4        5        6        7
 Bit 8......1 8......1 8......1 8......1 8......1 8......1 8......1
     [......] [....... .......] [....... .......] [......] [......]
     [......] [....... .......] [....... .......] [......] ↑ 8 bits, pCode minor
     [......] [....... .......] [....... .......] ↑ 8 bits, pCode major
     [......] [....... .......] ↑ 16 bits ( high - low ) unsigned, Subtype number
     [......] ↑ 16 bits ( high - low ) unsigned, Style number
     ↑ 8 bits, message type, 00000100 = 04hex
```

Example – [header] $18EF0580_{hex}$ [7 data bytes] $040D1C000E0204_{hex}$

This response from the unit at CAN node address 80_{hex} to the Integrator unit at CAN node address 05_{hex} indicates pCode information:

3356 Trident
[Style number = (high – low) 00001101 00011100 =
(unsigned 16 bit) 00001101 00011100 = $0D1C_{hex}$ = 3356]

Analog sensors
[Subtype number = (high – low) 00000000 00001110 =
(unsigned 16 bit) 00000000 00001110 = $000E_{hex}$ = 14]

Version 2.4
[pCode major = 00000010 = 02_{hex} = 2]
[pCode minor = 00000100 = 04_{hex} = 4]

Status Messages – Poll / Response (continued)

Message type 05_{hex} – Config Version Poll

Header per General Message Characteristics, Destination = Akron Brass node, Source = Integrator node

Data length – 1 byte

```
Byte    1
 Bit 8......1
     [......]
     ↑ 8 bits, message type, 00000101 = 05hex
```

Example – [header] $18EF8005_{hex}$ [1 data byte] 05_{hex}

This message requests config version information from the unit at CAN node address 80_{he}x to be sent to the Integrator unit at CAN node address 05_{hex}.

Message type 05hex – Config Version Response

Header per General Message Characteristics, Destination = Integrator node, Source = Akron Brass node

Data length – 7 bytes

```
Byte    1        2        3        4        5        6        7
 Bit 8......1 8......1 8......1 8......1 8......1 8......1 8......1
     [......] [....... ........ ........ .......] [......] [......]
     [......] [....... ........ ........ .......] [......] ↑ 8 bits, Config minor
     [......] [....... ........ ........ .......] ↑ 8 bits, Config major
     [......] ↑ 32 bits ( high - low ) unsigned, Config type
     ↑ 8 bits, message type, 00000101 = 05hex
```

Example – [header] $18EF0580_{hex}$ [7 data bytes] $050398690E0204_{he}$x

This response from the unit at CAN node address 80_{hex} to the Integrator unit at CAN node address 05_{hex} indicates Config information:

Trident, Novotechnik sensors

[Config type = (high – low) 00000011 10011000 01101001 00001110 =
(unsigned 32 bit) 00000011 10011000 01101001 00001110 = $0398690E_{hex}$ = 60320014]

Version 2.4

[Config major = 00000010 = 02_{hex} = 2]

[Config minor = 00000100 = 04_{hex} = 4]

Status Messages – Poll / Response (continued)

Message type 21_{hex} – Data Poll for Vista

This message is accepted by 6032 Universal II controllers to provide for the display of certain information on a specifically programmed Vista display.

Header per *General Message Characteristics*, Destination = Monitor node, Source = Vista node

Data length – 1 byte

```
Byte    1
 Bit 8......1
     [......]
     ↑ 8 bits, message type, 00100001 = 21hex
```

Example – [header] $18EF8005_{hex}$ [1 data byte] 21_{hex}

This message from the Vista display at CAN node address 05_{hex} requests the monitor at CAN node address 80_{hex} to provide certain information via one type 21_{hex} record and one type 22_{hex} record.

Message type 21hex – Data Response for Vista

Header per *General Message Characteristics,* Destination = Vista node, Source = Monitor node

Data length – 8 bytes

