POX-OEM Serial Communication Protocol Revision 0.9

Mediaid Inc,4025 Spencer St, Suite103,Torrance CA 90503 USA

1. Scope

The purpose of this document is to define the serial communication protocol between all PALCO POX OEM boards and OEM equipment.

2. Command/Response packet format

The character format is: one start bit, 8 data bits(lsbit first) one stop bit. Multiple characters make up a command/response packet. The N+2 characters in the command/response packet are: "command"(or “response”), "data1", "data2",..., "dataN", "checksum" N is determined by the command/response and may be zero.

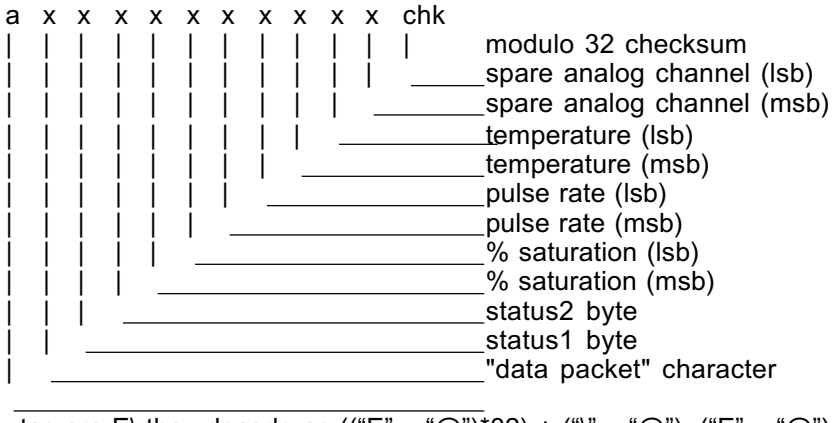
The timing between the "command/response" and "checksum" must not exceed two hundred fifty milliseconds (250 mSec).

The command characters (information from the OEM product to the POX-OEM) are the ASCII characters from “!” to “?”. The response characters (information from the POX_OEM to the OEM product) are the ASCII characters from “a” to “~”. They are all defined in a following section.

All dataN used in the command packets are base 32 digits where the base 32 zero character is ASCII “@”, the base 32 thirty-one character is ASCII “_”. This makes all characters displayable and contiguous. If a number to be transmitted is greater than 31 then it is split into groups of 5 bits and sent most significant first. Example: the number 263 is split as an 8 and 7 [(8*32) + 7 = 263], the 8 is sent first.

The checksum is the negative modulo 32 summation of all the previous characters in the packet (including the command/response) plus the offset of 0x40. That is if all characters are modulo 32 summed in a packet their sum would be zero. (Hint: To create a checksum add all transmitted characters, negate, AND that negated sum with 0x1f and add 0x40, makes all checksum characters “@” through “_”, same as the dataN characters.)
A graphical representation is:

Command Packet Format for “data packet”:



If the two temperature bytes are E\ they decode as ((“E” – “@”)*32) + (“\” – “@”). (“E” – “@”) is equal to 5, multiplied by 32 is 160. Then add (“\” – “@”), which is equal to 28, to get 188. This represents 18.8 degrees centigrade.
An ACK is send upon reception of a valid command packet. A NAK is sent when something is wrong with the packet or it’s contents.

3. Available commands and responses

3.1. Commands to the POX-OEM

3.1.1. ASCII ! data request

!_

3.1.2. ASCII “ reset the POX-OEM

“^

3.1.3. ASCII # data send mode

#@] query mode
#A\ auto send mode(1 second interval)
#B[query time/date stamped
#CZ auto send mode(when new data is ready)
#DY auto send time/date mode (when new data is ready)

3.1.4. ASCII $ Perfusion send interval(incomplete, x = data, chk = checksum)

$ x x chk
| | | |_____ modulo 32 checksum
| | |_____ time (lsb)
| |_____ time (msb)
|_____ "perfusion" character

The perfusion send interval can range from 0 to 255 (truncated if greater than 255), values of 0 through 3 turn perfusion off, each interval is 5msec, minimum interval is 20msec (example: perfusion send interval = 20, time between transmissions is 100msec).

3.1.5. ASCII % diagnostics mode

%[

3.1.6. ASCII & error code request

&Z

3.1.7. ASCII ‘ baud rate change request (this must be the first command received after the reset command is sent(b^), otherwise this command is regarded as an error.