```
Byte    1        2        3        4        5        6        7        8
 Bit 8......1 8......1 8......1 8......1 8......1 8......1 8......1 8......1
     [......] [....... .]|||||| |||||||| [....... .]|||||| [....... .]||||||↑ Joystick X right positive
     [......] [....... .]|||||| |||||||| [....... .]|||||| [....... .]|||||↑ Joystick X left negative
     [......] [....... .]|||||| |||||||| [....... .]|||||| [....... .]||||↑ Joystick Y back negative
     [......] [....... .]|||||| |||||||| [....... .]|||||| [....... .]|||↑ Joystick Y fwd positive
     [......] [....... .]|||||| |||||||| [....... .]|||||| [....... .]||↑ Joystick sw1
     [......] [....... .]|||||| |||||||| [....... .]|||||| [....... .]↑ Joystick sw2
     [......] [....... .]|||||| |||||||| [....... .]|||||| ↑ Joystick Y position, 10 bits ( hi-lo )
     [......] [....... .]|||||| |||||||| [....... .]||||||↑ Joystick sw3
     [......] [....... .]|||||| |||||||| [....... .]|||||↑ Joystick sw10
     [......] [....... .]|||||| |||||||| [....... .]||||↑ ( reserved )
     [......] [....... .]|||||| |||||||| [....... .]|||↑ ( reserved )
     [......] [....... .]|||||| |||||||| [....... .]||↑ ( reserved )
     [......] [....... .]|||||| |||||||| [....... .]↑ ( reserved )
     [......] [....... .]|||||| |||||||| ↑ Joystick X position, 10 bits ( hi-lo )
     [......] [....... .]|||||| |||||||↑ Digital input 1
     [......] [....... .]|||||| ||||||↑ Digital input 2H
     [......] [....... .]|||||| |||||↑ Digital input 2L
     [......] [....... .]|||||| ||||↑ Digital input 3H
     [......] [....... .]|||||| |||↑ Digital input 3L
     [......] [....... .]|||||| ||↑ Digital input 4H
     [......] [....... .]|||||| |↑ Digital input 4L
     [......] [....... .]|||||| ↑ Digital input 5H
     [......] [....... .]||||||↑ Digital input 5L
     [......] [....... .]|||||↑ Digital output 1
     [......] [....... .]||||↑ Digital output 2
     [......] [....... .]|||↑ Digital output 3
     [......] [....... .]||↑ K1 relay set
     [......] [....... .]↑ K1 relay reset
     [......] ↑ System voltage, 10 bits ( hi-lo )
     ↑ 8 bits, message type, 00100001 = 21hex
```

This response data is formatted for use by a specifically programmed Vista display.

Message type 22_{hex} – Data Response for Vista

Header per *General Message Characteristics,* Destination = Vista node, Source = Monitor node

Data length – 5 bytes

```
Byte    1        2        3        4        5
 Bit 8......1 8......1 8......1 8......1 8......1
     [......] [....... .......] [....... .......]
     [......] [....... .......] ↑ Elevation position, 16 bits ( hi-lo ), unsigned scaled
     [......] ↑ Rotation position, 16 bits ( hi-lo ), unsigned scaled
     ↑ 8 bits, message type, 00100010 = 22hex
```

This response data is formatted for use by a specifically programmed Vista display.

Configuration Messages – Request / Acknowledgement

Message type 41_{hex} – Zero Axis Positions Request

This message is accepted by 6032 Universal II controllers supporting monitors with analog position feedback to allow the integrator to externally set zero axis positions.

Header per General Message Characteristics, Destination = Akron Brass node, Source = Integrator node

Data length – 1 byte

```
Byte    1
 Bit 8......1
     [......]
     ↑ 8 bits, message type, 01000001 = 41hex
```

Example – [header] $18EF8005_{hex}$ [1 data byte] 41_{hex}

This request from the integrator at CAN node address 05_{hex} to the unit at CAN node address 80_{hex} causes the device to set its zero positions for rotation and elevation to their current positions.

Message type 41_{hex} – Zero Axis Positions Acknowledgement

Header per General Message Characteristics, Destination = Integrator node, Source = Akron Brass node

Data length – 1 byte

```
Byte    1
 Bit 8......1
     [......]
     ↑ 8 bits, message type, 01000001 = 41hex
```

Example – [header] $18EF0580_{hex}$ [1 data byte] 41_{hex}

This acknowledgement from the unit at CAN node address 80_{hex} to the integrator at CAN node address 05_{hex} indicates the unit has completed setting its zero positions for rotation and elevation to their current positions.

Configuration Messages – Request / Acknowledgement (continued)

Message type 45_{hex} – Reset Controller

This message is accepted by 6032 Universal II, 6040 Forestry, 6042 Global Platform and 6052 StreamMaster II controllers. Upon receipt, they perform an immediate system reset.

Header per *General Message Characteristics*, Destination = Akron Brass node, Source = Integrator node

Data length – 1 byte

```
Byte    1
 Bit 8......1
     [......]
     ↑ 8 bits, message type, 01000101 = 45hex
```

Example – [header] $18EF8005_{hex}$ [1 data byte] 45_{hex}

This request from the integrator at CAN node address 05_{hex} to the unit at CAN node address 80_{hex} causes the device to reset and perform normal
powerup actions (address claim, Akron Brass announcement, etc.).

Note that this request is *not* acknowledged.

Configuration Messages – Request / Acknowledgement (continued)

Message type 46_{hex} – Enter Controller Setup Mode

This message is accepted by 6032 Universal II, 6040 Forestry, 6042 Global Platform and 6052 StreamMaster II controllers. It causes the controller to enter its setup mode.

Header per *General Message Characteristics*, Destination = Akron Brass node, Source = Integrator node

Data length – 1 byte

```
Byte    1
 Bit 8......1
     [......]
     ↑ 8 bits, message type, 01000110 = 46hex
```

Example – [header] $18EF8005_{hex}$ [1 data byte] 46_{hex}

This request from the integrator at CAN node address 05_{hex} to the unit at CAN node address 80_{hex} causes the device to immediately enter its setup mode.

Note that this is functionally equivalent to activating the Pattern Stream input at powerup.

Message type 46_{hex} – Enter Controller Setup Mode Acknowledgement

Header per *General Message Characteristics*, Destination = Integrator node, Source = Akron Brass node

Data length – 1 byte

```
Byte    1
 Bit 8......1
     [......]
     ↑ 8 bits, message type, 01000110 = 46hex
```

Example – [header] $18EF0580_{hex}$ [1 data byte] 46_{hex}

This acknowledgement from the unit at CAN node address 80_{hex} to the integrator at CAN node address 05_{hex} indicates the unit has accepted the request to enter setup mode.

ASCII Bidirectional Data Pipe Messages (Akron Brass internal / diagnostic use)

Message type FE_{hex} – ASCII Bidirectional Data Pipe Send

Header per General Message Characteristics, Destination = Akron Brass | Diagnostic node, Source = Diagnostic | Akron Brass node

Data length – 2 to 8 bytes

```
Byte    1        2        3        4        5        6        7        8
 Bit 8......1 8......1 8......1 8......1 8......1 8......1 8......1 8......1
     [......] [......] [......] [......] [......] [......] [......] [......]
     [......] [......] [......] [......] [......] [......] [......] ↑ 8 bits, ASCII character
     [......] [......] [......] [......] [......] [......] ↑ 8 bits, ASCII character
     [......] [......] [......] [......] [......] ↑ 8 bits, ASCII character
     [......] [......] [......] [......] ↑ 8 bits, ASCII character
     [......] [......] [......] ↑ 8 bits, ASCII character
     [......] [......] ↑ 8 bits, ASCII character
     [......] ↑ 8 bits, ASCII character
     ↑ 8 bits, message type, 11111110 = FEhex
```

Example – [header] $18EF80F2_{hex}$ [5 data bytes] $FE413F0D0A_{hex}$

This message from the diagnostic tool at CAN node address $F2_{hex}$ to the unit at CAN node address 80_{hex} sends the 4 ASCII characters 'A', '?', CR, LF.

[Character = 01000001 = 41hex = ASCII 'A']
[Character = 00111111 = 3Fhex = ASCII '?']
[Character = 00001101 = 0Dhex = ASCII carriage return]
[Character = 00001010 = 0Ahex = ASCII line feed]

Message type FE_{hex} – ASCII Bidirectional Data Pipe Acknowledgement

Header per General Message Characteristics, Destination = Diagnostic | Akron Brass node, Source = Akron Brass | Diagnostic node

Data length – 1 byte