‘@Y 9600 baud
‘AX 4800 baud

3.1.8. ASCII (parametric request

(X

3.1.9. ASCII) software version request

)W

3.1.10. ASCII * serial number request

*V

3.1.11. ASCII + set date and time (incomplete, x = data, chk = checksum)

```
+  x  x  x  x  x  x  chk
|  |  |  |  |  |  |  |  modulo 32 checksum
|  |  |  |  |  |  |  |__minutes(lsb)
|  |  |  |  |  |  |_____minutes(msb)
|  |  |  |  |  |________hours
|  |  |  |  |___________day
|  |  |  |______________month
|  |  |_________________year (offset from 1998)
|  |____________________"data packet" character
|_____________________
```

3.1.12. ASCII , model number request

```
,T
```

3.1.13. ASCII - POX on/off request

```
-@S    POX off
-AR    POX on
```

3.1.14. ASCII . sensor type request (used on models that have removable sensor modules)

```
.R
```

3.1.15. ASCII / perfusion waveform inversion request

```
/@Q    perfusion normal
/AP    perfusion inverted
```

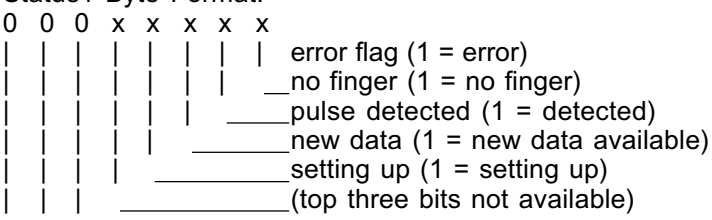
3.2. Responses from the POX-OEM

3.2.1. ASCII a POX-OEM Data(incomplete, x = data, chk = checksum)

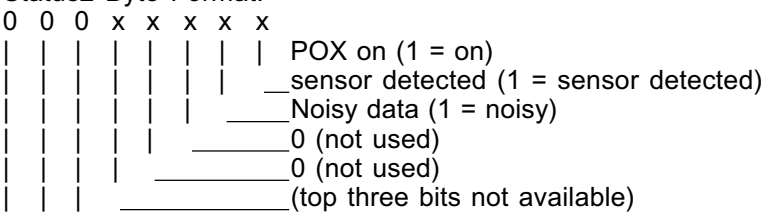
```
a  x  x  x  x  x  x  x  x  x  x  x  chk
|  |  |  |  |  |  |  |  |  |  |  |       modulo 32 checksum
|  |  |  |  |  |  |  |  |  |  |  |_____spare analog channel (lsb)
|  |  |  |  |  |  |  |  |  |  |________spare analog channel (msb)
|  |  |  |  |  |  |  |  |  |___________temperature (lsb)
|  |  |  |  |  |  |  |  |______________temperature (msb)
|  |  |  |  |  |  |  |_________________pulse rate (lsb)
|  |  |  |  |  |  |____________________pulse rate (msb)
|  |  |  |  |  |_______________________% saturation (lsb)
|  |  |  |  |__________________________% saturation (msb)
|  |  |  |_____________________________status2 byte
|  |  |________________________________status1 byte
|  |___________________________________"data packet" character
|______________________________________
```

The status bytes are defined as follows:

Status1 Byte Format:



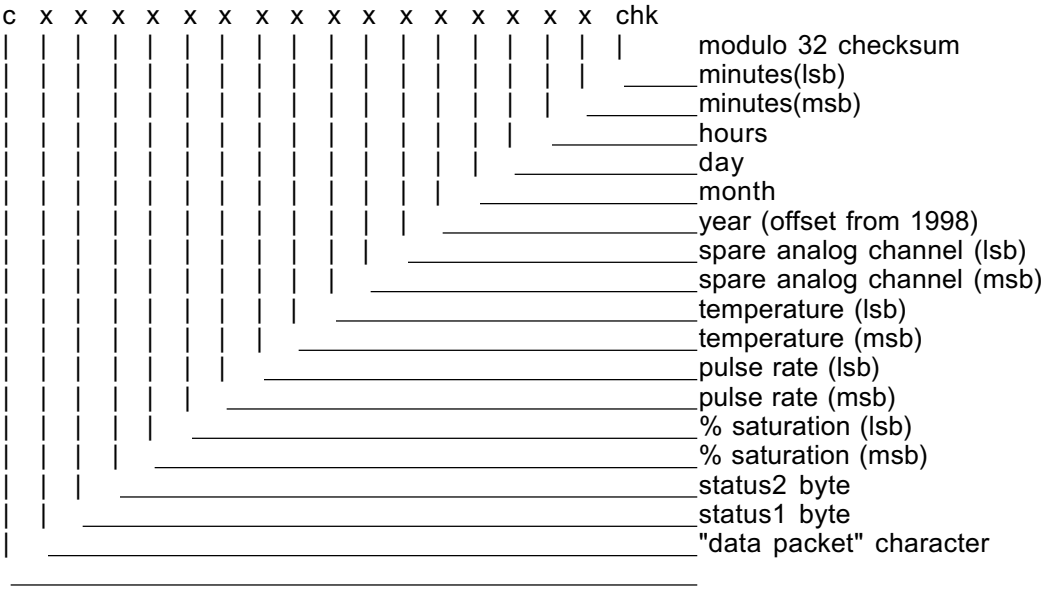
Status2 Byte Format:



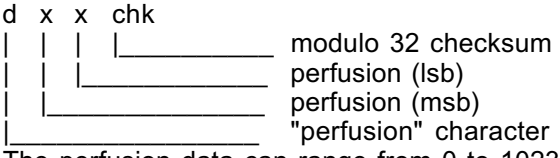
3.2.2. ASCII b POX-OEM power up OK response(complete)

b^

3.2.3. ASCII c POX-OEM Data, Date and time stamped(incomplete, x = data, chk = checksum)



3.2.4. ASCII d Perfusion Data(incomplete, x = data, chk = checksum)



The perfusion data can range from 0 to 1023

3.2.5. ASCII e Error(incomplete, x = data, chk = checksum)

```
e  x  x  chk
|  |  |  |_____________ modulo 32 checksum
|  |  |_______________ error code (lsb)
|  |_________________ error code (msb)
|___________________ "error code" character
```

Error data can range from 0 to 1023. There may be multiple errors in a response.