```
Byte    1
 Bit 8......1
     [......]
     ↑ 8 bits, message type, 11111110 = FE(hex)
```

Example – [header] $18EFF280_{hex}$ [1 data byte] FE_{hex}

This acknowledgement from the unit at CAN node address 80_{hex} to the diagnostic tool at CAN node address $F2_{hex}$ indicates the receipt of the previous ASCII Bidirectional Data Pipe Send.

A transmitting node must wait for receipt of an acknowledgement of each data packet send before transmitting another one. This serves to limit bus usage, keep the message stream in logical order, and prevent the transmitter from overrunning the receiver's input buffer.

Note that this is a simplistic data pipe with minimal error handling intended for transfer of ASCII messages, program downloads, etc. in the background, and as such is not appropriate for control-related messaging. It is described here for completeness, but is not available for integrator use.

Appendix A – Address Claiming

Address claiming, detailed in SAE J1939-81 Network Management, is the technique used by Akron Brass controllers and display devices to acquire a node address on the CAN network. It is generally compatible with devices using fixed node addresses, as the fixed node address range is 7Fhex and below, with the dynamic address range 80hex and above. Akron Brass devices start claiming addresses at 80hex, so as long as integrator devices use node addresses less than 80hex, or use address claiming, there are no node addressing conflicts.

The negotiation normally takes place within 2-3 seconds of system power up, but if 'hot pluggable' devices are added to the bus later, it can occur again at that time. The J1939 spec defines a priority scheme that includes the possibility of a device successfully claiming an address, and then later losing it to a higher priority device.

To address this and relieve the integrator of concerns about the specific mechanics of address claiming, Akron Brass devices issue a type FFhex 'Address Announcement' message upon successful address claim. If they lose their address later and must claim another, they again issue the Address Announcement. The interested integrator node(s) should listen for this message and use its source node address to update their node address for the Akron Brass device.

In the case of multiple Akron Brass devices on a CAN bus, it is up to the integrator to use the PIN field of the address announcement to differentiate among them. One suggested way to do this is to have the integrator device 'remember' the identity of a specific Akron Brass device, and use that to select the address announcement to use for the node address of the device of interest.

The first 4 digits of the PIN are used to identify the Akron Brass device type:

1032 – 6032 Universal II
1033 – 6033 Mini Universal
1034 – 6034 Operator Station
1035 – 6035 Joystick (and some 6041 Switchboxes)
1036 – 6036 Direction Indicator
1037 – 6037 Wireless Interface
1040 – 6040 Forestry controller
1041 – 6041 Switchbox (and some 6035 Joysticks)
1042 – 6042 Global Platform controller (Portable Electric monitor et.al.)
1051 – 6051 CAN Gateway
1052 – 6052 StreamMaster II

If the integrator for some reason cannot (or does not want to) process CAN messages early enough to see the address announcements at power up, they can broadcast the J1939 Name Request (see Status Messages – Poll / Response, J1939 PGN 59904 – Field Request). In response to this message, all addressed devices which have claimed addresses will broadcast a J1939 Name response (see Status Messages – Poll / Response, J1939 PGN 60928 – Name Field) with their information (these may be ignored). After they send this response, Akron Brass devices will additionally broadcast a message type FFhex address announcement, which the integrator can then process as required.

Appendix B – Handling of Joystick Input Data

Joystick inputs are mapped to 6032 Universal II, 6040 Forestry and 6042 Global Platform inputs as follows:

- X axis right - Rotate Right with proportional speed control
- X axis left - Rotate Left with proportional speed control
- Y axis back - Elevate Up with proportional speed control
- Y axis forward - Elevate Down with proportional speed control
- Switch 1 - Pattern Stream
- Switch 2 - Pattern Fog
- Switch 3 - Discharge On
- Switch 4 - Oscillate On | Set
- Switch 5 - Flow High
- Switch 6 - Flow Low
- Switch 7 - Oscillate Pause | Resume
- Switch 8 - Deploy
- Switch 9 - Stow
- Switch 10 - Discharge Off
- Switch 11 - CAFS dry mode
- Switch 12 - CAFS wet mode

Extended (optional)

- Grip X axis – Point Aim rotation target position (±100% = ±175°)
- Grip Y axis – Point Aim elevation target position (±100% = ±90°)
- Switch 13 – (reserved)
- Switch 14 – (reserved)
- Switch 15 – (reserved)
- Switch 16 – (reserved)

The units will accept input requests from multiple sources and will try to honor any and all non-conflicting requests. Conflicting requests will be handled according to the following default priority scheme (meaning input requests from a particular source will override conflicting requests from all lower priority sources):

Highest priority

- Physical input switches (only available on 6032 Universal II)
- CAN device at node address 33_{hex}
- CAN device at node address 34_{hex}
- CAN device at node address 35_{hex}
- CAN device at node address 36_{hex}
- CAN device at node address 37_{hex}
- CAN device at node address 38_{hex}

Lowest priority

By default, all control devices [6035 Joystick, 6037 Wireless Interface, 6041 Switchbox, and any integrator device(s)] appear to Akron Brass controllers as a J1939 joystick at one of the above addresses broadcasting J1939 PGN 64982 messages.

J1939 joysticks broadcast their messages on a periodic basis, every 100ms or less (Akron Brass control devices default to every 20ms). As a safety precaution against control device failure or CAN bus interruption, the controller monitors message timing and if it does not receive any message from an input device (CAN node address) for 200ms, the input buffer for that CAN node is cleared, stopping any manual motion that was being requested by that device.

Appendix C – Default Axis Usage for Akron Brass Devices

6032 Universal II controller

Axis 1: Rotation, (+) = right, (-) = left

Axis 2: Elevation, (+) = up, (-) = down

Axis 3: Pattern, (+) = stream, (-) = fog

Axis 4: Monitor specific
Trident Dual Gallonage: Flow control, (+) = high flow, (-) = low flow
DeckMaster: Swing arm, (+) = deploy, (-) = stow
Others: N/A

Axis 5: Discharge valve, (+) = open, (-) = close

6040 Forestry controller

Axis 1: Rotation, (+) = right, (-) = left

Axis 2: Elevation, (+) = up, (-) = down

Axis 3: Pattern, (+) = stream, (-) = fog

Axis 4: Discharge valve, (+) = open, (-) = close

6042 Global Platform controller

Axis 1: Rotation, (+) = right, (-) = left

Axis 2: Elevation, (+) = up, (-) = down

Axis 3: Pattern, (+) = stream, (-) = fog

6052 StreamMaster II controller

Axis 1: Rotation, (+) = right, (-) = left

Axis 2: Elevation, (+) = up, (-) = down

Axis 3: Pattern, (+) = stream, (-) = fog

AKRON
BRASS COMPANY

Tested... Proven... Trusted.

ISO 9001 REGISTERED COMPANY

PHONE: 330.264.5678 or 800.228.1161 | FAX: 330.264.2944 or 800.531.7335 | akronbrass.com

REVISED: 5/17

WARRANTY AND DISCLAIMER: We warrant Akron Brass products for a period of five (5) years after purchase against defects in materials or workmanship. Akron Brass will repair or replace product which fails to satisfy this warranty. Repair or replacement shall be at the discretion of Akron Brass. Products must be promptly returned to Akron Brass for warranty service.

We will not be responsible for: wear and tear; any improper installation, use, maintenance or storage; negligence of the owner or user; repair or modification after delivery; damage; failure to follow our instructions or recommendations; or anything else beyond our control. WE MAKE NO WARRANTIES, EXPRESS OR IMPLIED, OTHER THAN THOSE INCLUDED IN THIS WARRANTY STATEMENT, AND WE DISCLAIM ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE. Further, we will not be responsible for any consequential, incidental or indirect damages (including, but not limited to, any loss of profits) from any cause whatsoever. No person has authority to change this warranty.