3.2.5.1. Error codes

3.2.5.1.1. No error, 0

3.2.5.1.2. ROM checksum error, 1(0x001)

3.2.5.1.3. Low power supply, 2(0x002)

3.2.5.1.4. EEPROM error, 4(0x004)

3.2.5.1.5. No red LED, 8(0x008)

3.2.5.1.6. No Ir LED, 16(0x010)

3.2.5.1.7. Thin tissue, 32(0x020)

3.2.5.1.8. Thick tissue, 64(0x040)

3.2.5.1.9. Maximum perfusion, 128(0x080)

3.2.5.1.10. System failure error, 256(0x100)

3.2.5.1.11. No module attached, 512(0x200)

3.2.5.1.12. Analog output calibration failure, 1024(0x400)

3.2.6. ASCII f Parametric(incomplete, x = data, chk = checksum)

```
f  x  x  x  x  x  x  chk
|  |  |  |  |  |  |  |                  modulo 32 checksum
|  |  |  |  |  |  |  _________________red pulse height gain
|  |  |  |  |  |  _____________________red perfusion gain
|  |  |  |  |  _______________________red LED drive
|  |  |  |  _________________________IR pulse height gain
|  |  |  ___________________________IR perfusion gain
|  |  _____________________________IR LED drive
|  _______________________________"parametric" character
 _____________________________
```

3.2.7. ASCII g Version(incomplete, x = data, chk = checksum)

```
g  x  x  x  chk
|  |  |  |  |_________          modulo 32 checksum
|  |  |  |____________  software version (3rd byte - lsb)
|  |  |_______________  software version (2nd byte)
|  |__________________  software version (1st byte - msb)
|_____________________  "s/w version" character
```

The software version number can range from 0 to 32,767.

3.2.8. ASCII h Serial Number(incomplete, x = data, chk = checksum)

```
h  x  x  x  x  x  x  x  x  chk
|  |  |  |  |  |  |  |  |  |            modulo 32 checksum
|  |  |  |  |  |  |  |  |  ____________serial number (lsb)
|  |  |  |  |  |  |  |  _______________serial number
|  |  |  |  |  |  |  ____________________serial number
|  |  |  |  |  |  _____________________serial number
|  |  |  |  |  ______________________serial number
|  |  |  |  _______________________serial number
|  |  |  ________________________serial number
|  |  _________________________serial number (msb)
|  __________________________"serial number" character
```

The serial number can range from 0 to 1,099,511,627,775.

3.2.9. ASCII i Model number(incomplete, x = data, chk = checksum)

```
i  x  x  x  chk
|  |  |  |  |_________          modulo 32 checksum
|  |  |  |____________  model number (lsb)
|  |  |_______________  model number
|  |__________________  model number (msb)
|_____________________  "model number" character
```

The model number can range from 0 to 32,767.

3.2.10. ASCII j NAK(complete)

```
j@V   NAK because of bad command
jAU   NAK because of check sum
jBT   NAK because of internal error
jCS   NAK because of time out
jDR   NAK because of bad parameter
```

These NAK commands (Not AcKnowledge) are transmitted upon there being an error in the last OEM data packet, or, an internal POX error.

3.2.11. ASCII k ACK(complete)

```
kU
```

This ACK command (ACKnowledge) is transmitted upon acceptance of the last transmitted command.

3.2.12. ASCII l Sensor type(incomplete, x = data, chk = checksum)

```
l   x  chk
|   |   |_________      modulo 32 checksum
|   |_____________      sensor type
|_________________      "sensor type" character
```

The sensor type can range from 0 to 31.
1 = M120 finger sensor module
2 = M120 cable adaptor module
3 = RS-232 interface
The sensor type response is a 0 for a request to an M15

3.2.13. ASCII m Request for OEM version number(complete)

mS
This command is used for PALCO testing, not for general use.

3.2.14. ASCII n Request for self test(incomplete, x = data, chk = checksum)

```
n   x  chk
|   |   |_________      modulo 32 checksum
|   |_____________      test type
|_________________      "self test" character
```

The test types available from 0 to 31.
0 = Lo Batt cal
1 = Display test
2 = EEPROM test
3 = Initialization of EEPROM

Revision History

Rev 0.6 Added baud rate modifier
Rev 0.7 Added command "m" for requesting OEM connected rev
Added command "n" for requesting self tests, cannot be used by OEM
Rev 0.8 Added System Failure and No Module Attached to error codes
Rev 0.9 Added error code for analog output calibration failure, used on M15As
Added note in sign-off sheet for electronic copy